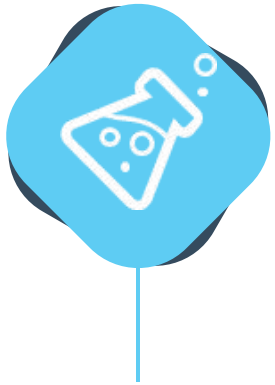
网络安全创新大会
Cyber Security Innovation Summit

DevSecOps敏捷安全技术落地实践探索

子芽　悬镜安全

软件工程安全的两个共识



系统一定有未被发现

的安全漏洞

程序员每写**1000行代码**，就会出现2~40个逻辑性缺陷。每个逻辑性的缺陷，或者若干个逻辑性缺陷，最终导致一个漏洞；“缺陷是天生的，漏洞是必然的”。

现代应用都是组装的

而非纯自研

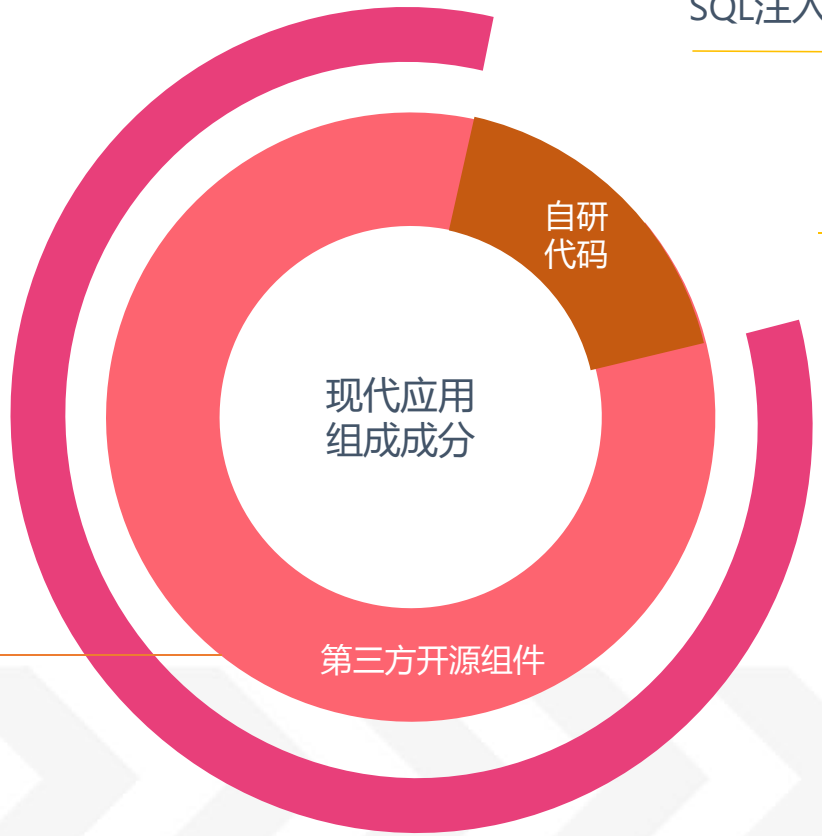
78%-90%的现代应用融入了开源组件，平均每个应用包含147个开源组件，且**67%**的应用采用了带有已知漏洞的开源组件，软件供应链安全威胁迫在眉睫。

现代应用软件的安全风险面

针对现代应用全面风险审查应考虑从第三方开源组件、自研代码通用漏洞、自研代码业务逻辑漏洞等维度综合审计。

Web通用漏洞

SQL注入、命令执行、XXE、XSS等OWASP TOP10



业务逻辑漏洞

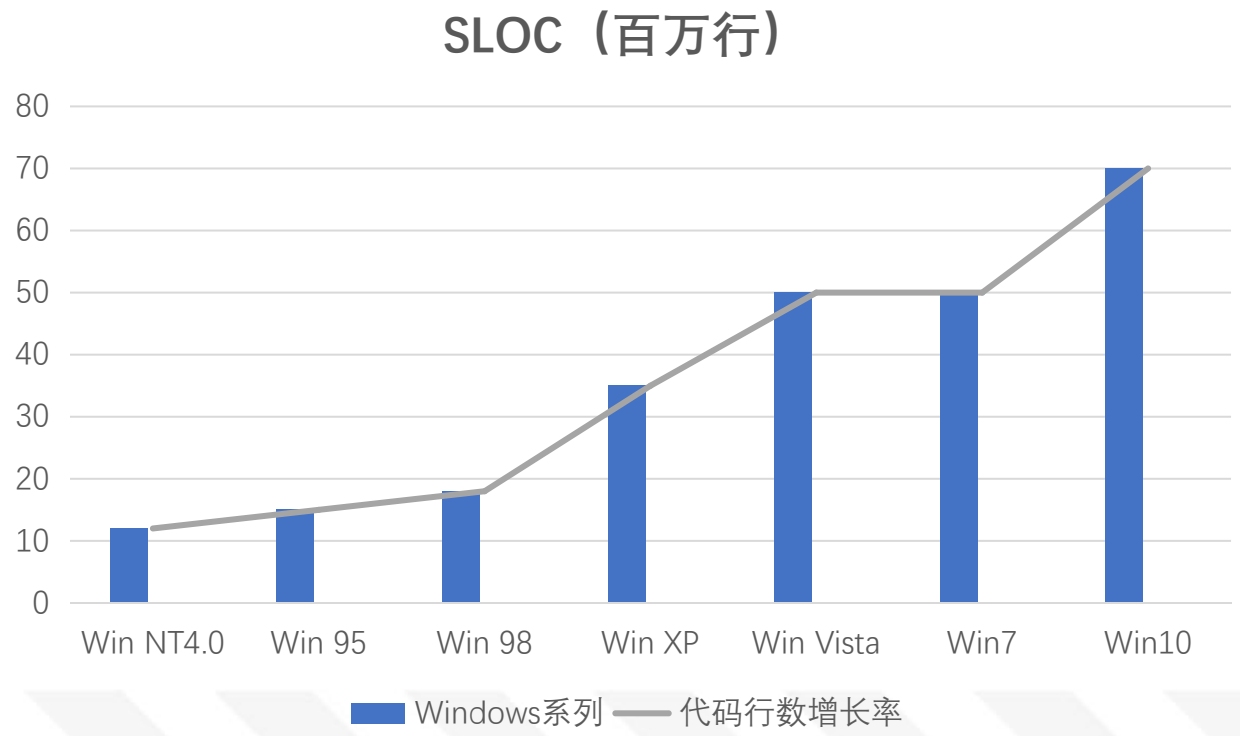
水平/垂直越权、短信轰炸、批量注册、验证码绕过等

开源组件缺陷

CNNVD、CNVD、CVE等

应用软件安全问题产生-内因

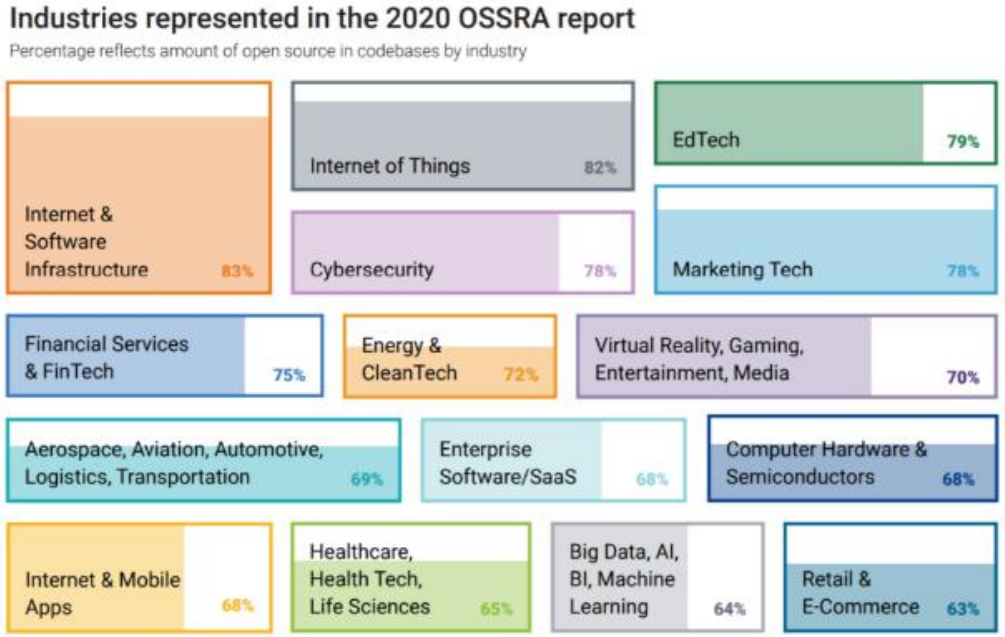
- 软件规模持续扩大，功能越来越多,越来越复杂
- 软件模块复用，导致安全漏洞延续
- 软件扩展模块带来的安全问题



Windows操作系统不同版本源代码数量

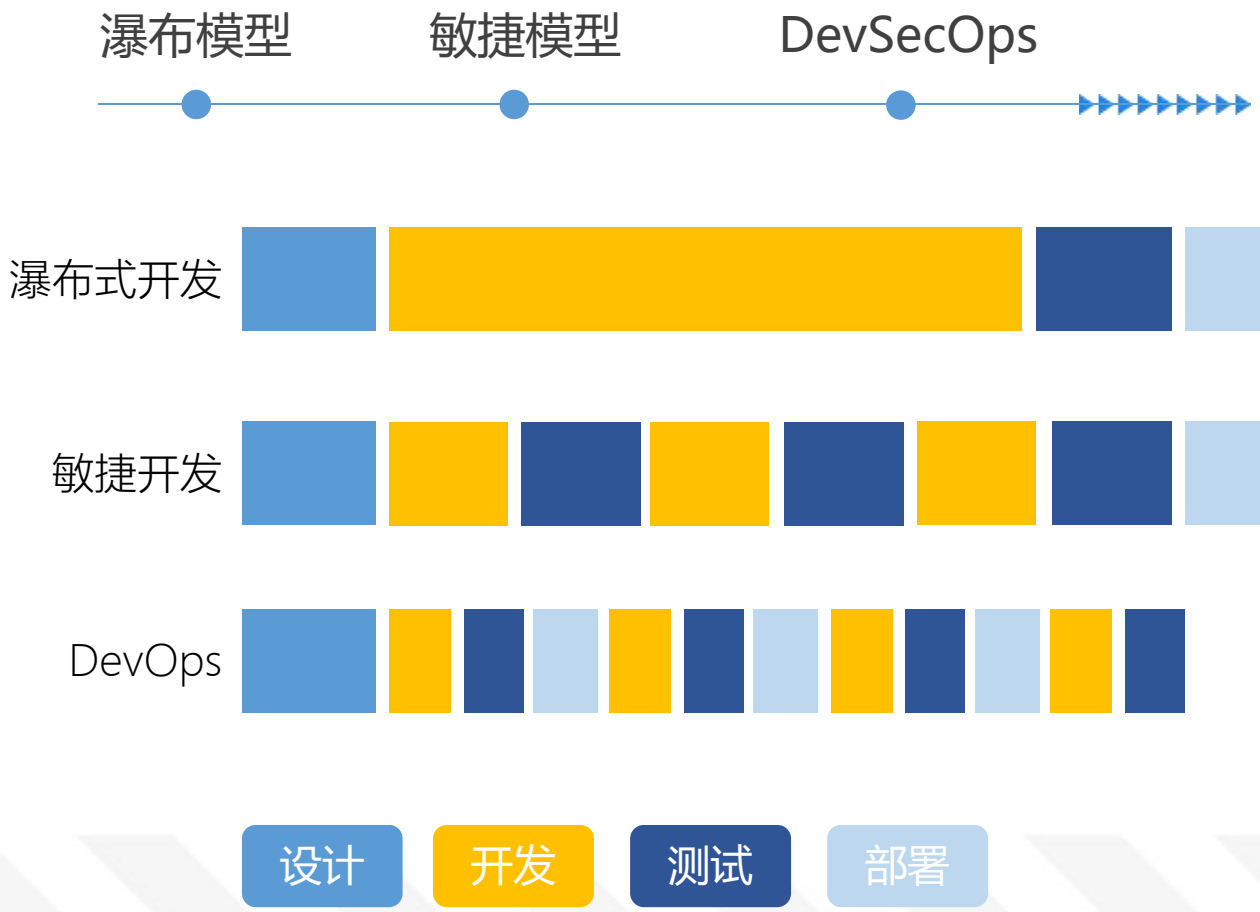
应用软件安全问题产生-外因

- **开源软件的盛行对应用安全的挑战**
- **开发环境和开发人员对软件安全的挑战**
  - 开发者缺乏安全开发的动机
    - 市场和业务要求将交付期和软件功能做主要因素
    - 用户方较少提供安全方面的压力
  - 开发者缺乏相关认知及知识
    - 软件复杂性加大，开发者需要学习更多东西
    - 传统软件开发缺乏针对性安全教育
  - 缺乏有效的安全开发工具
    - 缺乏安全开发配套管理、有效安全测试工具等



引用第三方《2020年开源安全和风险分析报告》

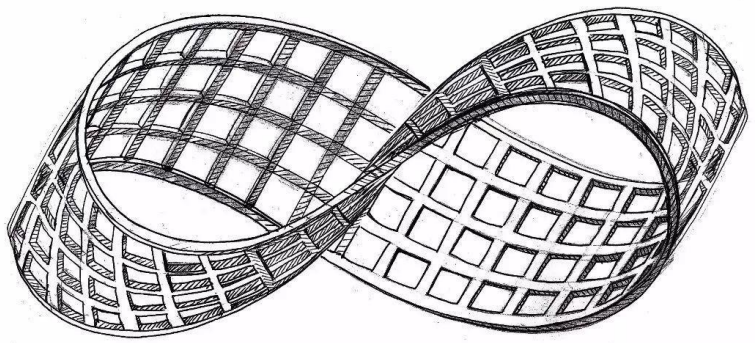
应用开发模式的技术演进



传统SDL面临的安全挑战

- 业务先上线，安全问题后补救
- 安全责任过于依赖有限安全团队资源
- 安全较缓慢，常置于流程之外，当版本更新快时，传统安全手段影响业务交付

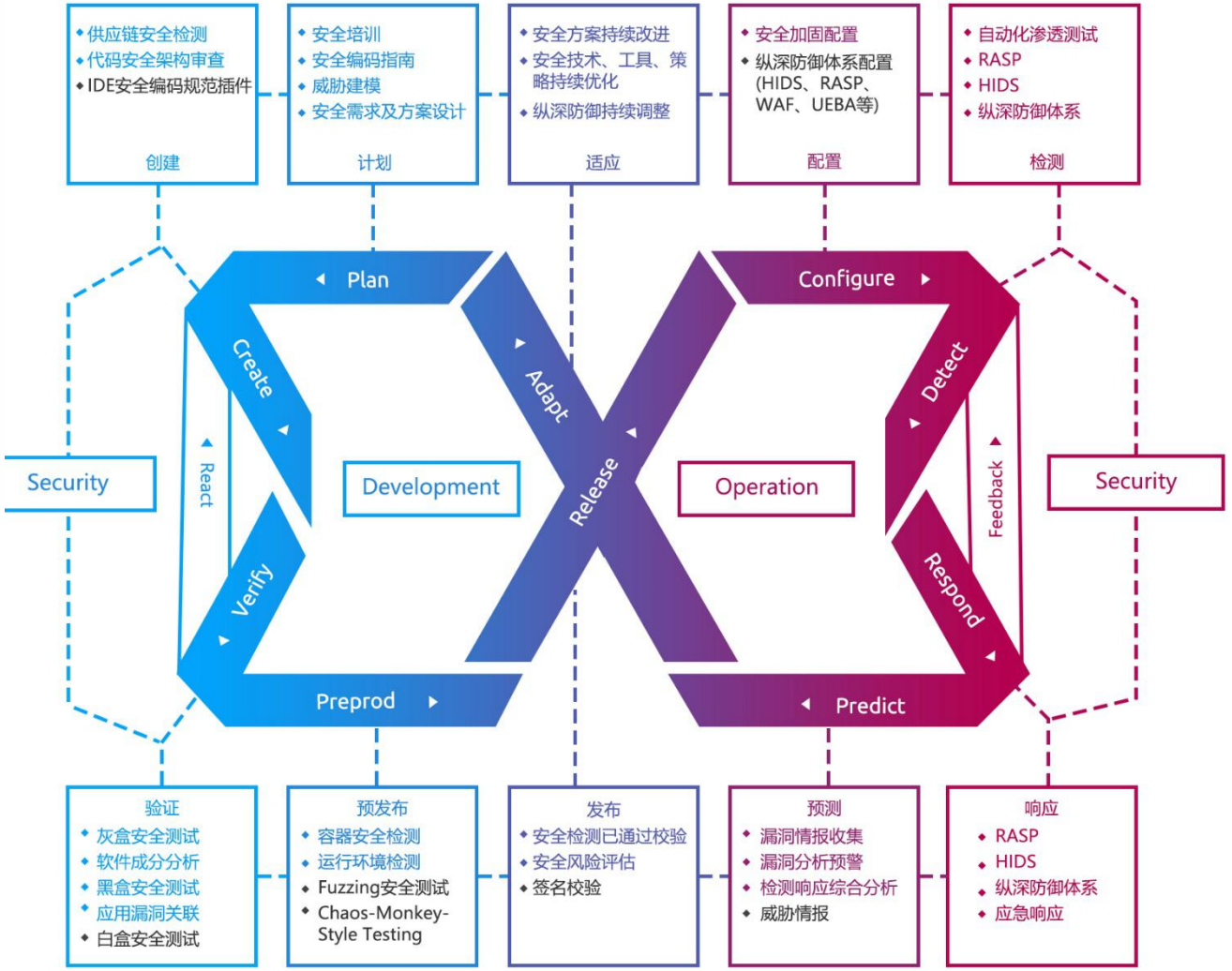
DevSecOps敏捷安全框架



莫比乌斯带，寓意无穷大“∞”，也象征着融合及亘古永恒。

DevSecOps新特性

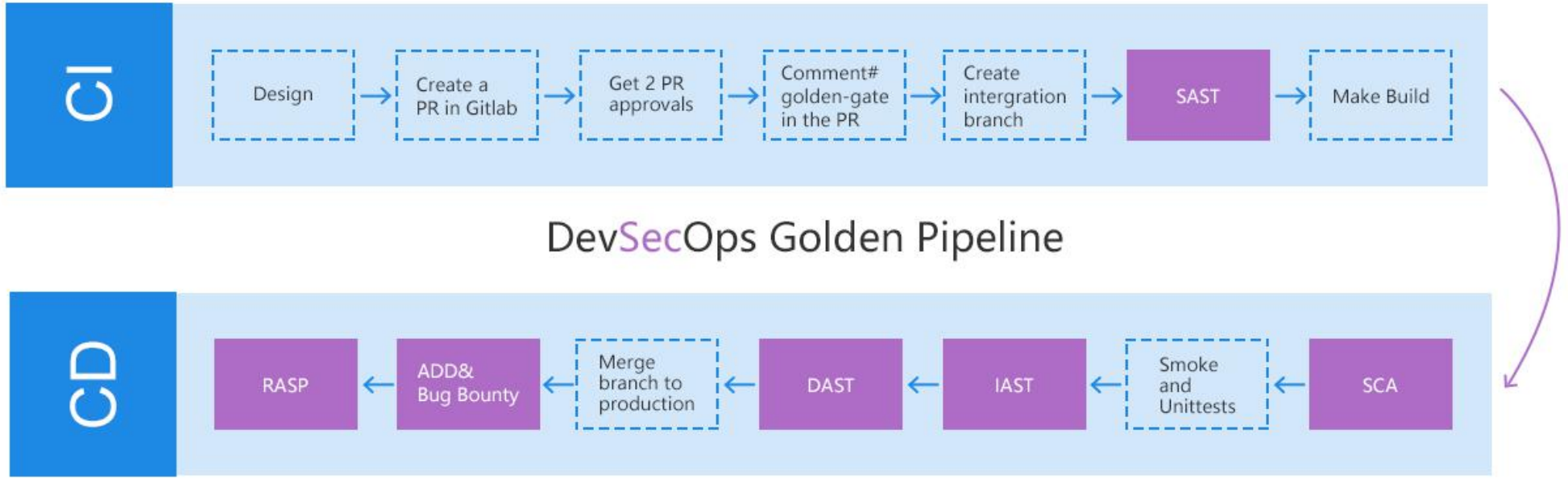
- 安全是每个人的责任
- 柔和嵌入研发运维流程，持续循环改进
- 自动化流程，人更趋向于运营反馈处理
- 适用于周期较短，迭代较快的业务



DevSecOps CI/CD黄金管道

RSAC2018正式提出“Golden Pipeline”软件流水线实践体系，强调 CI/CD 自动化工具链支撑

CI/CD黄金管道：



关键敏捷工具链技术：

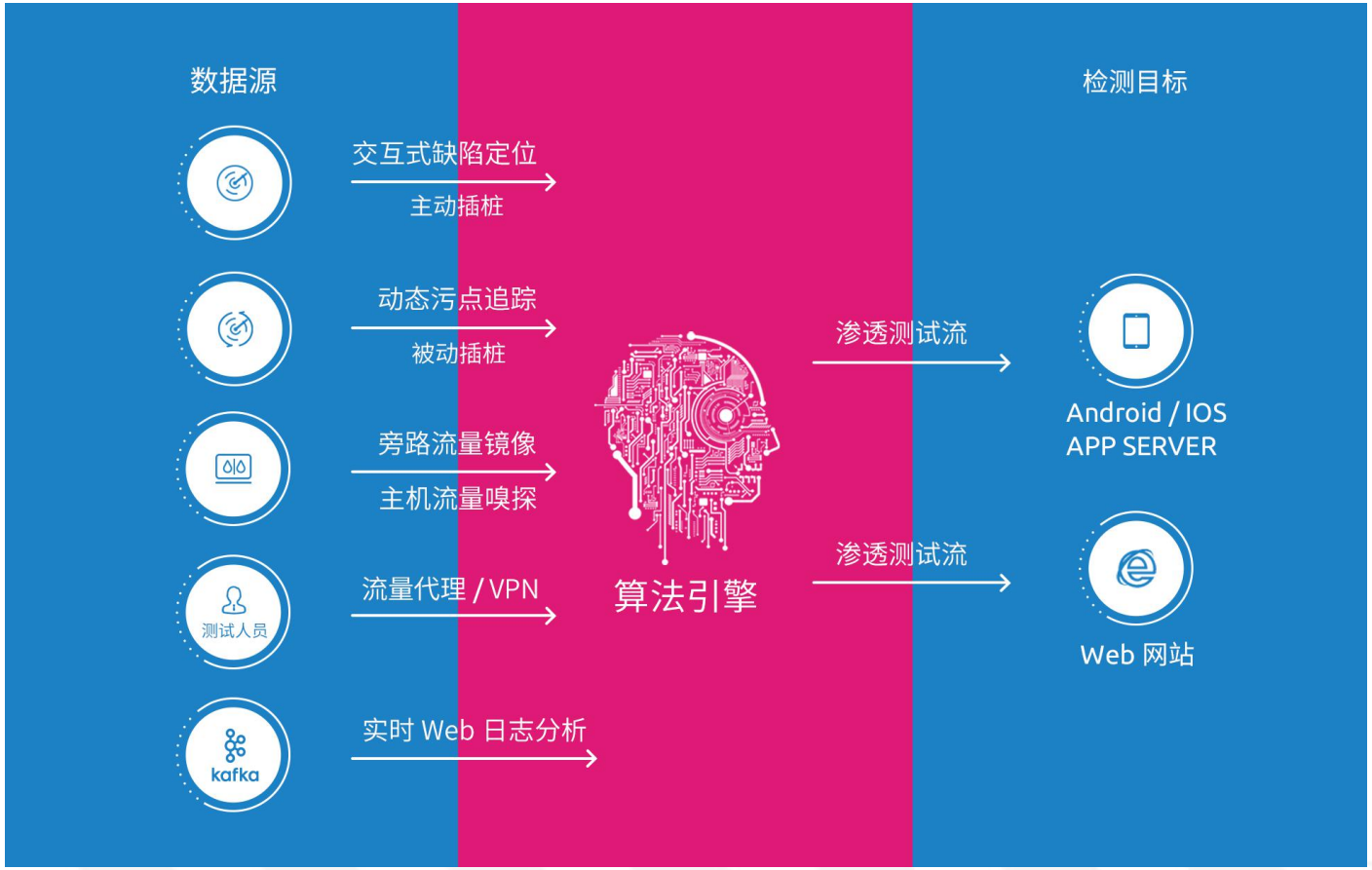
1. AST应用安全测试(灰、黑、白)
2. SCA第三方组件成分分析
3. RASP运行时应用自保护
4. 红蓝对抗和SRC众测

DevSecOps工具链-AST技术优劣分析对比

AST技术系	SAST白盒	IAST灰盒	DAST黑盒
误报率	高	极低	低
检出率	高	高	中
检测速度	随代码量	依点击流量实时检测	随URL、payload数量
第三方组件漏洞	静态扫描支持	运行时支持	依赖payload、指纹
语言支持	区分不同语言	区分不同的语言	不区分语言
框架支持	一定程度区分	一定程度区分	不区分框架
漏洞验证利用	很难验证利用	可验证利用	可验证利用
使用风险	无	无	脏数据、大流量
使用成本	高，人工排查误报	低，基本没有误报	较低
漏洞详情	代码行数、执行流	请求响应、代码行数、数据流、调用堆栈等	参数、请求响应
DevOps CI/CD支持	较高	高	低
漏洞种类覆盖	更偏向应用代码漏洞	更偏向应用本身漏洞，难以回显带外也可发现	可发现配置、运维、运行时层面漏洞

任何一项新兴技术出现，都有时代的背景和其适用的场景。

DevSecOps工具链-IAST技术实践



IAST主要关键技术

狭义的IAST特指运行时插桩模式，可帮助普通研发测试人员快速完成业务安全测试，精准定位漏洞细节及修复指导。

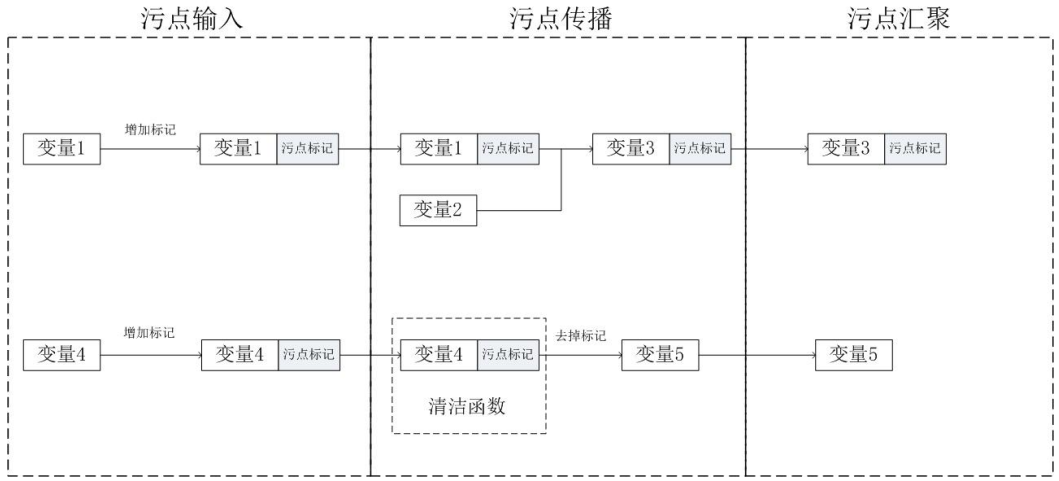
广义的IAST须包含流量学习和日志分析模式，对研发测试等使用人员完成透明，无流程侵入，不依赖应用编程语言

注意：IAST比较依赖用户点击流量覆盖的全面性，可通过主动模拟点击技术做补充。

DevSecOps工具链-被动IAST技术

被动IAST(动态污点追踪)

基于应用插桩探针， Web应用运行过程中，通过实时监控程序的污点数据在系统程序中的传播来检测数据能否从污点源传播到污点汇聚点。污点追踪主要包括：污点输入(Source)、污点传播(Propagation)、污点清洁(Sanitizer)、污点汇聚(Sink)。

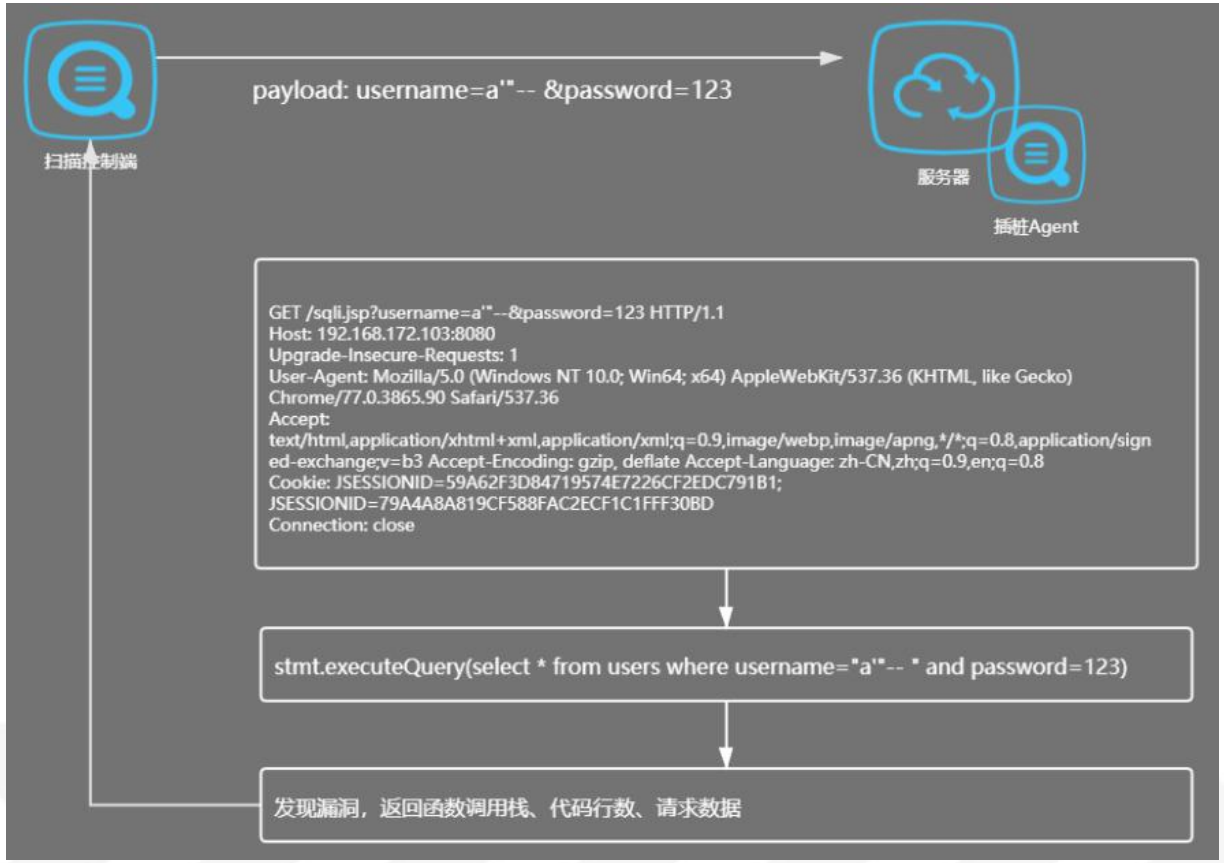


- 误报及漏报高于主动IAST
- 无数据重放、无脏数据
- 可处理签名、加密接口

DevSecOps工具链-主动IAST技术

主动IAST(交互式缺陷定位)

基于应用插桩探针， Web应用运行过程中，通过精心构造重放流量的payload来主动触发潜藏在业务应用里的安全漏洞，并在应用执行的关键函数点进行敏感操作判断和深度的脏数据处理，在发现潜藏漏洞的同时精准定位其所在代码行。

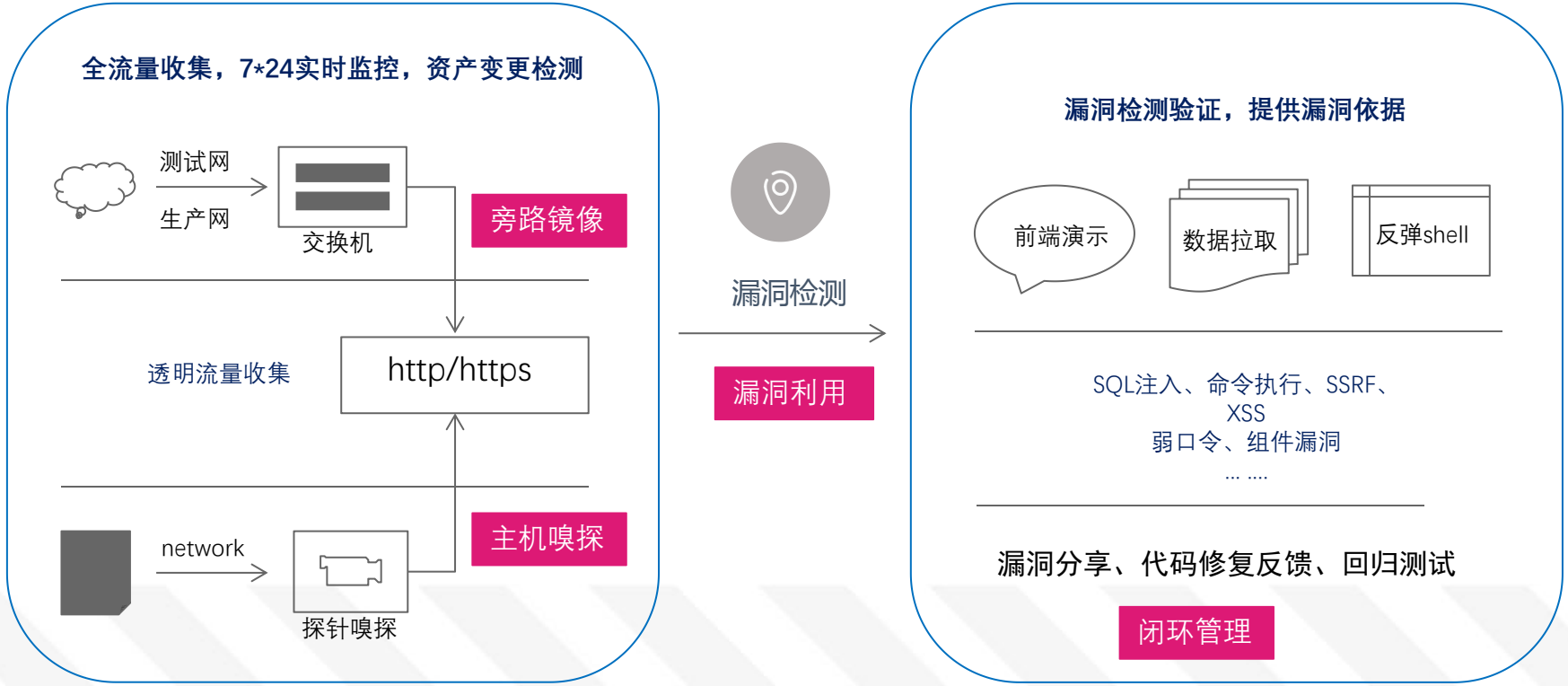


- 扫描端发送payload
- 插桩Agent在关键函数，获取上下文信息综合分析
  1）HTTP/HTTPS请求/响应
  2）URL是否触发相关插桩点
  3）payload是否进入了执行流程

- 精度更高，更易于指导研发修复
- 支持漏洞利用、漏洞复现
- 无法处理签名加密接口

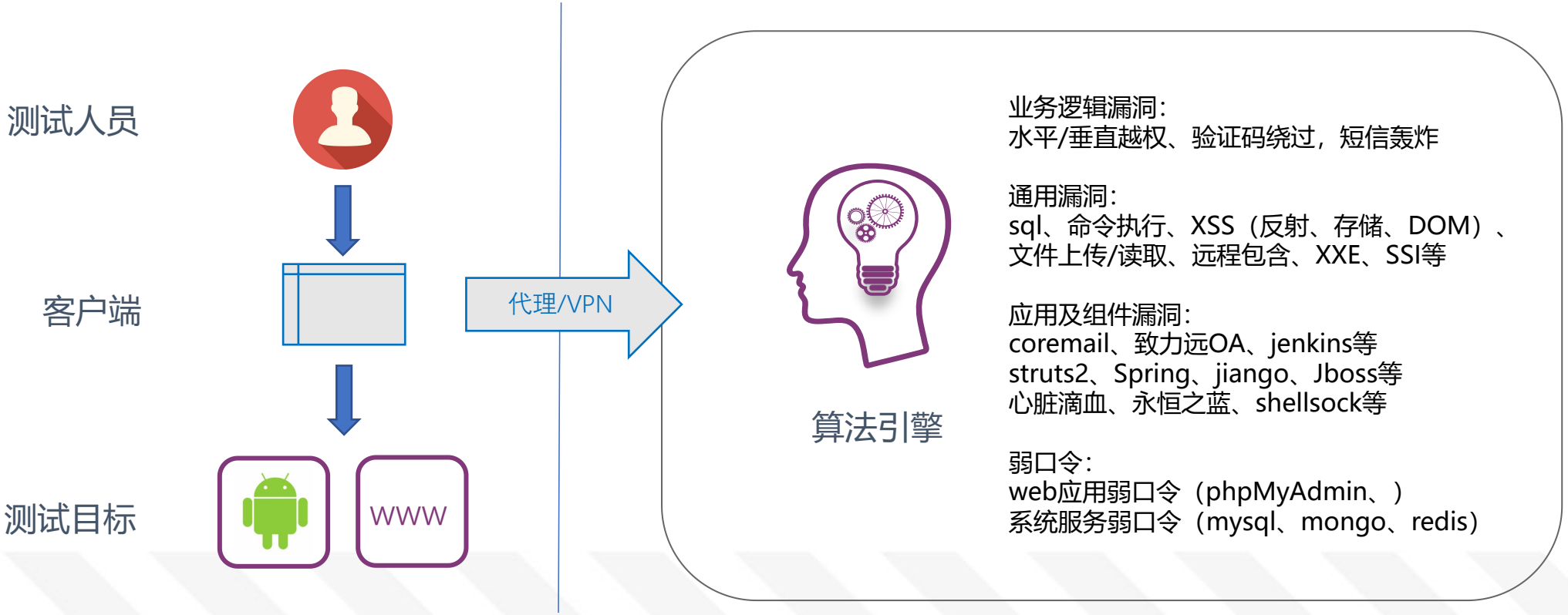
DevSecOps工具链-IAST流量镜像及嗅探技术

适合测试目标或部门之间存在隔离，无法清晰了解测试资产，通过在网络出口旁路部署流量镜像或目标主机上预置流量嗅探服务，透明无感知接入。



DevSecOps工具链-IAST用户端流量代理技术

适合测试人员对小规模网站进行深度渗透测试，不依赖测试系统语言环境，无须介入测试环境即可进行安全测试。



DevSecOps工具链-IAST运行时OSS开源组件分析

运行时OSS分析，更加侧重应用系统实际运行过程中动态加载的第三方组件及依赖；在此基础上实时检测组件中潜藏的各类安全漏洞及开源协议风险。

DevSecOps工具链-IAST技术的应用场景

应用类型	WEB应用	微服务接口	后台任务/服务	
协议类型	HTTP协议	开源RPC协议	开源框架	自有协议
关键技术	动态污点追踪	交互式缺陷定位	纵深流量学习	实时日志分析
场景适配	多检测模式	多语言支持	多平台对接	全流程闭环
风险覆盖	Web通用漏洞	业务逻辑缺陷	OSS组件缺陷	服务弱口令/未授权
部署环境	容器	虚拟机	物理机	

DevSecOps工具链-IAST技术总结

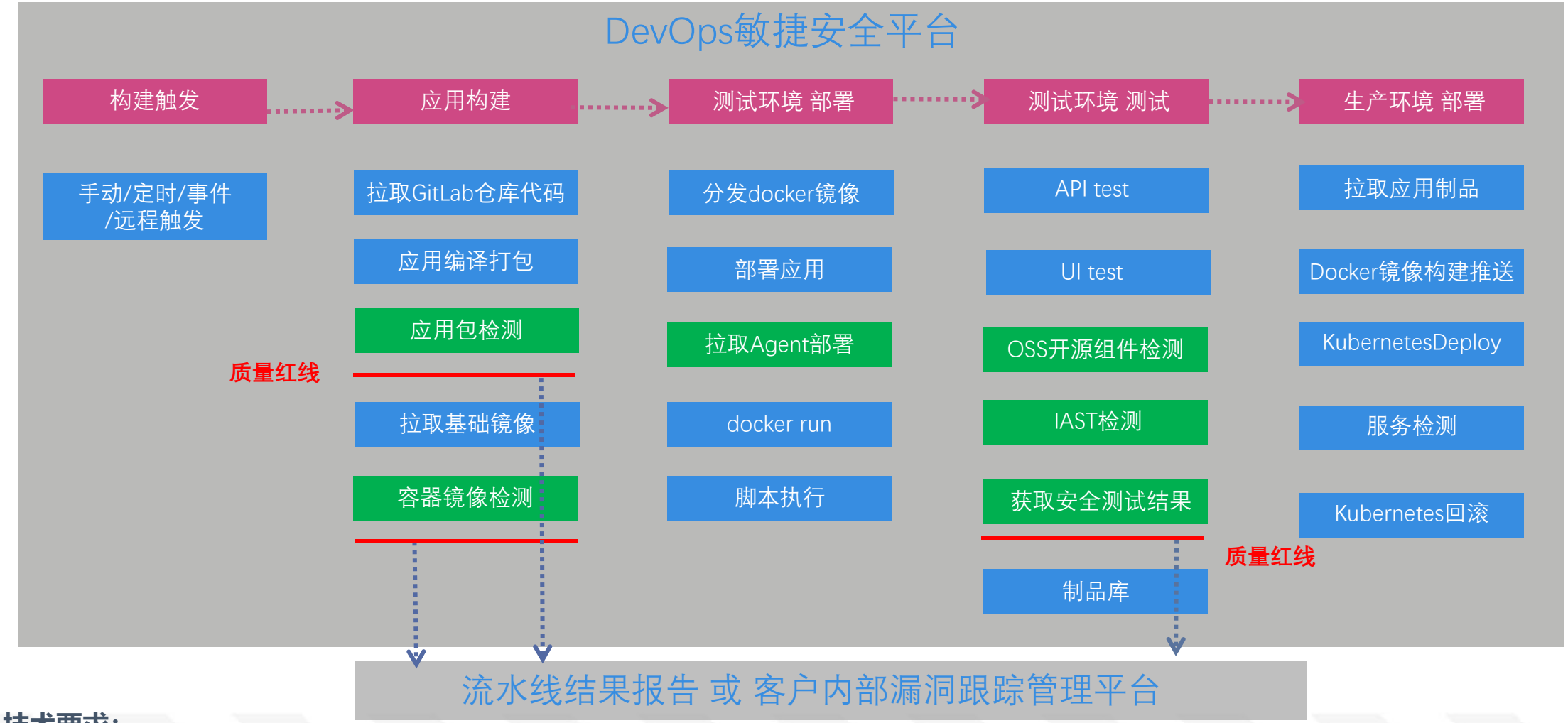
IAST技术特点

- 检测范围可覆盖黑盒OWASP TOP10及白盒CWE TOP25主要漏洞及缺陷；
- 相比传统SAST/DAST，IAST精准度更高；
- IAST可以解决签名接口问题；
- IAST获取信息全面和精细，有利于指导研发；
- IAST更易于整合到DevSecOps CI/CD流程；
- 主被动IAST融合，将使IAST技术优势更加突出。

落地重点考虑事项

- 同时支持应用插桩和多种流量追踪技术，应对业务场景丰富、开发语言众多、部署环境复杂等场景；
- 支持自动化安装，协调CI/CD完成批量部署；
- 支持配置热加载、性能异常熔断机制；
- 支持SCA第三方开源组件运行时监测分析；
- 支持Jenkins 、Jira、CAS等第三方平台；
- 支持全流程闭环漏洞管理，包括从漏洞发现、问题沟通、修复整改、漏洞复查及趋势分析等维度。

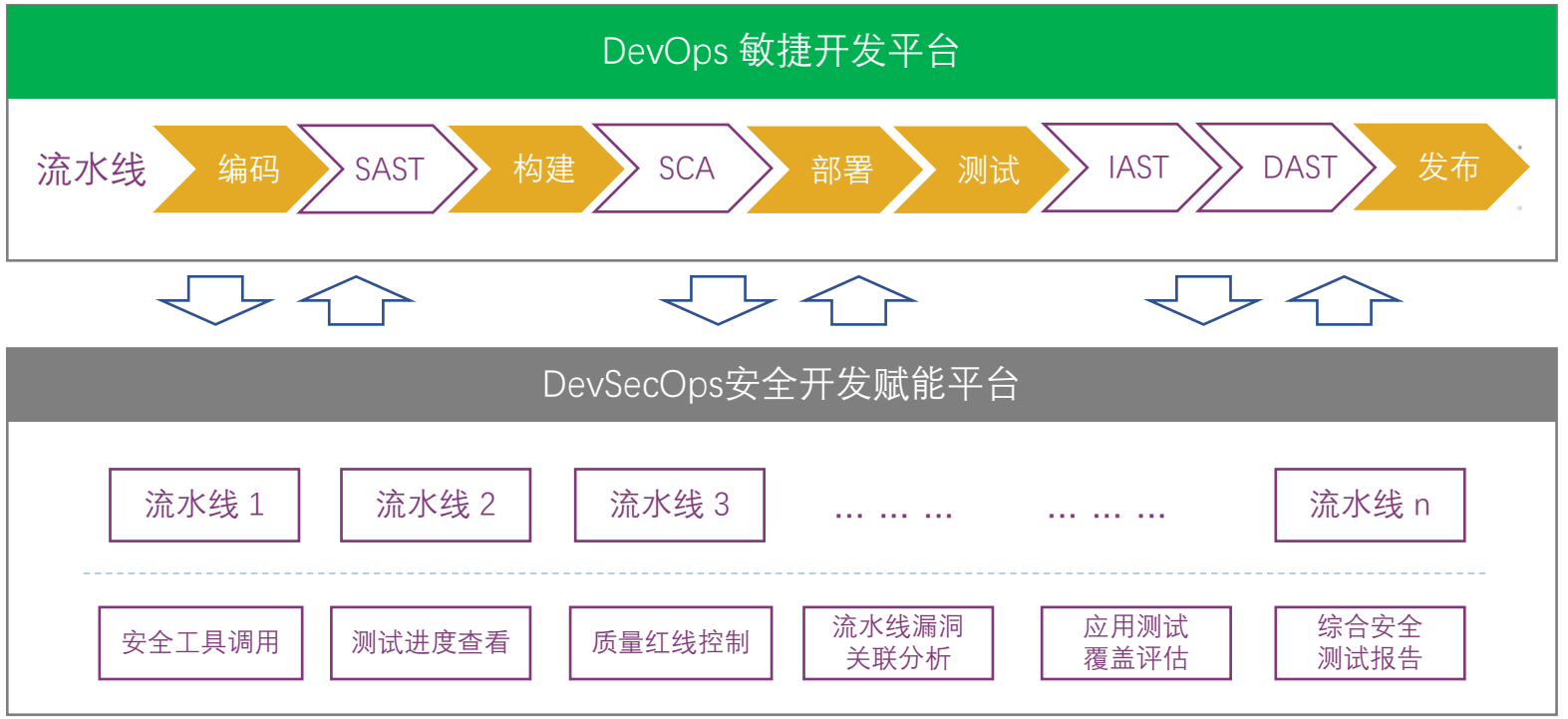
某证券机构DevSecOps（工具链）案例



技术要求：

1）整个流程十几分钟，测试用例覆盖度可度量(类数量、请求数量)；
2）质量红线设定很关键（漏洞等级、对应CVE是否有POC、应用内外网属性等）；
3）要求批量自动化安装部署，支持配置热加载，支持定制启动入口，对接DevOps平台；
4）支持自研私有协议和开发框架，对接流水线项目及子系统信息，便于追踪定位。

某银行机构DevSecOps（平台+工具链）案例



- 安全与研发流程融合

安全活动	DevOps质量门禁	DevOps数据接口	度量指标
标准化发放、 源代码检测、 开源组件检测、 IAST灰盒测试、 DAST黑盒测试	只从制品库获取、 源代码检测漏洞情况、 开源组件检测高危组件情况、 IAST灰盒测试漏洞情况、 DAST黑盒测试漏洞情况	获取制品信息、 获取源代码检测漏洞情况、 获取开源组件检测高危情况、 获取IAST灰盒测试漏洞情况、 获取DAST黑盒测试漏洞情况	制品全自动发放比例、 制品安全加固比例、 OSS组件扫描覆盖率、 OSS组件风险指标、 IAST检测覆盖率、 IAST漏洞检测数量、 DAST检测覆盖率、 DAST漏洞检测数量…

参考BSIMM、综合评估软件及团队安全成熟度、持续发现调整改进方向

- 项目风险数据收集
- 工具检测数据收集

- 非重度度量指标
- 指标覆盖各风险类型
- 指标覆盖各风险阶段

悬镜DevSecOps落地解决方案

悬镜DevSecOps智适应威胁管理体系

安全开发

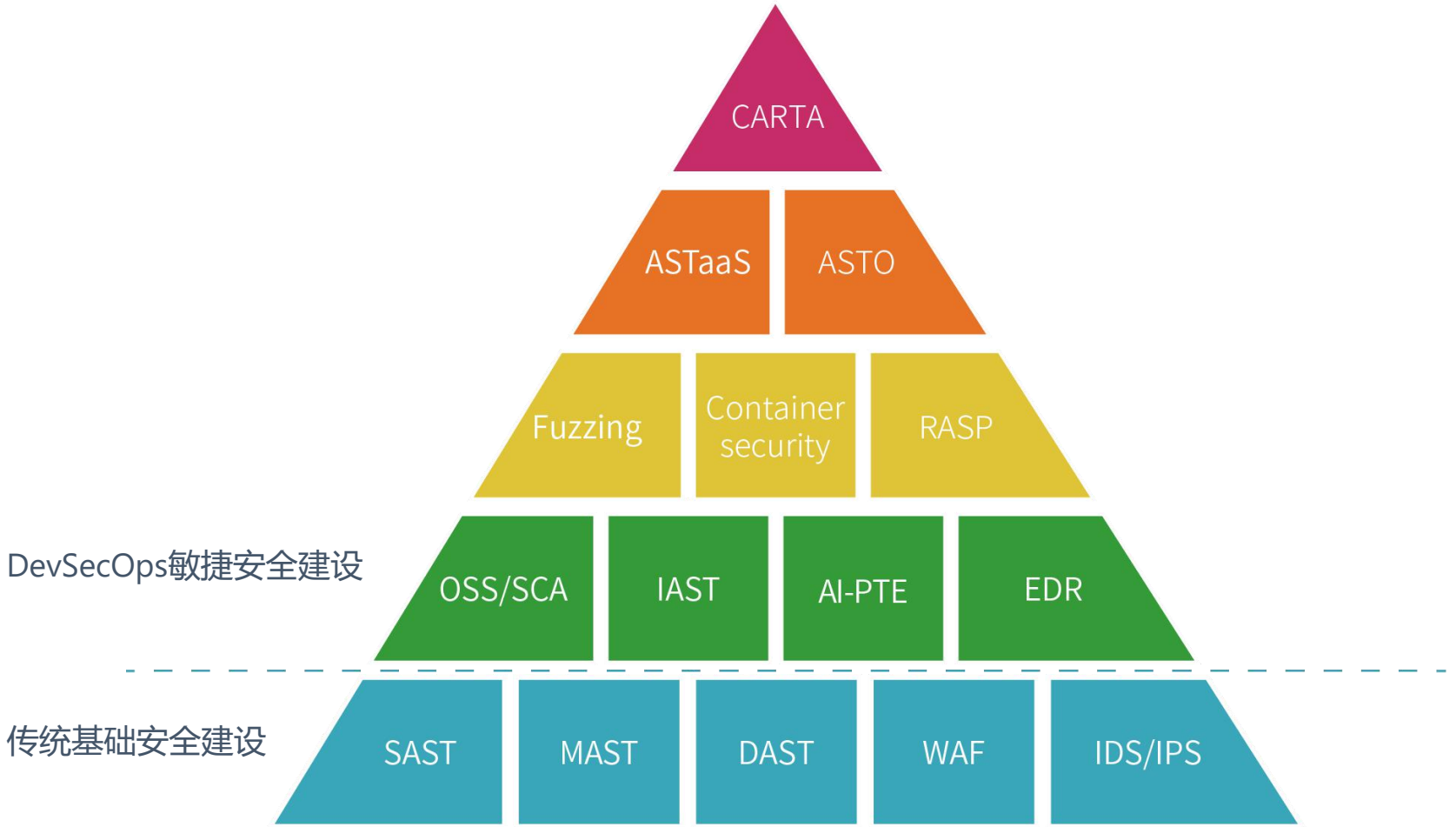
- SDL安全咨询
- 安全开发实训
- 安全编码指南
- 源代码审计

安全运营

- 渗透测试
- 攻防演练
- 风险评估
- 等保咨询

DevSecOps敏捷安全技术未来演进方向

网络安全创新大会
Cyber Security Innovation Summit



DevSecOps敏捷安全技术实践体系

DevSecOps建设初心：

1）安全是一门平衡艺术

- 本质是风险和信任的平衡
- 拥抱变化是安全建设的基石

2）人是安全的基本尺度

- 从源头做威胁治理
- 内生安全

3）从左移到无处不移

网络安全创新大会
Cyber Security Innovation Summit

THANKS