
Secure Sofware Development LifeCycle
企业最佳安全实践
为您构架安全的业务系统

网络攻击的方向

木桶原理

最薄弱环节的攻击

安全攻击，最薄弱环节的攻击；

安全问题，最薄弱环节的集群式爆发；

最新业务的攻击

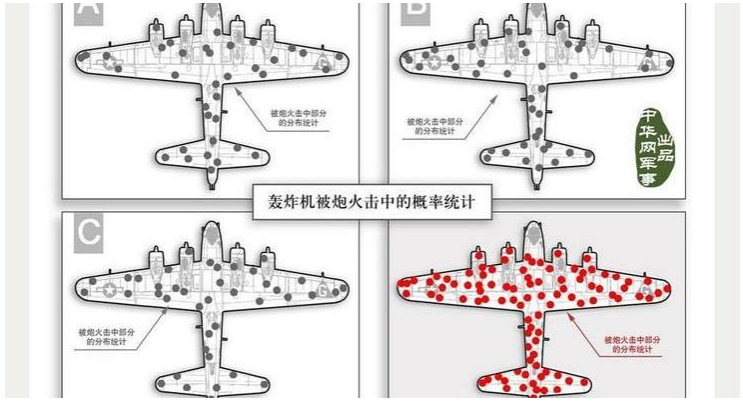
没有考虑安全设计的业务系统，理论上安全风险较高；

没有经过长时间考验的业务系统，实际安全风险较高；

没有经过验证的业务逻辑，带来的新形式的攻击较多。

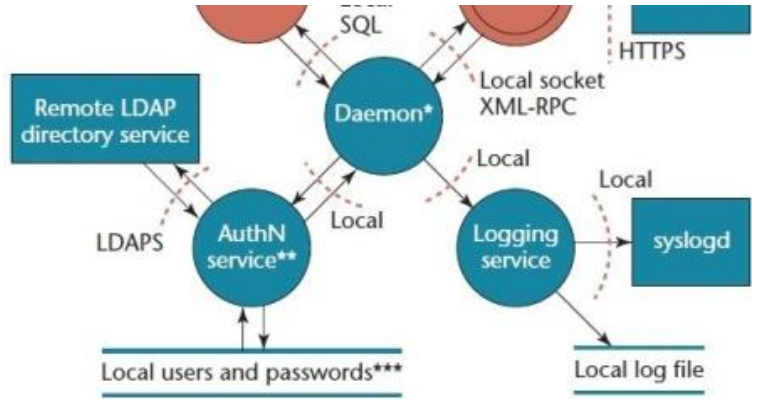
软件安全建设思路

安全防御体系



漏洞防御体系

威胁防御体系



漏洞防御体系

依照漏洞思维来建立的防御系统：

1、既有漏洞的安全经验；

2、测试发现的漏洞情况；

优势：简单、快速、高效；

劣势：不够全面、攻击样板点需要足够、未知风险较高

现状：目前互联网主流模式，大量SRC起来的原因都在此

增加攻击成本

网络安全

降低安全风险

威胁防御体系

通过建立威胁模型，来充分发现软件产品中的威胁，在设计、开发、测试、运维等各个角度来削减威胁，最后达到一个动态的平衡。

优势：系统、全面、成本可控；

劣势：体系建设需要一个周期，全员质量

现状：目前行业仅TOP 100强开始超这个思路建设

面临挑战

01

敏捷开发模式下安全测试

敏捷开发环境，特别在**DevOps**模式下，留给安全测试人员的时间非常有限，无法充分发现安全问题；

02

多项目并行环境

在开发环境中，安全人员数量有限，多项目并行交付、上线过程中，无法进行充分的安全测试；

03

第三方代码带来的安全问题

目前软件开发过程中，会引入大量的第三方组件，第三方组件的安全问题爆发，会导致整个产品的安全问题；

04

应急响应机制

在线系统出现问题后，如何快速解决以及溯源

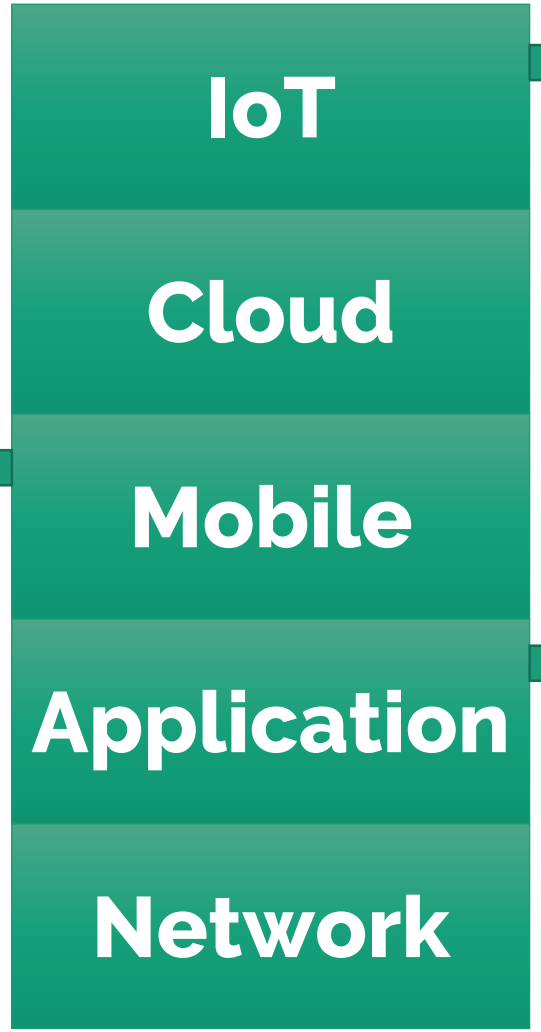
S-SDLC

软件安全开发生命周期

软件安全开发生命周期，主要目的是通过系统的体系，帮助软件开发厂商在需求、设计、开发、测试、部署上线等各个阶段降低安全风险，提升安全能力。

安全风险

Security Risk



IoT TOP 10	
I1 - 不安全的Web界面	I2 - 不完备的认证授权机制
I3 - 不安全的网络服务	I4 - 缺乏传输加密
I5 - 隐私处理存在问题	I6 - 不安全的云环境
I7 - 不安全的移动设备环境	I8 - 不完备的安全配置
I9 - 不安全的软件、固件	I10 - 薄弱的物理安全保护

Mobile TOP 10 Risk
M1-平台使用不当
M2-不安全的数据存储
M3-不安全的通信
M4-不安全的身份验证
M5-加密不足
M6-不安全的授权
M7-客户端代码质量问题
M8-代码篡改
M9-逆向工程
M10-无关的功能

OWASP TOP 10	
A1 - 注入	A2 - 失效的身份认证和会话管理
A3 - 跨站脚本（ XSS ）	A4 - 不安全的直接对象引用
A5 - 安全配置错误	A6 - 敏感信息泄漏
A7 - 功能级访问控制缺失	A8 - 跨站请求伪造（ CSRF ）
A9 - 使用含有已知漏洞的组件	A10 - 未验证的重定向和转发

安全风险导致问题

IoT全球DDOS攻击案例

100%的IoT设备接受123456这样的弱密码

100%的IoT设备没有闭锁机制

100%的IoT设备有枚举风险

70%的IoT设备的SSH通道有root帐号权限

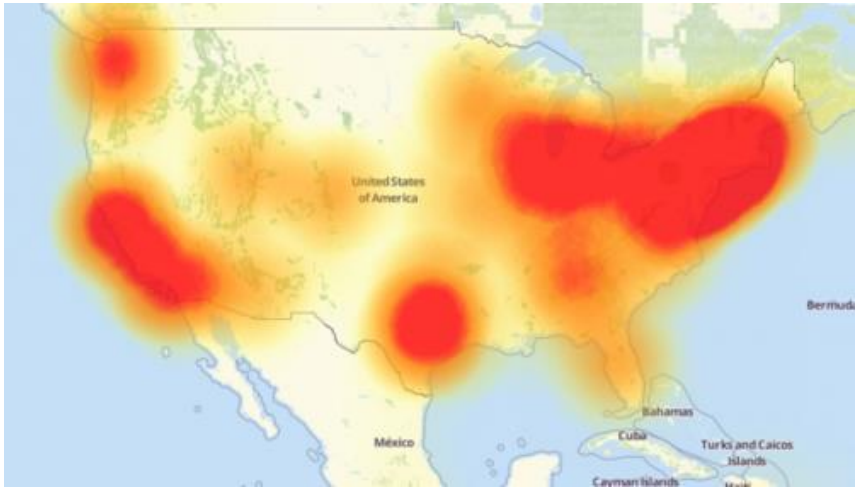
60%的IoT设备的web页面有XSS和SQL injection问题

70%的IoT设备没有加密机制

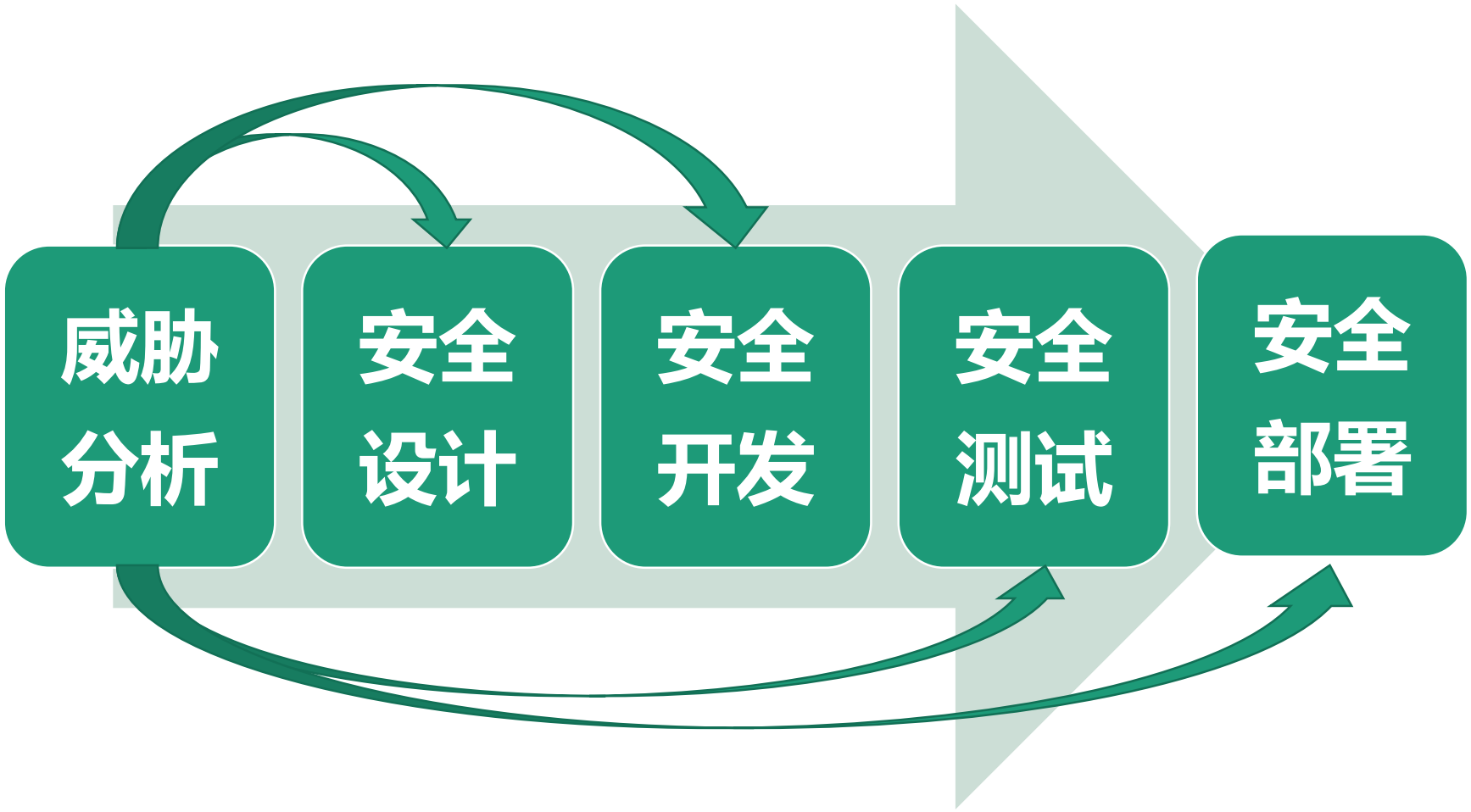
80%的IoT设备收集用户个人信息

90%的IoT设备没有多重认证机制

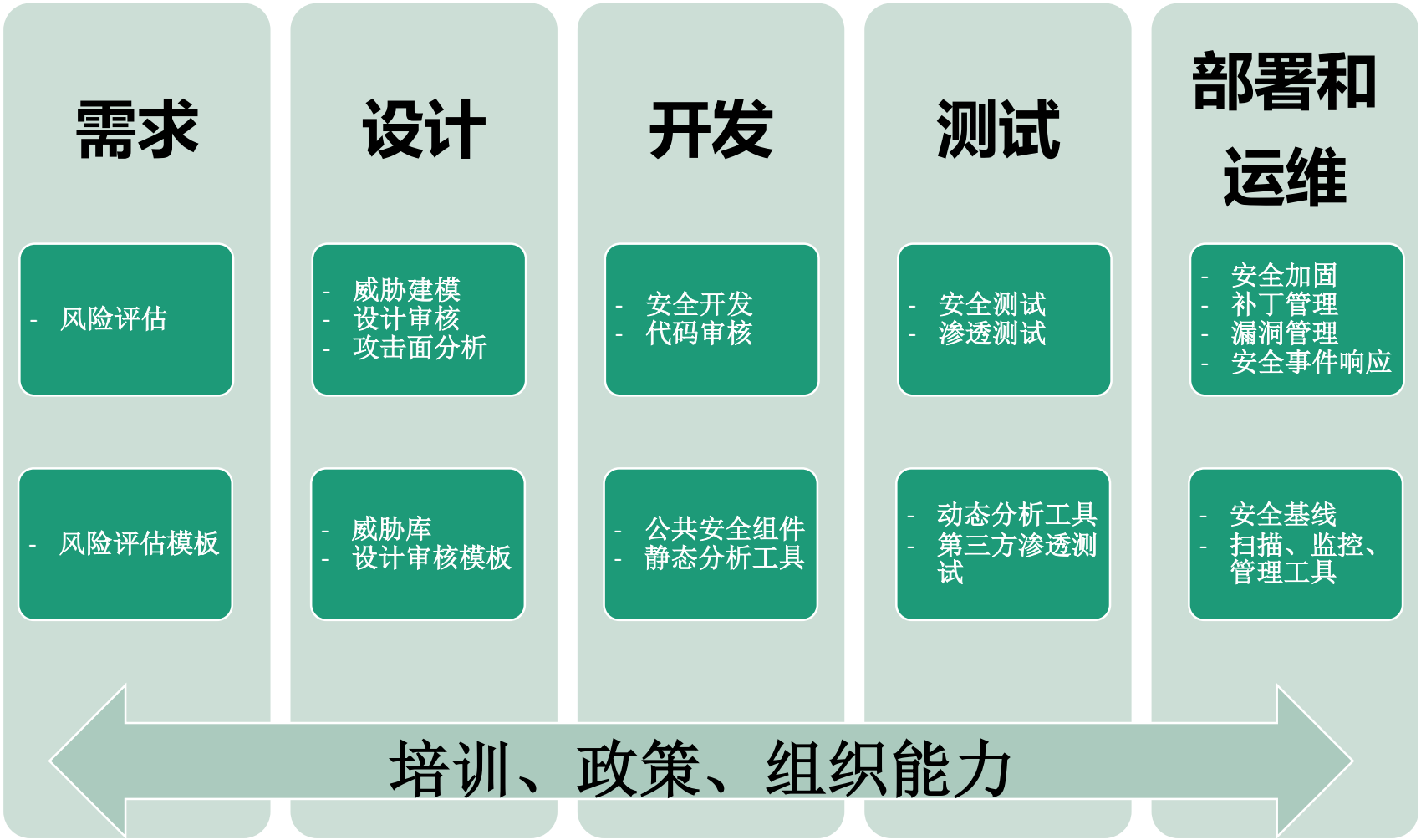
90%的IoT设备的软件升级过程有安全漏洞



安全风险削减过程 Security Risk Reduction

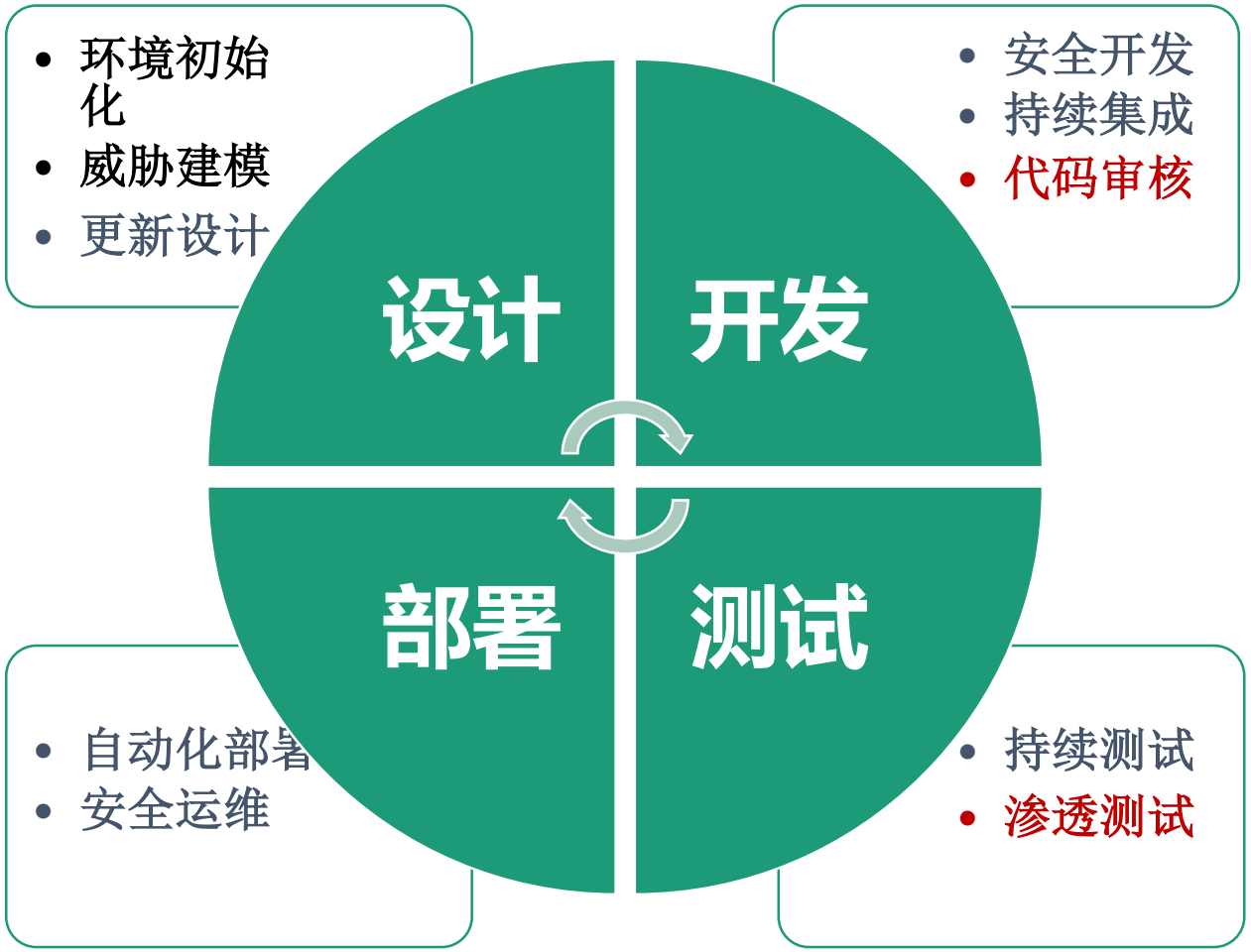


建立安全开发体系 S-SDLC



敏捷开发模式

S-SDLC



THANKS

www.seczone.cn

4000-983-183