轻量级应用安全开发生命周期管理实践

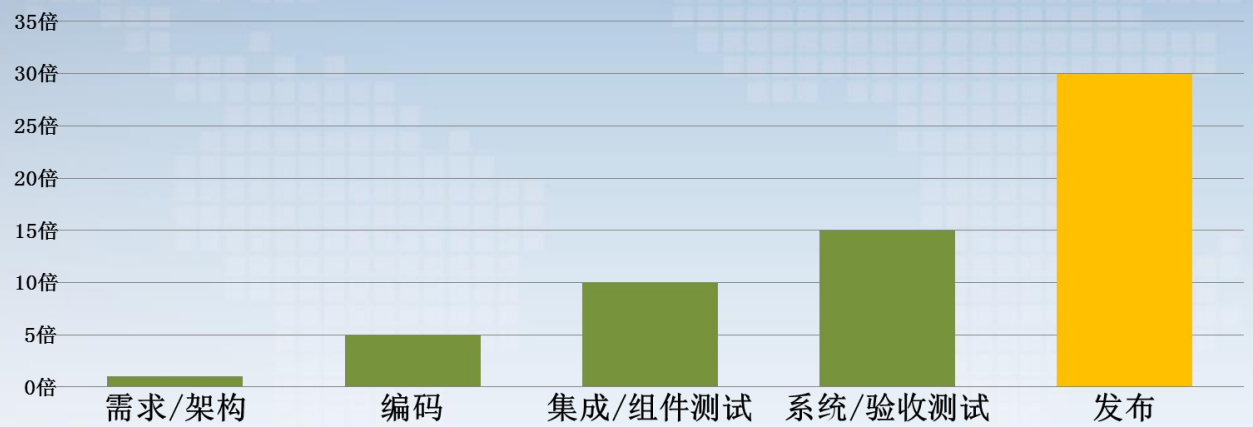
邵元明 国科科技信息安全总监

目录

- **应用安全开发生命周期管理的实施现状**
- 轻量级应用安全开发生命周期管理实践

OWASP
Open Web Application Security Project

这并不是一个新概念



需求调研/分析 → 人员培训 → 代码审计 → 渗透测试 → 设备防护

千行代码缺陷数量

普通软件公司：4～40

高管理软件公司：2~4

美国NASA软件：0.1

OWASP
Open Web Application Security Project

实施的情况如何？

最强王者：应用安全开发生命管理流程已成熟，并逐步向DevSecOps转型
典型代表：互联网公司

永恒钻石：已实现完整的应用安全开发生命管理流程，在标准化与自动化维度持续改进
典型代表：银行

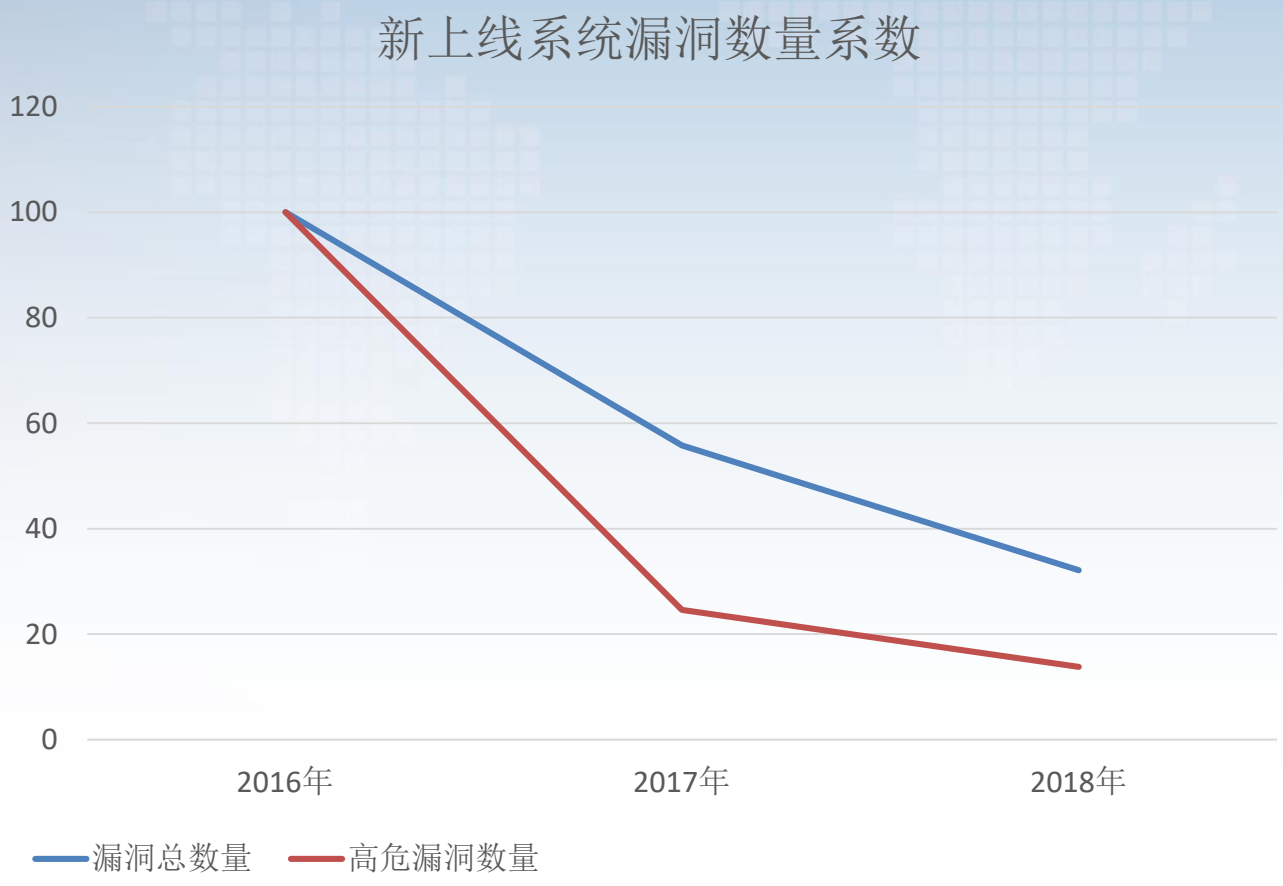
荣耀黄金：将上线测试纳入标准的开发流程，在向完整的应用安全开发生命管理流程探索
典型代表：大型国企、大型的软件开发商

倔强青铜：仅具备简单的应用安全概念，了解简单的安全漏洞，没有专门的安全性测试环节
典型代表：中小型软件开发商、中小型政企单位

OWASP
Open Web Application Security Project

真的有用吗？

某单位实施应用安全开发生命周期管理2年的成果：



新上线系统平均漏洞数量减少**67.9%**

新上线系统平均高危漏洞数量减少**87.2%**

OWASP
Open Web Application Security Project

真的有用吗？

VS

实施前

S：您好，发现了一个XSS注入漏洞，请及时修复
Y：啥？什么叫XSS注入，这是干嘛的？
S：XSS注入漏洞会将不安全的用户输入……blablabla
Y：额，这样啊，那这个要怎么修复呢？
S：XSS注入的修复包括输入过滤和输出编码……blablabla
Y：修复好了你看一下？
S：额，你这个过滤方式有点问题，可以绕过的……
Y：不是吧，那我再改一下
未完待续……

实施后

S：您好，发现了一个XSS注入漏洞，请及时修复
Y：WX，不好意思，我们马上修复
S：好的，有问题随时沟通
Y：修复完了，我们初步验证了一下应该没问题
S：好的，确实修复了，多谢支持

OWASP
Open Web Application Security Project

为什么不做呢？

流程复杂

OWASP Top 10 – 2013 （旧版）	OWASP Top 10 – 2017 （新版）
A1 - 注入	A1 - 注入
A2 - 失效的身份认证和会话管理	A2 - 失效的身份认证和会话管理
A3 - 跨站脚本（XSS）	A3 - 跨站脚本（XSS）
A4 - 不安全的直接对象引用 - 与A7合并	A4 - 失效的访问控制（最初归类在2003/2004版）
A5 - 安全配置错误	A5 - 安全配置错误
A6 - 敏感信息泄漏	A6 - 敏感信息泄漏
A7 - 功能级访问控制缺失 - 与A4合并	A7 - 攻击检测与防护不足(新增)
A8 - 跨站请求伪造（CSRF）	A8 - 跨站请求伪造（CSRF）

规范建设困难

人员培养周期长

投入过大

OWASP
Open Web Application Security Project

目录

- 应用安全开发生命周期管理的实施现状
- 轻量级应用安全开发生命周期管理实践

OWASP
Open Web Application Security Project

实施经验

基本思想

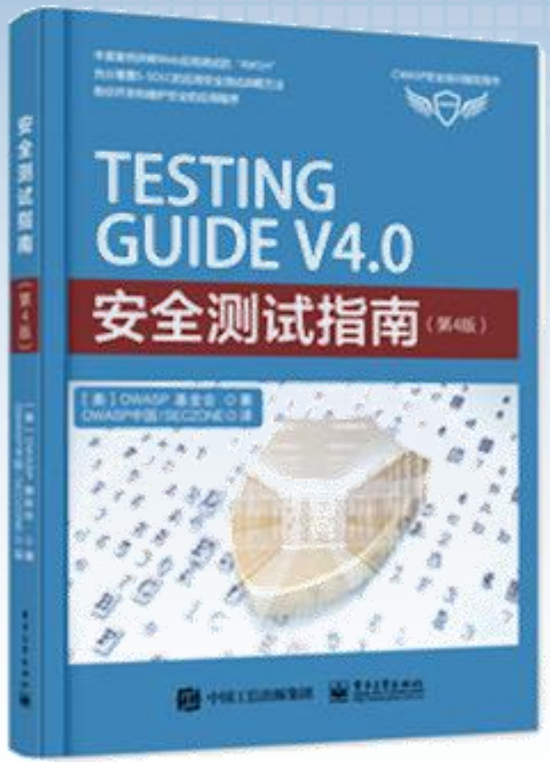
- 依托现有开发流程进行管理，降低流程改造难度
- 工具化的开发规范，**初始阶段解决20%的漏洞类型即可降低80%的漏洞数量。**
- 培训的目标是形成意识和掌握流程，技能是实践中学会的
- 把那些复杂的工作都交给持续改进

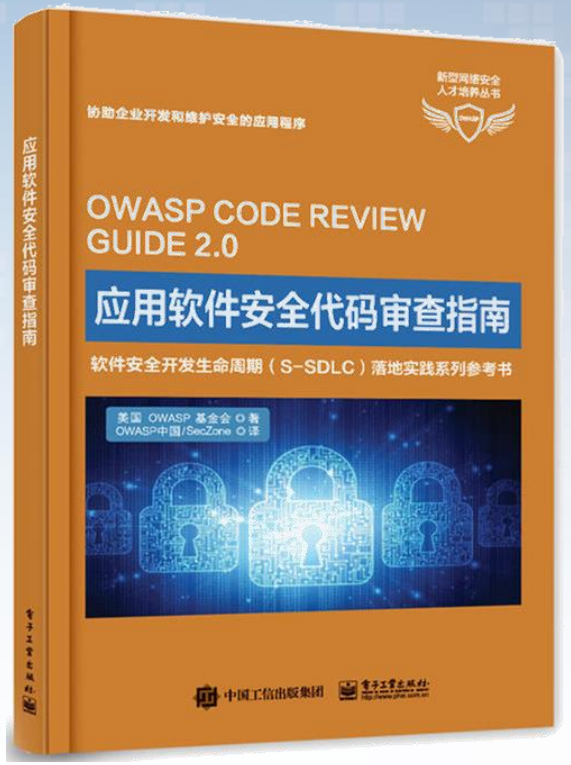
实施前提

- 起码有一个懂安全测试的人员
- 建立了相对规范的开发流程

OWASP
Open Web Application Security Project

参考标准





OWASP
Open Web Application Security Project

整体流程

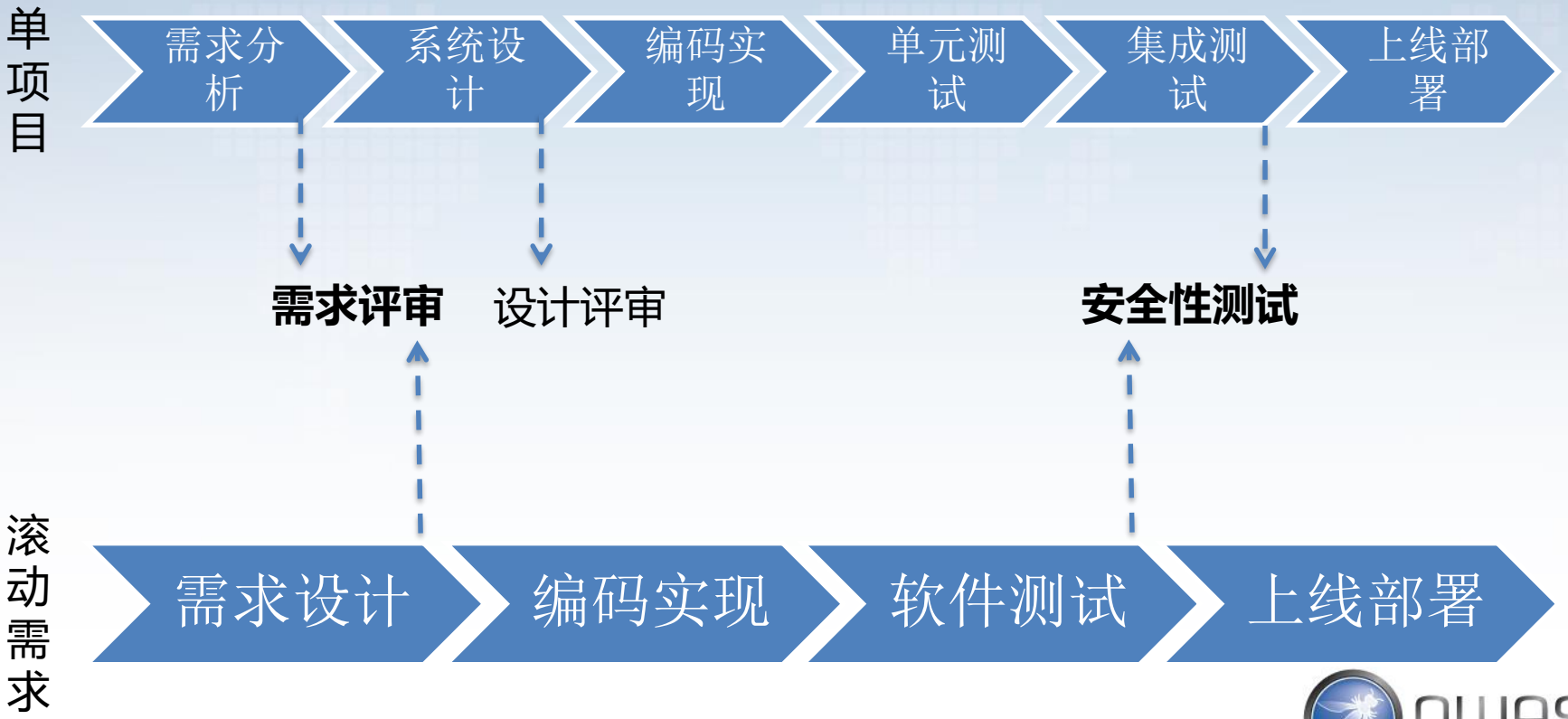
根据实施经验，经过2~4周的准备，即可启动管理流程。

准备阶段	实施阶段	检查阶段	改进阶段
• 流程制定 • 规范制定 • 培训宣贯	• 需求评审 • 开发支撑 • 安全测试 • 日常维护	• 流程审计 • 培训考核	• 技术培训 • 规范优化

OWASP
Open Web Application Security Project

准备阶段——流程制定

在开发过程管理的基础上增加安全性需求评审与安全性测试流程，为了减少流程改动，前期甚至可以考虑并行实施。



OWASP
Open Web Application Security Project

准备阶段——流程制定

在开发过程管理的基础上增加应用安全管理的审计流程

开发过程管理： 需求/设计文档　　软件测试文档

应用安全管理： 安全需求/设计审计　　安全测试文档

OWASP
Open Web Application Security Project

准备阶段——流程制定

培训流程

- 新员工入职须经过安全开发培训，通过考核后方可加入项目
- 制定季度培训计划，每季度开展半天的培训与考核

持续改进流程

- 制定开发规范体系的持续改进流程，建议第一年每季度进行一次修订，之后每半年或一年进行修订即可

OWASP
Open Web Application Security Project

准备阶段——规范制定

难点：如何保证构建的安全开发规范能够全面解决安全问题。

我问大师：“如何保证构建的安全开发规范能够全面解决安全问题？”
大师摇了摇头，伸出2个手指头，默然不语
我恍然大悟：“大师，你是说没有两个资深专家一起支撑是搞不定的？”
大师一竖眉头说到：“瞎扯，我是告诉你一开始别指望全面解决问题，先消除20%常见安全漏洞就够了！”

OWASP
Open Web Application Security Project

准备阶段——规范制定

难点：传统的安全开发规范多基于编码行为，不利于记忆，在需求分析与安全测试环节也难以直接参考。

输入验证：

- 在可信系统（比如：服务器）上执行所有的数据验证
- 识别所有的数据源，并将其分为可信的和不可信的。据库，文件流，等）的数据。
- 应当为应用程序应提供一个集中的输入验证规则
- 为所有输入明确恰当的字符集，比如：UTF-8。
- 在输入验证前，将数据按照常用字符进行编码（*规范*
- 丢弃任何没有通过输入验证的数据。
- 确定系统是否支持 UTF-8 扩展字符集，如果支持，在
- 在处理以前，验证所有来自客户端的数据，包括：所 cookie 名字和数据值）。确定包括了来自 JavaScript、
- 验证在请求和响应的报头信息中只含有 ASCII 字符。
- 核实来自重定向输入的数据（一个攻击者可能向重定应用程序逻辑以及在重定向前执行的任何验证）。
- 验证正确的数据类型。
- 验证数据范围。
- 验证数据长度。
- 尽可能采用“白名单”形式，验证所有的输入。
- 如果任何潜在的*危险字符*必须被作为输入，请确保您特定的安全 API、以及在应用程序中使用的原因。部 <>"'%()&+\\'\"。
- 如果您使用的标准验证规则无法验证下面的输入，那
  - 验证空字节 (%00)；
  - 验证换行符 (%0d, %0a, \r, \n)；
  - 验证路径替代字符“点-点-斜杠”（../或 ..\）。如代字符： %c0%ae%c0%ae/ (使用*规范化*验证双

1输入验证和数据合法性校验

程序接受数据可能来源于未经验证的用户，网络连接和其他不受信任的来源，如果未对程序接受数据进行校验，则可能会引发安全问题。

1.1避免SQL注入

使用PreparedStatement预编译SQL,解决SQL注入问题，传递给PreparedStatement对象的参数可以被强制进行类型转换，确保在插入或查询数据时与底层的数据库格式匹配。

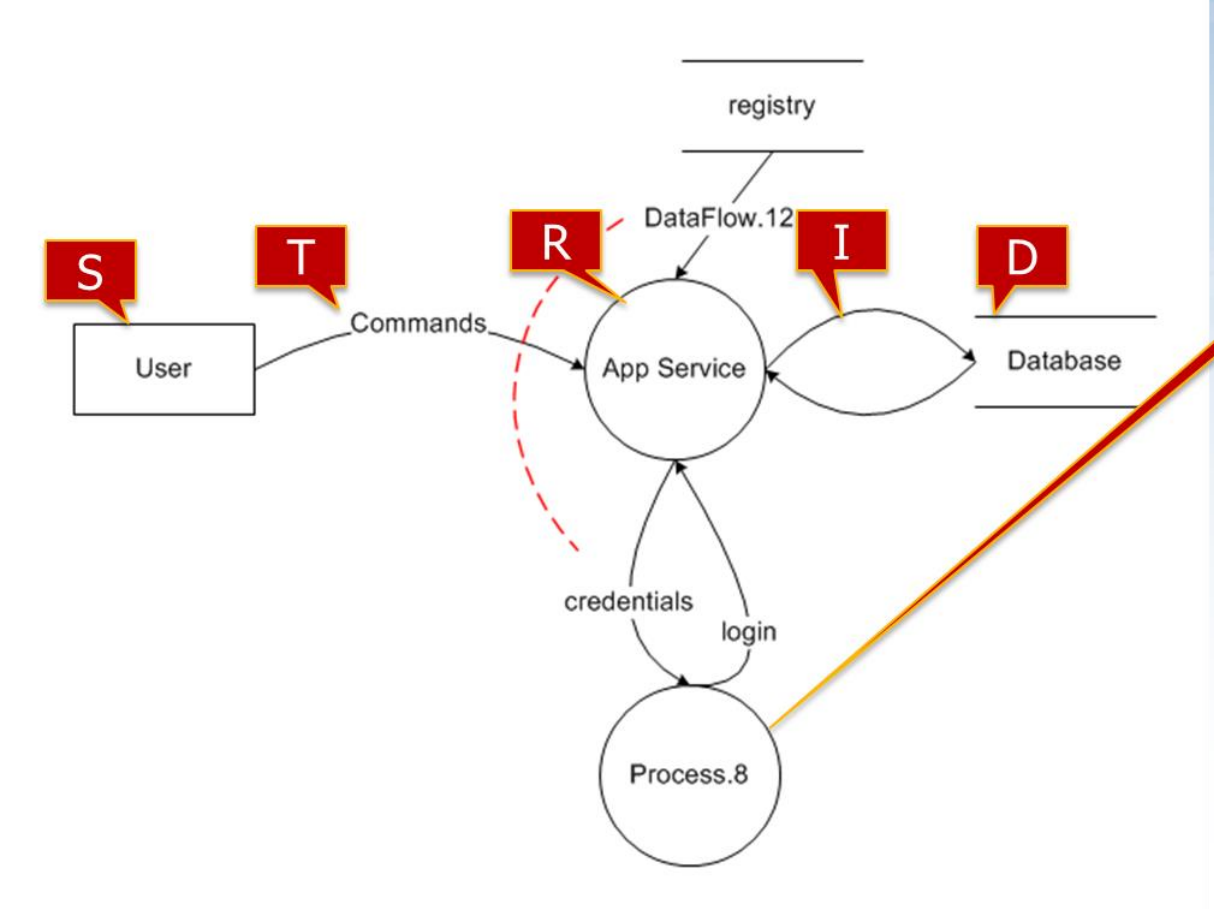
```
String sqlString = "select * from db_user where username=? and password=?";
PreparedStatement stmt = connection.prepareStatement(sqlString);
stmt.setString(1, username);
stmt.setString(2, pwd);
ResultSet rs = stmt.executeQuery();
```

1.2避免XML注入

通过StringBulider 或 StringBuffer 拼接XML文件时，需对输入数据进行合法性校验。 对数量quantity 进行合法性校验，控制只能传入0-9的数字：

```
if (!Pattern.matches("[0-9]+", quantity)) {
    // Format violation
  }
  String xmlString = "<item>\n<description>Widget</description>\n" +
                     "<price>500</price>\n" +
                     "<quantity>" + quantity + "</quantity></item>";
  outStream.write(xmlString.getBytes());
  outStream.flush();
```

准备阶段——规范制定

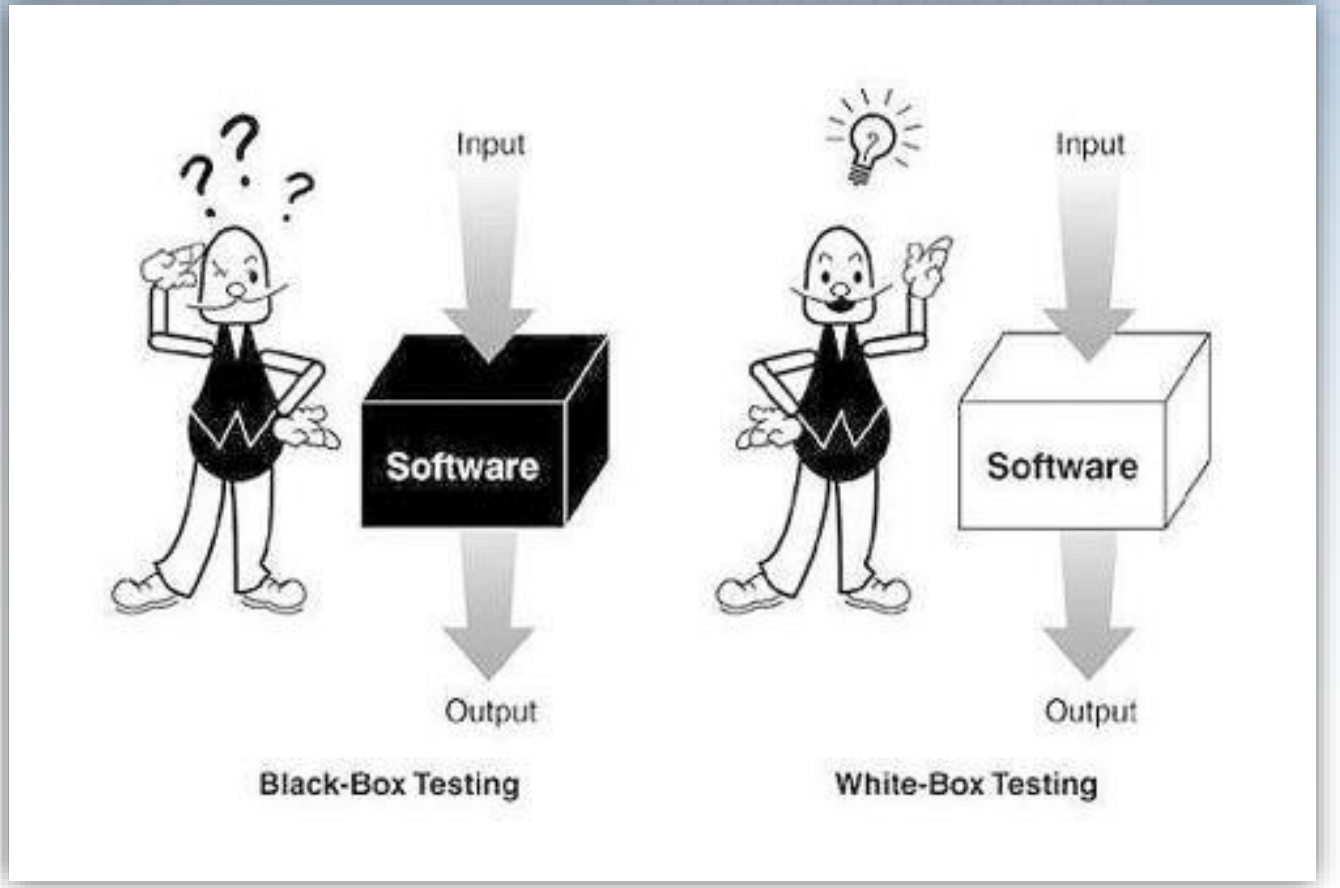


难点：传统的威胁建模方式复杂度较高，学习成本和实施成本都比较高

OWASP
Open Web Application Security Project

准备阶段——规范制定

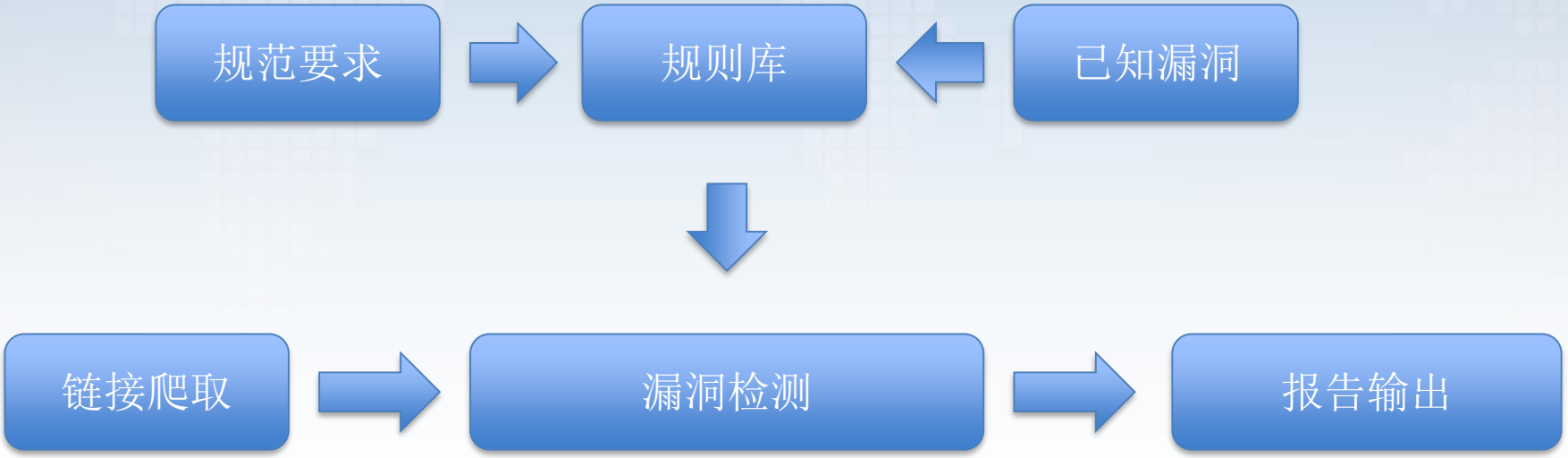
灵感：黑盒与白盒



OWASP
Open Web Application Security Project

准备阶段——规范制定

思路借鉴——WEB应用漏洞扫描器的工作原理，基于链接（功能点）的安全检查

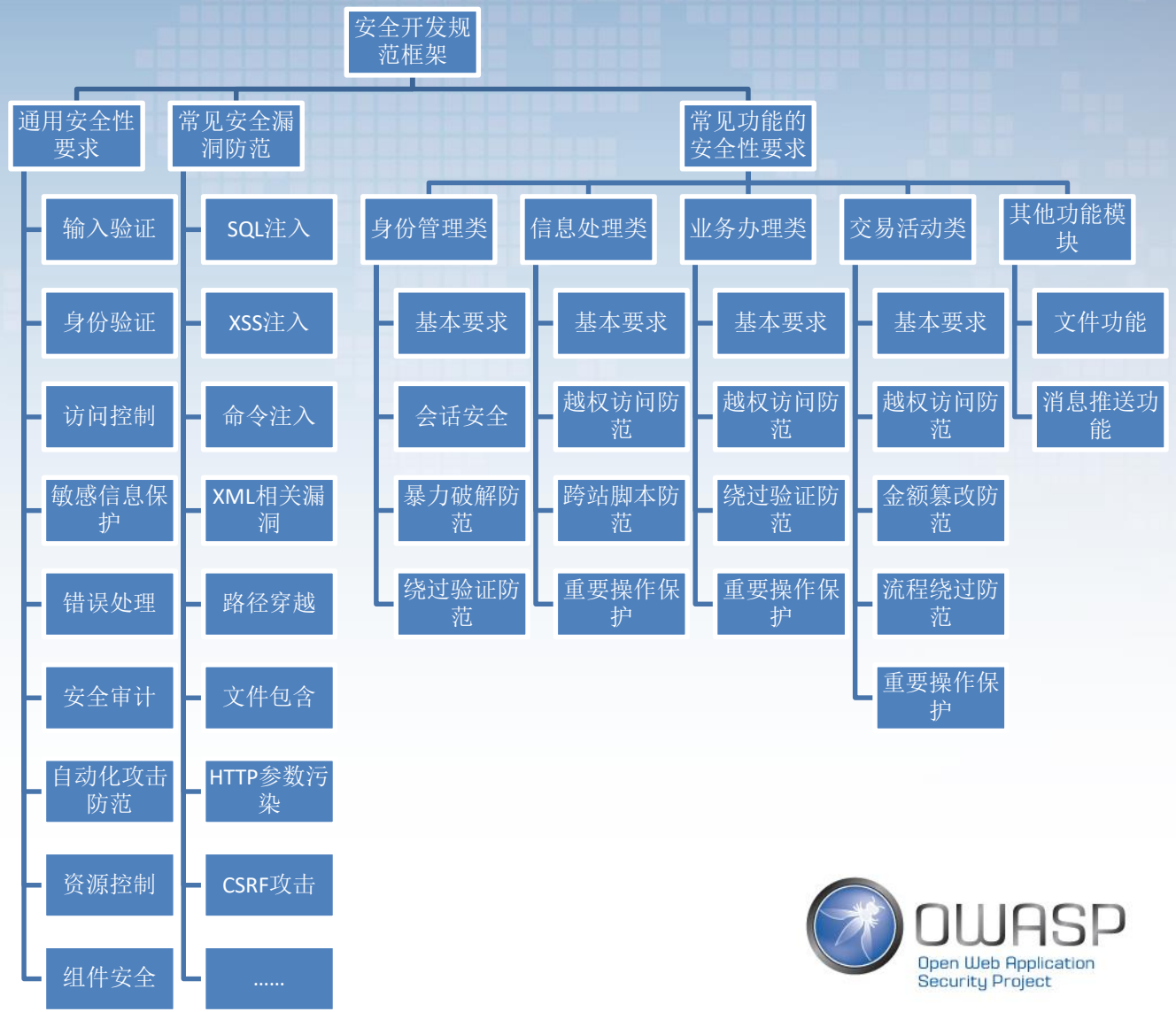


OWASP
Open Web Application Security Project

准备阶段——规范制定

以功能为索引，构建工具型安全开发规范

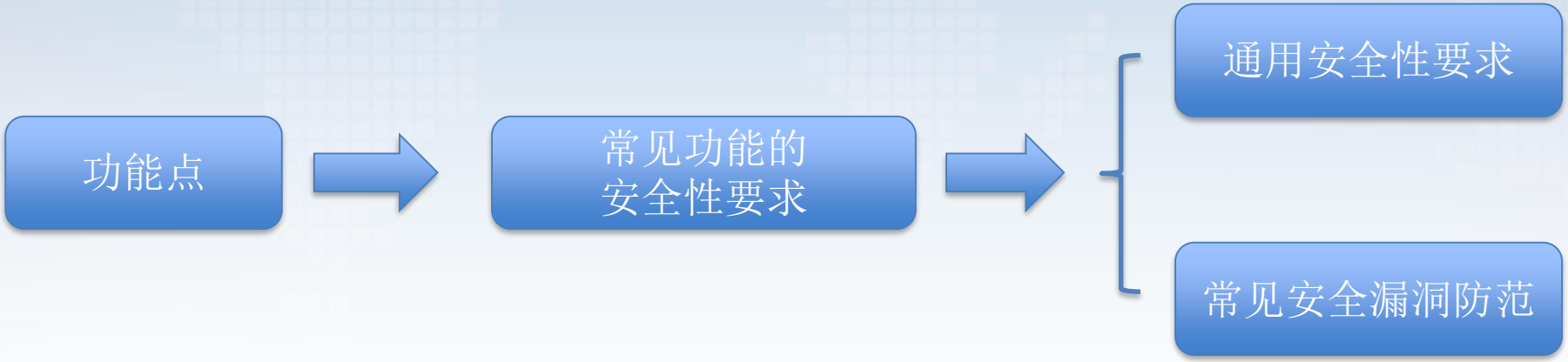
- 安全需求性分析指南
- 安全编码指南
- 安全性测试指南



OWASP
Open Web Application Security Project

准备阶段——规范制定

以功能为索引，构建工具型安全开发规范



OWASP
Open Web Application
Security Project

准备阶段——规范制定

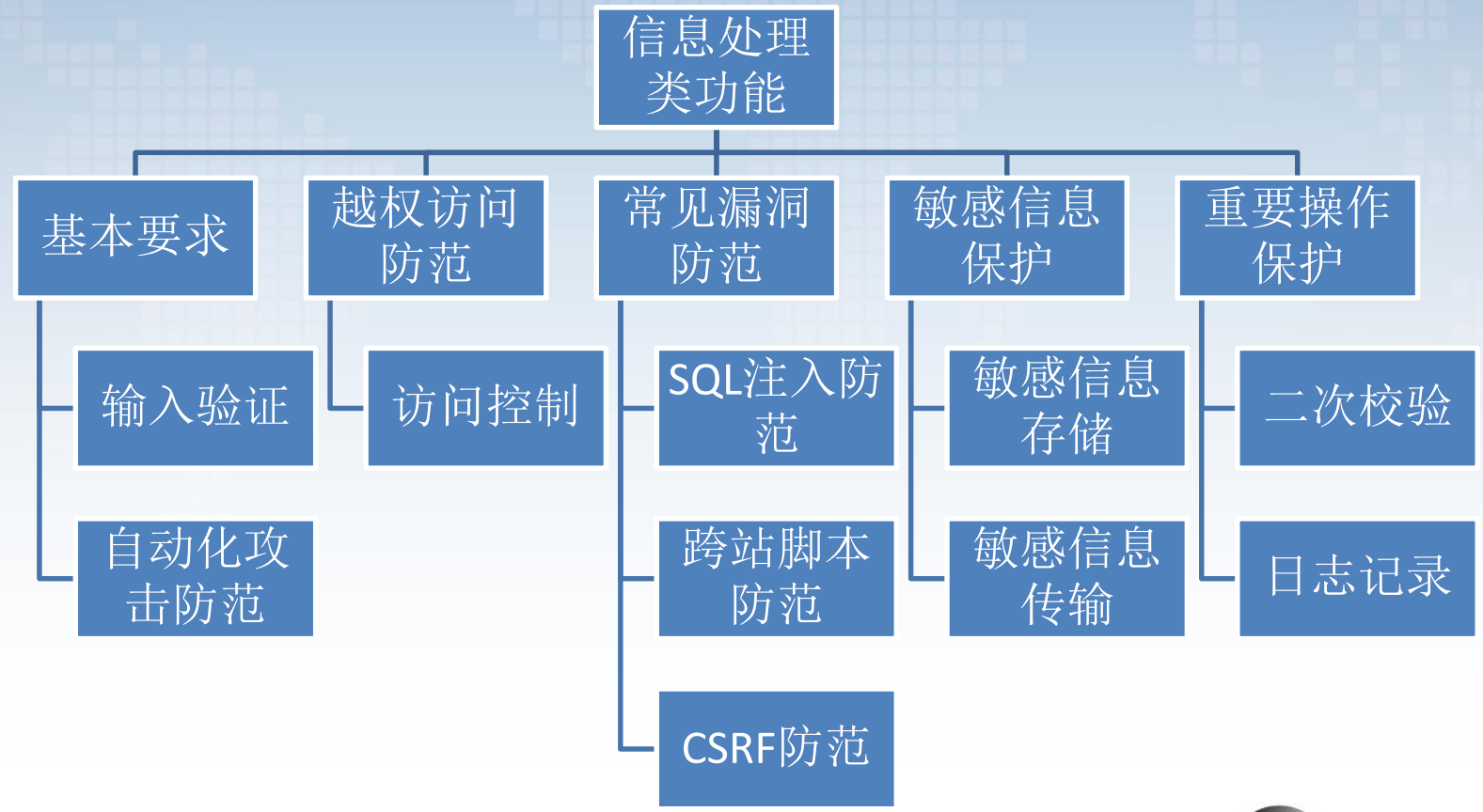
安全开发规范示例

如何实现安全开发？

OWASP
Open Web Application Security Project

准备阶段——规范制定

安全开发规范示例



OWASP
Open Web Application Security Project

准备阶段——规范制定

安全开发规范示例

越权访问防范——需求分析指南：

【基本要求】

信息处理类功能在进行增删改查等操作时应进行权限控制，避免攻击者通过越权访问的方式对其他用户的信息进行操作。

【分析说明】

- 应明确功能是否需要登录后才能访问、哪些用户可以访问以及可访问的数据范围（如只能访问自己的数据，或可以访问自己的数据和某一级账号的数据）。
- 应要求在进行信息处理之前在服务端进行权限校验操作，通常需要校验会话关联的用户是否有操作目标信息的权限。
- 权限校验失败时应进行日志记录，日志记录要求参考2.6安全审计章节

OWASP
Open Web Application Security Project

准备阶段——规范制定

安全开发规范示例

越权访问防范——编码指南：

- 在进行信息处理之前应进行权限校验操作，通常需要在服务端校验用户是否登录、会话关联的用户是否有操作目标信息的权限。
- 对于登录后的操作，建议优先采用基于会话关联的账号+索引的方式操作目标对象，而不是通过引用直接对象标识来操作目标对象，如：
  - 以查询用户基本个人信息为例，引用直接对象标识的方式使用客户端提交的直接对象标识如员工编号或个人信息编号作为参数进行查询并返回结果。而会话关联的账号+索引的方式先从当前会话中获取用户身份信息，基于用户身份在服务端找到对应的员工编号或个人信息编号并直接在服务端查询用户个人信息并返回结果，过程不需要客户端提交员工编号或个人信息编号等直接对象标识。
  - 以查询账户手机号的套餐信息为例，引用直接对象标识的方式将使用客户端提交的直接对象标识如手机号作为参数进行查询并返回结果。而会话关联的账号+索引的方式先从当前会话中获取用户身份信息，基于用户身份信息查询用户关联的手机号并直接在服务端查询该手机号办理的套餐信息，如果用户有多个手机号，客户端只需要提交手机号的索引即可，过程无需提供要查询的手机号。

准备阶段——规范制定

安全开发规范示例

越权访问防范——编码指南：

- 对于不需要登录的信息操作，如果需要通过直接对象标识来访问，直接对象标识应具备足够的复杂度避免攻击者根据自身的直接对象标识推导出其他直接对象标识。如通过匿名方式提交的投诉或报障请求，允许通过单号进行查询时，单号应足够复杂。复杂度要求可参考以下要求：
  - 直接对象标识中随机生成的部分应不少于10位
  - 随机生成的范围应至少包含数字和大小写字母
  - 随机生成的算法应具备足够的随机性，避免攻击者根据已知的标识推导出其他标识
- 权限校验过程与要执行的操作应在同一流程中进行，基于校验结果的下一步操作应在服务端强制执行，不可依赖客户端进行控制。
- 权限校验失败时应进行日志记录，日志记录要求参考2.6安全审计章节

OWASP
Open Web Application Security Project

准备阶段——规范制定

安全开发规范示例

越权访问防范——编码指南：

- 对于不需要登录的信息操作，如果需要通过直接对象标识来访问，直接对象标识应具备足够的复杂度避免攻击者根据自身的直接对象标识推导出其他直接对象标识。如通过匿名方式提交的投诉或报障请求，允许通过单号进行查询时，单号应足够复杂。复杂度要求可参考以下要求：
  - 直接对象标识中随机生成的部分应不少于10位
  - 随机生成的范围应至少包含数字和大小写字母
  - 随机生成的算法应具备足够的随机性，避免攻击者根据已知的标识推导出其他标识
- 权限校验过程与要执行的操作应在同一流程中进行，基于校验结果的下一步操作应在服务端强制执行，不可依赖客户端进行控制。
- 权限校验失败时应进行日志记录，日志记录要求参考2.6安全审计章节

OWASP
Open Web Application Security Project

准备阶段——规范制定

安全开发规范示例

越权访问防范——测试指南：

【测试目标】

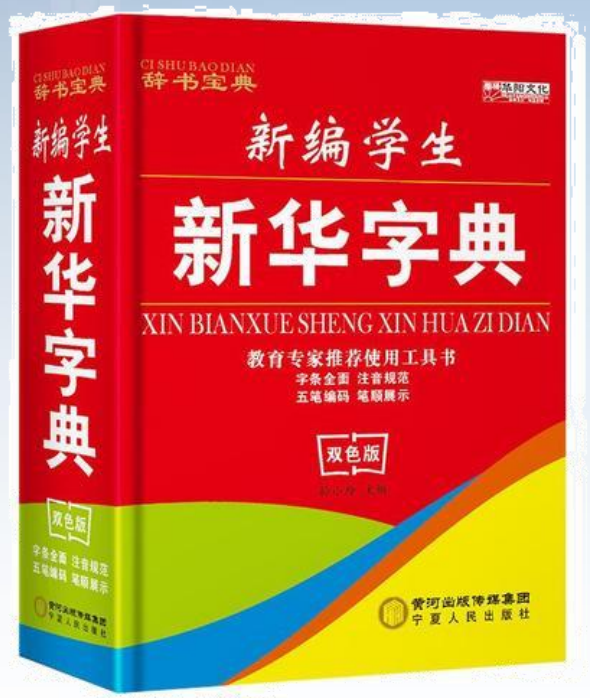
- 测试能否对其他用户的信息进行查询、新增、修改、删除操作。

【测试方法】

- 使用抓包工具跟踪信息查询、新增、修改、删除操作的流程，查看是否存在疑似对象标识，如果存在则尝试使用越权访问测试的方法进行测试，如果能够对其他用户进行新增、修改、删除或可查询用户敏感信息则记录高风险漏洞，如果只能查询敏感性不高的信息则记录中风险漏洞。
- 进行越权访问尝试，并查看越权访问失败时是否有相应的日志记录。
- 本项测试的具体方法可参考通用测试项-权限控制测试的测试方法进行测试。

OWASP
Open Web Application Security Project

准备阶段——技术培训



教一个人把字典背下来的时间：
10年

教一个人如何使用字典的时间：
10分钟

OWASP
Open Web Application Security Project

准备阶段——技术培训

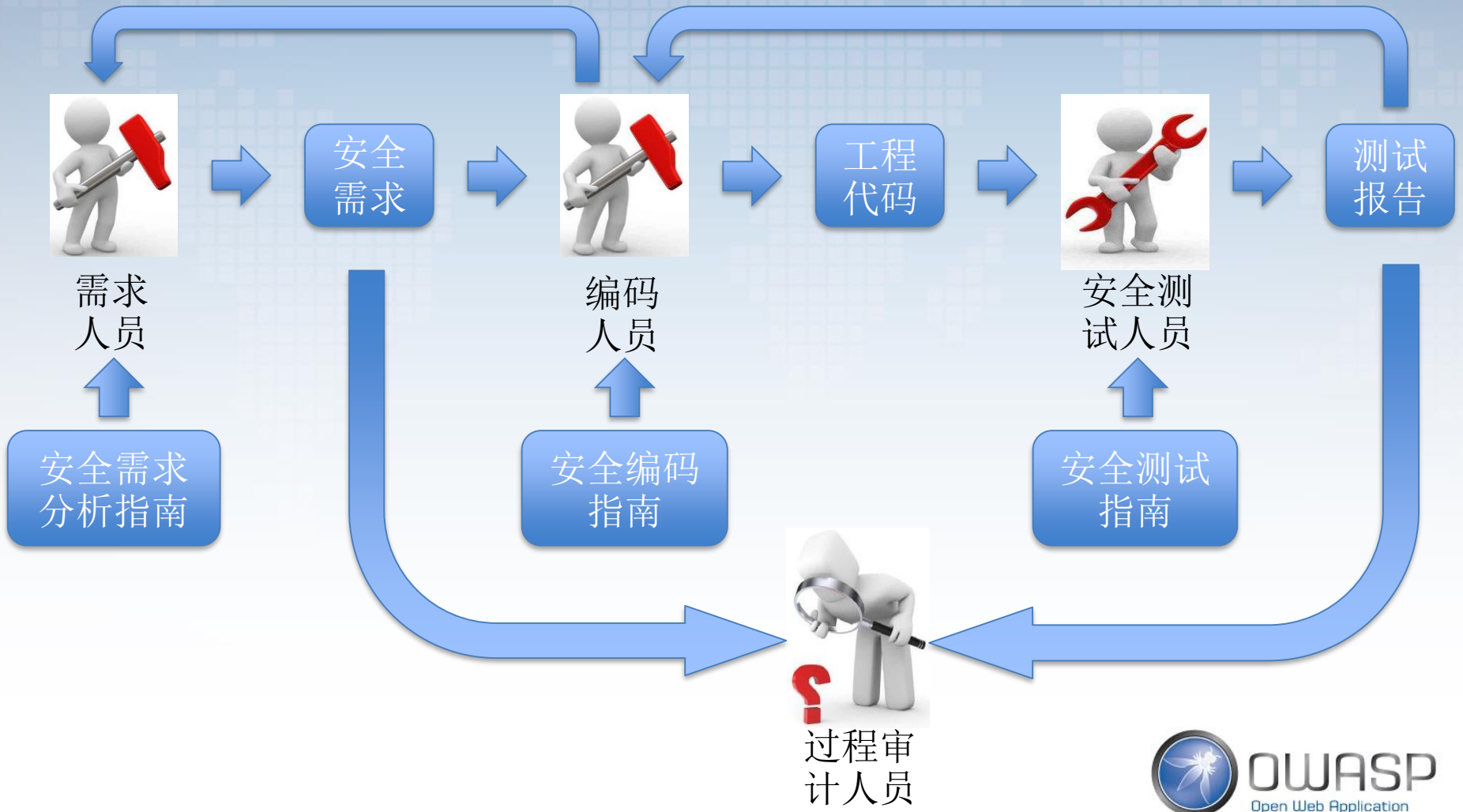
上课时间	课程名称	上课形式	培训对象	课程内容	课时
第1次课	常见漏洞及开发规范	理论+演示	开发流程所有相关人员	1.OWASPTOP10及常见漏洞介绍 2.软件安全开发生命周期管理体系介绍 3.安全开发规范培训	1天
第2次课	安全需求分析指南	理论	需求人员及其他希望参加学习的人员	1.通用安全性需求 2.常见安全漏洞防范 4.常见功能的安全性需求 5.举例说明安全需求分析设计	0.5天
第3次课	安全编码指南	理论	研发人员及其他希望参加学习的人员	1.通用安全性需求 2.常见安全漏洞防范 4.常见功能的安全性需求 5.举例说明安全编码	0.5天
第4次课	安全测试指南	理论+演示	测试人员及其他希望参加学习的人员	1.信息收集 2.漏洞扫描与验证 3.手工漏洞挖掘 4.安全测试报告分析	0.5天
合计					2.5天

培训目标：

- 形成意识
- 了解风险
- 掌握流程
- 理解工具

OWASP
Open Web Application Security Project

实施与审计的管理闭环



OWASP
Open Web Application
Security Project

持续改进

总体流程	需求阶段	编码阶段	测试阶段
• 控制节点强化 • 季度培训	• 需求分析指南优化 • 典型案例补充	• 编码实例 • 标准化组件	• 经典测试用例 • 自动化手段引入

OWASP
Open Web Application Security Project

Thank you !

OWASP
Open Web Application
Security Project