API设计与实现

主讲人：陈长兵

目 录

CONTENTS

5 API安全

- API安全概述
- 安全设计原则
- API安全技术
- API安全治理

5.1 API安全概述

什么是API安全

从安全的角度关注API领域的安全问题和这些问题的解决方案，从技术和管理两个层面提高API自身和API周边生态的安全性。

5.1 API安全概述

API安全内容

5.1 API安全概述

安全问题主要成因

- 企业API安全意识不足
- 技术革新导致API安全风险增加
- API自身安全机制不足

5.1 API安全概述

安全面临的主要挑战

- API广泛使用带来攻击面的扩大
- API安全实践经验缺失
- 外部环境变化带来的合规性挑战

5.1 API安全概述

常见API安全漏洞类型

- 未受保护的API
- 弱身份鉴别
- 中间人劫持
- 传统Web攻击
- 弱会话控制
- 反向控制
- 框架攻击

5.1 API安全概述

OWASP API安全漏洞

- OWASP
  - Open Web Application Security Project
  - 开放式Web应用程序安全项目
- 是一个开源的、非盈利的全球性安全组织
- 致力于应用软件的安全研究，有很多开源项目

5.1 API安全概述

OWASP API安全漏洞

- OWASP API安全Top10
  - 延续Web安全传统，收集公开的与API安全事件有关的数据
  - 安全专家组分类，挑选十大API安全漏洞类型
  - https://owasp.org/www-project-api-security/
- API1-失效的对象级授权
- API2-失效的用户认证

5.1 API安全概述

OWASP API安全漏洞

- API3-过度的数据暴露
- API4-缺乏资源和速率控制
- API5-失效的功能级授权
- API6-批量分配

5.1 API安全概述

OWASP API安全漏洞

- API7-安全性配置错误
- API8-注入
- API9-资产管理不当
- API10-日志记录和监控不足

5.2安全设计原则-5A原则

5A原则

- 在安全设计时，从这5个方面综合考量安全设计的合理性。
  - Authentication身份认证
  - Authorization授权
  - AccessControl访问控制
  - Auditable可审计性
  - AssetProtection资产保护

5.2安全设计原则-5A原则

5A原则

- Authentication身份认证
  - 知识谁在与API服务进行通信
  - 是否是API服务允许的客户端请求
  - 类似普通Web应用程序，提供注册、登录功能

5.2安全设计原则-5A原则

5A原则

- Authorization授权
  - 描述“你能访问什么”
  - 通过身份认证后，访问者被授予可以访问哪些API
  - 赋予某个客户端调用权限的过程，通常为授权操作的过程

5.2安全设计原则-5A原则

5A原则

- AccessControl访问控制
  - 是对授权后的客户端访问时的正确性验证
  - 与授权的差异，一个配置，一个行为

5.2安全设计原则-5A原则

5A原则

- Auditable可审计性
  - 业务功能 & 可审计功能
  - 记录接口调用的关键信息，以便通过审计手段及时发现问题，并在发生问题时通过审计日志进行溯源，找到问题的发生点

5.2安全设计原则-5A原则

5A原则

- AssetProtection资产保护
  - 主要是指API接口自身的保护
  - 限速限流，防止恶意调用
  - 传输的个人信息，隐私数据，信息资产保护

5.2安全设计原则-纵深防御原则

纵深防御

- 军事领域
  - 前方到后方之间，构建多道防线，达到整体防御的目的
- 网络安全领域
  - 指不能只依赖单一安全机制，建立多种安全机制，互相支撑以达到相对安全的目的
- API安全设计

在不同层面使用不同的安全技术，来达到纵深防御的目的

5.2安全设计原则-纵深防御原则

纵深防御

- API业务属性划分
  - 公共型API、私有型API
- 再根据业务需求、粗细粒度划分
  - 采用不同的身份认证和授权技术实现

5.2安全设计原则-纵深防御原则

纵深防御

- 举例：网银转账接口
  - 登录时的身份认证
  - 转账时的身份认证

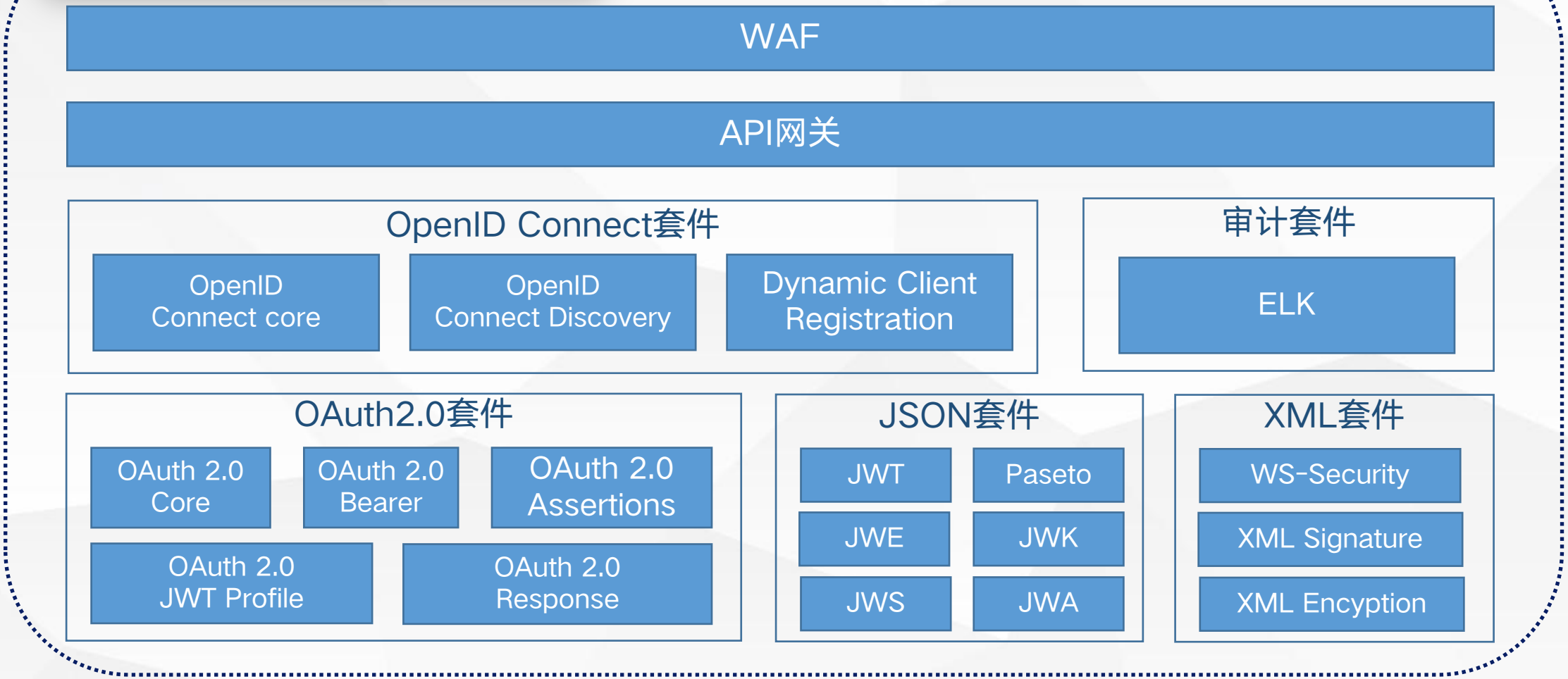
5.2安全设计原则

5A与纵深防御

- 5A原则
  - 重点强调每一层安全架构设计的合理性
  - 是横向的安全防护，强调宽度
- 纵深防御
  - 是对同一问题从不同的层次、不同的角度做安全防护
  - 是纵向的安全防护，强调深度
- 两者构成有机的防护整体

5.3API安全技术

API安全技术栈



5.3API安全技术

API安全技术栈

- 最上层WAF、API网关
  - API安全的基础套件
  - 为API安全提供综合的安全支撑能力
- 认证与授权
  - 已OpenID Connect套件、OAuth2.0套件为代表
  - 提供API的身份证和鉴权解决方案
- 审计套件、JSON套件、XML套件
  - 为API消息保护和安全审计提供技术支持

5.3API安全技术-身份认证技术

认证方式

- 基于用户身份的认证技术
- 基于应用程序身份的认证技术

5.3API安全技术-身份认证技术

基于用户身份的认证技术

- 常用认证方式
  - 用户名/密码认证
  - 动态口令
  - 数字证书认证
  - 生物特征认证

5.3API安全技术-身份认证技术

基于用户身份的认证技术

- 多因子认证
  - 用户名/密码 + 短信验证码
  - 用户名/密码 + 动态令牌
  - 用户密码/密码 + 人脸识别
  - 人脸识别 + 短信验证码

5.3API安全技术-身份认证技术

基于用户身份的认证技术

- 举例：CAS单点登录流程

5.3API安全技术-身份认证技术

基于应用程序身份的认证技术

- 常用认证方式
  - HTTP Basic认证
  - Token认证
  - 数字证书认证

5.3API安全技术-身份认证技术

基于应用程序身份的认证技术

- OpenID Connect标准规范
  - 将身份认证融入授权码、简易授权码、客户端凭据等授权流程
  - 身份认证 + OAuth2.0

5.3API安全技术-授权与访问控制技术

授权方式

- 基于使用者身份代理的授权与访问控制技术
  - 以OAuth2.0协议为代表
  - 获取使用者的授权许可
  - 使用者：某个自然人用户，或者某个客户端应用程序
- 基于使用者角色的授权与访问控制技术
  - 以RBAC模型为代表
  - 授予角色，角色包含资源的访问权限

5.3API安全技术-授权与访问控制技术

OAuth协议

- 目前最流行的客户端应用授权机制
  - 解决API在多个应用程序之间调用时的授权问题
- 基本思路
  - 采用授权令牌的代理机制
  - 在客户端应用程序、授权服务器、被调用API或资源之间，构建一个虚拟的令牌层
  - 用于资源访问的授权确认

5.3API安全技术-授权与访问控制技术

OAuth协议

- OAuth授权核心流程
  - 访问客户端应用程序
  - 请求授权
  - 确认授权
  - 申请令牌
  - 颁发令牌
  - 申请资源
  - 开放资源

5.3API安全技术-授权与访问控制技术

RBAC模型

- Role-Based Access Control基于角色的访问控制
- 基础是业务角色
  - 依赖于角色构建授权和访问控制能力
- 授权核心要素
  - 账号，代表用户身份或者调用应用客户端的身份
  - 权限，将系统提供的业务功能安装数据和功能维度划分数据权限和功能权限
  - 角色，是账号和权限之间的桥梁，将调用者身份与可操作的具体功能或数据进行授权关联

5.3API安全技术-授权与访问控制技术

RBAC模型

- 举例：RBAC模型，用户-角色-权限关系图

5.3API安全技术-消息保护技术

保护机制

- 通信链路保护
  - 在传输层保护，使用TLS/SSL来提高通信链路的安全性
- 应用层消息加密和签名
  - 应用层对消息体进行加密和签名
  - 加密：保护数据的机密性
  - 签名：保护数据的防劫持和防篡改

5.3API安全技术-消息保护技术

防护套件

- 应用层API交互技术的多样性
  - 对消息体的保护更多要围绕具体的交互细节去实现
  - 认证令牌的保护、对访问令牌的保护、对敏感信息的保护等
- 防护套件
  - JSON和XML作为消息传递的数据格式
  - 相关技术标准JWT、JWE、WS-Security等

5.3API安全技术-消息保护技术

加密算法

- 对称算法
  - 通信双方使用相同秘钥
  - 多方都知道秘钥，安全性大大降低
- 非对称加密算法
  - 公钥加密，私钥解密
  - 性能影响
  - 协商对称加密的秘钥，使用非对称加密算法加密消息体

5.3API安全技术-日志审计

审计目的

- 通过审计策略和日志分析，发现系统在某一时间内发生的异常事件，通过事件关联和追溯，分析与事件相关联的内外部人员、系统、事件涉及的范围等。

5.3API安全技术-日志审计

日志结构

- 时间：一般精确到毫秒
- 来源：日志操作的来源，比如源IP、源主机
- 结果：日志涉及的操作是否成功
- 操作者：由谁操作，某个用户或者客户端应用程序
- 操作详情：具体操作内容，比如给某个API授权
- 目标对象：被操作的对象，比如被请求的API端点

5.3API安全技术-日志审计

采集位置

- 接口层
  - 记录在什么时间，谁调用了哪个API端点，是否调用成功等信息
  - API网关，使用代理或切面
- 操作层
  - 记录API端点被调用执行的业务操作
  - 比如：创建某个资源，删除某个资源，查询哪些数据等
  - 与业务逻辑相关，按照日志标准格式输出

5.3API安全技术-日志审计

开源日志技术组件

- 日志采集
  - Filebeat，Logstash，Fluentd
- 日志存储
  - Elasticsearch
- 日志展示
  - Kibana

5.3API安全技术-威胁防护

当前WAF产品的不足

- 认证和授权流程的绕过
  - API，OAuth2.0和OpenID Connect
  - 认证和授权流程负责，也业务耦合度高
- 数据格式难以识别
  - API交互过程使用JSON、XML等对象实体
  - 需要深入这些格式的数据结构内容分析

5.3API安全技术-威胁防护

当前WAF产品的不足

- 流量控制能力难以满足业务需求
  - 传统检测和防护策略，难以对新型攻击起到有效的防护作用
  - API层面的CC攻击、慢BOT攻击
  - 传统策略：访问频率限制、IP黑名单设置、二次验证机制等

5.3API安全技术-威胁防护

针对API安全的防御安全

- WAF
  - 分析流量特征，拦截带有攻击特征的请求
- 开发安全机制
  - 根据不同业务需求来定制化设计
- 使用RASP防护
  - 数据已被应用层解析

5.3API安全技术-威胁防护

针对API安全的防御安全

- 限流和熔断需求
  - 通过流程控制策略和API管理来实现
- 流量控制策略，从三个层面设置流量阈值规则
  - API端点
  - 应用程序
  - 用户
- 关注策略触发优先级

5.3API安全技术-威胁防护

针对API安全的防御安全

- 具体规则，可从三个方面考虑
  - API调用频次
  - API调用时长
  - API调用总数

5.3API安全技术-威胁防护

针对API安全的防御安全

- RASP，Runtime Application Self-Protection
  - 将防护功能注入到应用程序中，通过Hook少量关键函数，来实时观测程序运行期间的内部情况
  - 当应用出现可疑行为时，RASP根据当前上下文环境精准识别攻击事件，并给予实时阻断，是应用程序具备自我防护能力，而不需要进行人工干预。

5.3API安全技术-威胁防护

针对API安全的防御安全

- RASP优势和应用场景
  - 深入代码上下文，基于行为精准识别攻击路径
  - 对应用打虚拟补丁，修复官方未修复的漏洞
  - 赋能扫描器实现灰盒检测IAST
  - 第三方组件安全风险梳理

5.4API安全治理

从API治理的角度，讨论在API生命周期中，如何综合性融合管理手段和技术手段进行API安全治理，将从四个方面展开：

- SDL
- DevSecOps
- API网关
- 数据隐私

5.4API安全治理-SDL

SDL是什么

- 最早由微软提出，围绕软件生命周期的安全管理模型
  - 2004，在软件开发各阶段引入安全和隐私问题的考量，将SDL引入其内部软件开发流程
  - 2008，发布一系列重要的指南，SDL优化模型
  - 2010，添加敏捷开发模板，综合落地过程中安全投入成本、应用安全性和易用性之间的考量，做了更易于落地实施的改进

5.4API安全治理-SDL

SDL是什么

◆ SDL关键安全活动

5.4API安全治理-SDL

API安全培训

- 开展安全培训活动考虑内容
  - 培训对象
  - 培训目标和内容
  - 培训形式
  - 培训计划
  - 效果评估

5.4API安全治理-SDL

API安全培训

- 培训对象
  - 项目经理
  - 技术负责人
  - 架构师
  - 开发工程师
  - 测试工程师
  - 运维工程师
  - 质量工程师

5.4API安全治理-SDL

API安全培训

- 培训内容
  - API安全管理框架和关键指标
  - 常见API安全问题
  - 常见API安全技术和安全设计
  - API安全编码案例
  - API安全测试与工具使用

5.4API安全治理-SDL

API安全培训

- 培训相关工具
  - API安全小贴士文档
  - OWASP API安全 Top10官方文档
  - API技术官方文档，Open API规范、OAuth协议核心规范、JWT规范等
  - Spring-Security官方文档

5.4API安全治理-SDL

API安全需求

- 开展安全需求的目的
  - 在开发早期对需求开发的API服务或接口进行安全评估
  - 识别不同类型的安全需求
  - 通过安全设计或其他消减措施来控制安全风险
  - 提高系统的安全性

5.4API安全治理-SDL

API安全需求

- 开展安全需求考虑内容
  - 过程保证类需求，组织级的安全要求
  - 监管合规率需求，监管部门的安全要求
  - 技术保障类需求，从安全攻防的角度

5.4API安全治理-SDL

API安全需求

- 过程保证类需求
  - 定义SDL的各个关键活动在什么阶段开展，输出产物
  - 组织层面已明确的操作流程、工具、关键里程碑、交付成果、验收标准等
  - 需要在管理实践中落地的内容

5.4API安全治理-SDL

API安全需求

- 监管合规类需求
  - 一般源于国家法律法规、行业规范、上级管理部门要求等
  - 比如：个人隐私保护类合规要求

5.4API安全治理-SDL

API安全需求

- 技术保障类需求
  - 系统提供API调用客户端应用程序注册功能
  - API调用监控功能
  - API流量限制功能等
  - 可以使用业界公开的威胁库或检查表作为需求分析的参考依据

5.4API安全治理-SDL

API安全需求

- 常规落地方式
  - 由安全人员牵头，组织编写安全需求检查表checklist
  - 评审通过后，由业务侧技术人员去开展安全需求工作
  - 安全人员再负责把关审核

5.4API安全治理-SDL

API安全需求

- 安全需求参考资料
  - OWASP应用安全验证标准
  - 常见攻击模式枚举和分类CAPEC
  - OWASP API安全Top10文档

5.4API安全治理-SDL

API安全需求

华为产品安全基线	
1.保护用户通信内容	4条
2.保护用户隐私	7条
3.防止后门	5条
4.防止恶意软件、恶意行为	2条
5.访问通道控制	5条
6.系统加固	4条
7.应用安全	3条
8.加密	5条
9.敏感数据保护	4条
10.管理和维护安全	5条
11.安全启动和完整性保护	2条
12.安全资料	3条
13.安全编码	2条
14.安全编译	1条
15.生命周期管理	2条

5.4API安全治理-SDL

API安全设计

- 将安全需求分析后确定所采取的的风险处理措施，应用到技术架构设计中，并形成相关规范
  - 明确设计要求，制定安全设计规范
  - 进行受攻击面分析，制定受攻击面降低措施
  - 进行威胁分析，建立威胁模型，明确风险并建立相应消减机制

5.4API安全治理-SDL

API安全设计

- 明确设计要求
  - 基本隐私设计：明确国家法律法规，对获取记录用户隐私的相关产品做出设计要求。在告知用户并征得同意的情况下，仅收集程序必须用到的隐私数据
  - 基本安全设计：默认安全（在程序的默认配置中，需包括安全配置，确保程序初始状态是安全的）和最低加密（在程序处理之前，对所有数据进行严格验证或通过加密方式进行可靠地传输）

5.4API安全治理-SDL

API安全设计

- 减少受攻击面
  - 尽量减少暴露给恶意用户可能访问到的程序相关资源
  - 系统服务最小原则
  - 应用最小权限原则
  - 分层防御原则

5.4API安全治理-SDL

API安全设计

- 威胁建模
  - 一种分析应用程序威胁的方法，可识别潜在的安全问题并实施相应的解决或缓解措施。
  - 通常使用微软提出的SERIDE威胁等级分类法
  - 将威胁分为：Spoofing仿冒，Tampering篡改，Repudiation抵赖，Information Disclosure信息泄露，Denial of Service拒绝服务，Elevation of Privilege权限提升六部分

5.4API安全治理-SDL

API安全设计

◆ 威胁建模

- 六大威胁的定义、安全属性、消减措施

威胁	安全属性	定义	举例	消减措施
仿冒S	认证	冒充人或物	冒充其他用户账号	身份管理、认证（密码、单点、双因素、证书）、会话管理
篡改T	完整性	修改数据或代码	修改订单信息	完整性校验、访问控制
抵赖R	审计	不承认做过某行为	不承认修改行为	安全管理、安全审计、监控
信息泄露I	保密性	信息被泄露或窃取	用户信息被泄露	敏感信息保护、数据加密、访问控制
拒绝服务D	可用性	消耗资源、服务不可用	DDOS导致网站不可用	DDOS防护、负载均衡
权限提升E	授权	未经授权获取、提升权限	普通用户提升到管理员	授权、最小化

5.4API安全治理-SDL

API安全设计

- 安全设计参考资料
  - 各大云厂商API使用说明文档

序号	云厂商名称	
1	亚马逊aws	https://docs.aws.amazon.com/general/latest/gr/aws-apis.html
2	微软Azure	https://learn.microsoft.com/zh-cn/azure/architecture/best-practices/api-design
3	阿里云	https://help.aliyun.com/product/29462.html
4	腾讯云	https://cloud.tencent.com/document/api

5.4API安全治理-SDL

API安全设计

- 安全设计参考资料
  - 各大厂API使用说明文档

序号	平台名称	
1	微信开放平台	https://open.weixin.qq.com/
2	支付宝开放平台	https://open.alipay.com/api
3	淘宝开放平台	https://open.taobao.com/docCenter
4	抖音开放平台	https://open.douyin.com/platform/doc
5	百度AI开放平台	https://ai.baidu.com/ai-doc

5.4API安全治理-SDL

API安全设计

◆ 安全设计参考资料

◘ 企业或组织公开的API安全设计文档

序号	文档名称	
1	OWASP REST安全检查表	https://cheatsheetseries.owasp.org/cheatsheets/REST_Security_Cheat_Sheet.html
2	微服务架构安全检查表	https://cheatsheetseries.owasp.org/cheatsheets/Microservices_based_Security_Arch_Doc_Cheat_Sheet.html
3	Web Service安全检查表	https://cheatsheetseries.owasp.org/cheatsheets/Web_Service_Security_Cheat_Sheet.html
4	API安全检查表	https://github.com/shieldfy/API-Security-Checklist
5	OAuth2.0最佳安全实践	https://pragmaticwebsecurity.com/files/cheatsheets/oauth2securityfordevelopers.pdf
6	JWT安全检查表	https://pragmaticwebsecurity.com/files/cheatsheets/jwt.pdf

5.4API安全治理-SDL

API安全实现

- 侧重编码开发，根据安全设计阶段的产物，选择相应的编程语言，完成API编码实现过程。
- 一般包括三个阶段：
  - 安全编码培训
  - 安全编码
  - 静态检测

5.4API安全治理-SDL

API安全实现

- 静态检测工具衡量指标
  - 支持的开发语言
  - 漏报率
  - 误报率
  - 运行环境与配置
  - 报告格式
  - 报告内容
  - 性价比或购买方式

5.4API安全治理-SDL

API安全实现

◆ API安全实现相关工具

- 安全编码规范

序号	文档名称	
1	Spring Security官方文档	https://docs.spring.io/spring-security/reference/index.html
2	绿盟-安全编码规范	http://blog.nsfocus.net/web-vulnerability-analysis-coding-security/
3	OWASP 安全编码实践	https://owasp.org/www-project-secure-coding-practices-quick-reference-guide/
4	SEI CERT安全编码规范	https://wiki.sei.cmu.edu/confluence/display/seccode/SEI+CERT+Coding+Standards

5.4API安全治理-SDL

API安全实现

◆ API安全实现相关工具

- 静态检测工具

序号	工具名称	是否开源	适用语言
1	Checkmarx	否	Java、.net、JavaScript、C/C++等
2	Fortify SCA	否	Java、JSP、.net、C#、C/C++等
3	MobSF	是	Android
4	PMD	是	Java、Python、C/C++、PHP等
5	Dependency-Check	是	Java、.net为主

5.4API安全治理-SDL

API安全验证

- 对SDL流程中安全需求、安全设计、安全实现环节的验证，通过管理手段和技术手段来评估安全实现的正确性，保证安全设计与安全实现的一致性。
- 一般包括三种方式：
  - 工具验证
  - SDL过程保证
  - 个人隐私合规类的安全验证

5.4API安全治理-SDL

API安全验证

- 安全验证工具
  - Burp Suite工具，商业渗透测试套件
  - Astra，REST API安全测试工具
  - OWASP ZAP，Web应用渗透测试工具
  - FuzzDB，应用模糊测试工具
  - Postman，商业API管理工具
  - SoapUI，商业API测试软件

5.4API安全治理-DevSecOps

软件产品生命周期

- 产品需求-设计-开发-测试-部署维护
- 涉及几个不同的团队
  - 负责产品的客户需求与市场推广
  - 产品的设计与开发
  - 产品的运维和客户服务

5.4API安全治理-DevSecOps

DevOps理念

- 业务高速发展，团队协作效率
- 一种新型的协作方式
  - 注重于构建容易维护和自动化运维的产品和服务
  - 起源于通过合并开发和运维实践
  - 消除隔离，统一关注点
  - 提升团队和产品的效率和性能

5.4API安全治理-DevSecOps

DevSecOps背景

- 安全在软件质量中逐渐被重视
- 各团队协作
  - 安全团队协作不畅
  - 安全工作滞后
  - 研发流程缺少安全控制环节

5.4API安全治理-DevSecOps

DevSecOps产生

- 2012年，Gartner首次提出DevSecOps的概念，一种糅合开发、安全及运营理念的全新安全管理模式
- 2016年，Gartner发布报告《DevSecOps: How toSeamlessly integrate Security into DevOps》，核心理念是：安全是全体IT团队所有成员的责任，要贯穿到业务生命周期的每一个环节。
- 对应 DevOps 快速交付和灵活响应变化，DevSecOps 的价值是在不牺牲安全性的前提下，快速落地和实施安全。

5.4API安全治理-DevSecOps

实施关键点

- 安全左移：在设计开发阶段就引入安全环节，从整个生命周期开发与维护的视角关注安全质量与成本
- 安全自动化：关注整个流程的工具化与自动化，在CI/CD工具链嵌入安全检查流程，减少对研发流程的影响
- 持续运营：采取循序渐进的方式在组织中推进解决安全问题，构建数字化运营指标，获得最佳的投入产出比

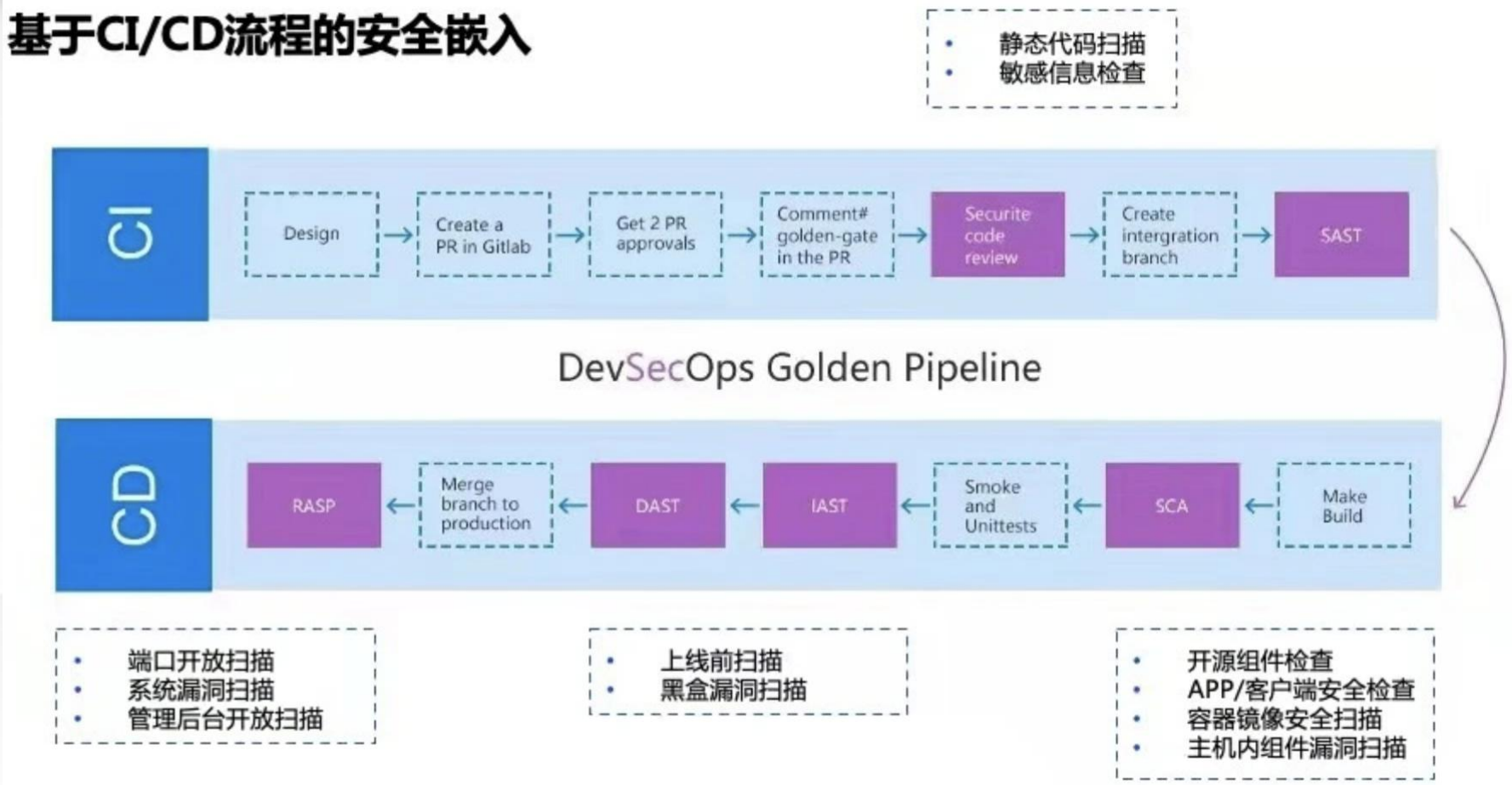
5.4API安全治理-DevSecOps

DevSecOps管道

- 为减轻安全活动介入后对原有流程的影响，将安全工作加入现有的开发平台和工具中
  - 在项目管理平台导入安全等级定义
  - 在需求管理平台导入安全需求
  - 在持续集成平台与安全测试能力对接
  - 在缺陷管理系统导入安全测试结果

5.4API安全治理-DevSecOps

DevSecOps管道



5.4API安全治理-DevSecOps

DevSecOps管道

- DevOps管道
  - 持续集成、持续发布、基础设施架构
- DevSecOps提倡持续安全，融入到DevOps各个环节
  - 安全团队与DevOps团队，定义和实施安全控制要求，明确安全基线
  - CI/CD持续运行，静态检测、动态扫描、运营检测，伴随版本迭代持续运转

5.4API安全治理-DevSecOps

DevSecOps平台

- 流水线管理
  - 为各个角色提供统一入口，包括任务编排与配置，过程数据的统计度量，组织机构与用户的基础信息以及权限访问控制等
- 工程管理
  - 从项目的角度，管理需求和任务以及整体缺陷，包括功能缺陷，安全缺陷，质量缺陷等
- 构建管理
  - 为持续集成的构建环境，管理编译选项配置，容器构建参数配置等，通过编译与构建，生成代码制品

5.4API安全治理-DevSecOps

DevSecOps平台

- 制品库管理
  - 构建完成后生成的代码制品，统一存放的制品库，并对代码制品进行持续检测，关注组件依赖于组件漏洞，组件许可协议等
- 部署管理
  - 管理自动化部署的各种环境，如应用、主机、基础设施环境等。管理部署架构以及部署前的各项自动化测试，比如功能测试、安全测试、性能测试等。
- 运维管理
  - 通过线上服务的周期性监控，及时发现线上问题，做出应急响应

5.4API安全治理-DevSecOps

API安全实践

- 设置关键卡点
  - 落实自动化API安全测试
  - 采用API网关
  - 接入Web应用防火墙

5.4API安全治理-DevSecOps

API安全实践

- 构建不同层面的安全能力
  - 持续集成管道安全，保证CI/CI环境的安全性
  - API基础设施安全，保证API运行环境的安全性

5.4API安全治理-API网关

什么是API网关

- 是面向API、串行、集中化的管控工具，提供高性能的API托管服务，使用户能够快速、低成本、低风险地开放服务能力

5.4API安全治理-API网关

API网关优势

- 将内部服务和外部调用隔离，隐藏 内部服务数据，保障了服务的私密性和安全性
- 在整体架构上，可以让业务提供者抽出更多精力来关注核心业务能力建设，而不是通用的安全性、流控等边界特性的问题
- 在快速增加新业务或改变原有应用系统、服务时，对现有架构和应用程序的影响降低到最小

5.4API安全治理-API网关

API网关功能

- 由核心控制系统和后台管理系统两部分构成
- 核心控制系统
  - 为满足业务需要所对外提供的核心API能力的总称
  - 处理安全策略、流量控制、服务鉴权、熔断、参数校验、参数映射、协议转换、服务生命周期等所有核心业务能力
- 后台管理系统
  - 用于辅助核心控制系统所提供的能力总称
  - 用户管理、应用接入管理、SDK和API文档生成、服务授权控制、服务策略绑定等功能的管理能力，为管理人员提供可视化的操作界面

5.4API安全治理-API网关

API网关边界

- 和后端API服务之间的关系
  - API提供者在提供稳定的API后，在API网关中注册并发布API，才可以本终端业务系统调用
- 和客户端应用程序之间的关系
  - 不同的客户端或终端应用程序首先访问API网关，经过一系列的API控制器、网关路由到目标API

5.4API安全治理-API网关

API网关边界

- 和运维监控系统之间的关系
  - API网关接入运维监控系统，用于监控网关和服务器的运行情况，也监控各个已注册API的运行健康情况，并在异常时触发告警
- 和日志平台之间的关系
  - API网关接入日志平台，用于采集API的调用信息，完成问题分析、调用链跟踪、调用数据统计等

5.4API安全治理-API网关

API网关应用场景

- 多种API协议转换
  - 外部调用者使用HTTP/HTTPs，WebSocket协议
  - 后端服务使用RESTful服务，Dubbo服务，WebService服务，WebSocket服务，SpringCloud注册中心等

5.4API安全治理-API网关

API网关应用场景

- API安全接入
  - 服务鉴权、流量控制、熔断机制、参数校验等多种安全机制
  - 不同机制间的自由组合

5.4API安全治理-API网关

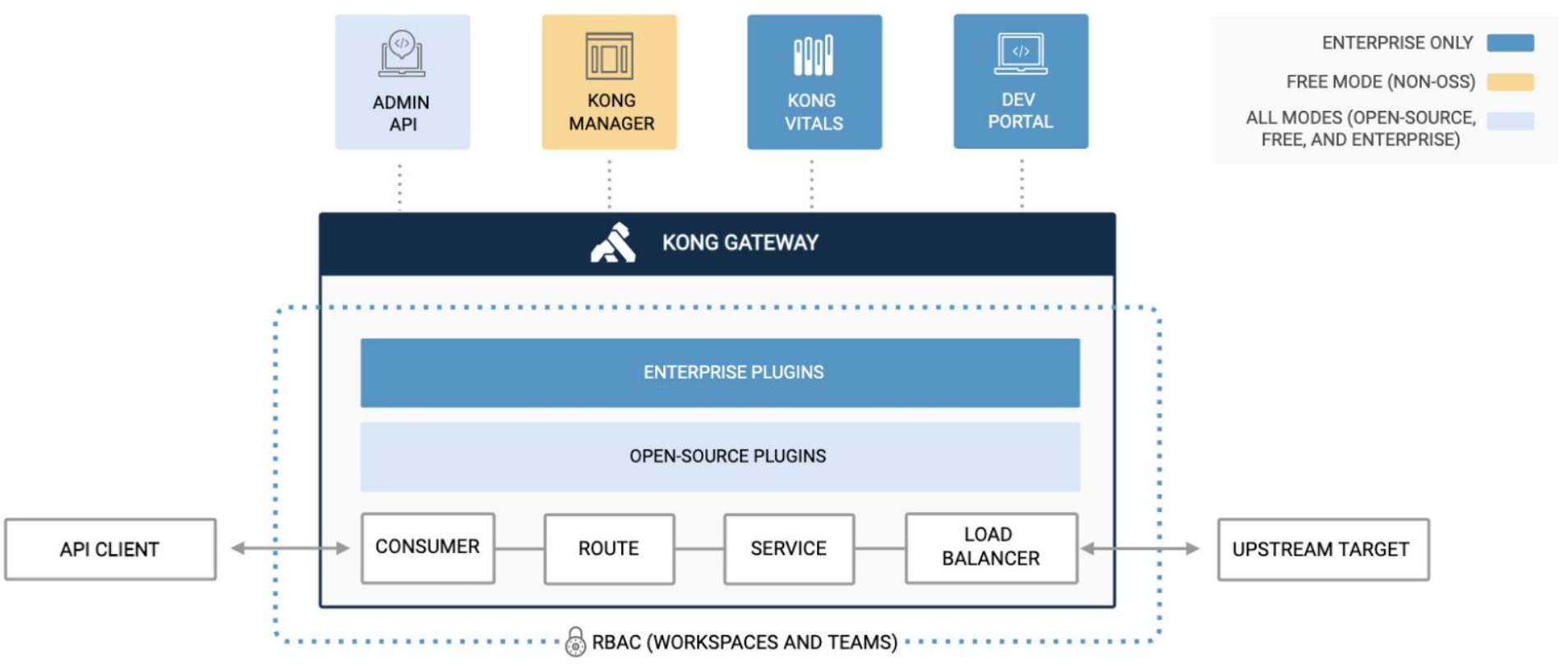
开源API网关

- Kong API网关介绍
  - 以OpenResty(Nginx + Lua模块)为基础
  - 采用Cassandra或PostgreSQL作为数据存储
  - 通过集群化模式为企业提供高效可用的API管理服务能力
  - 通过插件的形式，提供负载均衡、日志审计、身份验证、速率限制、协议转换等功能

5.4API安全治理-API网关

开源API网关

◆ Kong API网关架构



5.4API安全治理-API网关

开源API网关

- Kong API网关安全特性
  - 身份认证插件，支持Basic，OAuth2.0，JWT，LDAP等认证
  - 访问控制插件，支持ACL，CORS，请求路径控制，IP限制，爬虫检测等
  - 流量控制插件，支持请求限流，上游响应限流，应答限流，集群限流等
  - 日志审计插件，支持文件日志，httpserver日志，StatsD日志等
  - 安全防护插件，集成ACME协议，API威胁保护等
  - 协议转换插件，支持GraphQL，RESTful，gRPC等API协议

5.4API安全治理-API网关

开源API网关

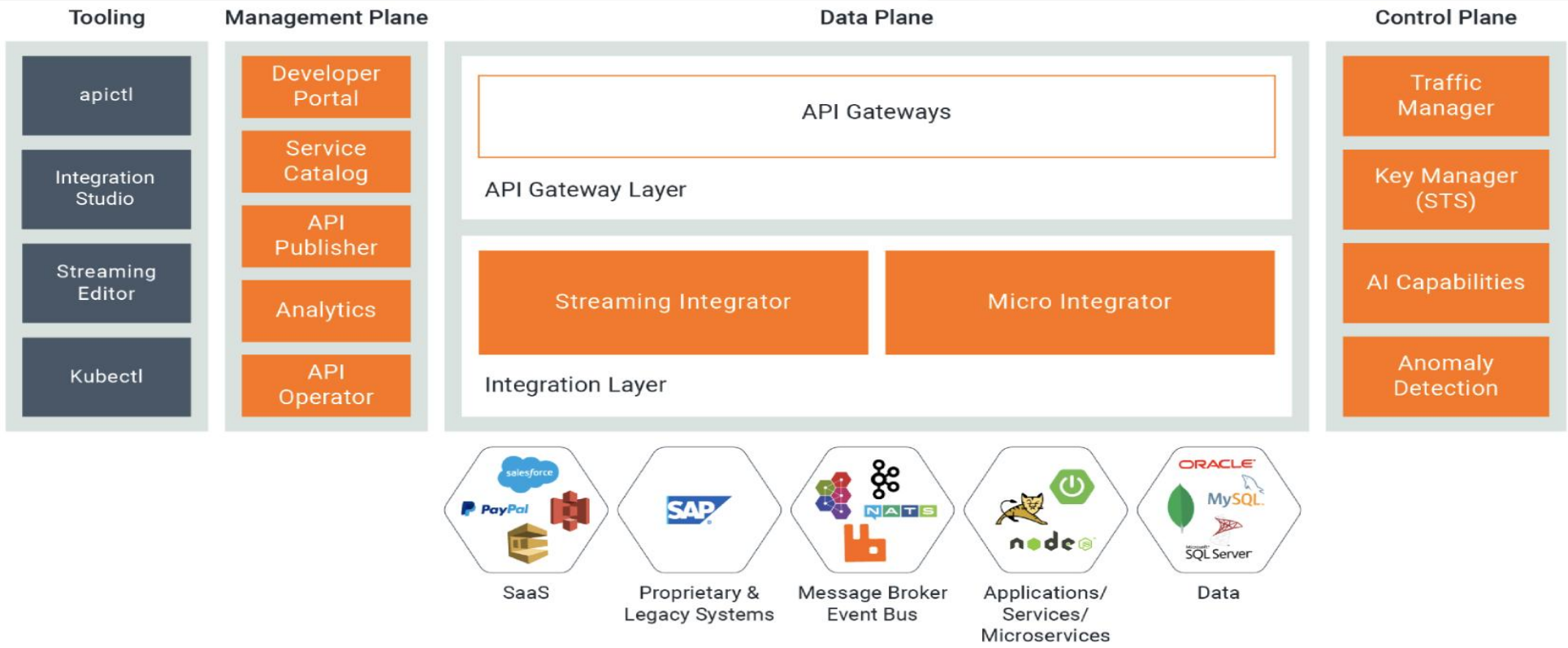
◆WSO2 API管理平台介绍

- 开源企业级API管理解决方案
- 从API的全生命周期管理、应用程序开发、第三方合作伙伴调用、内部应用程序开发等使用者的角度为客户提供API访问控制、速率限制、流量分析、异常检测、DevSecOps集成等功能
- 由API发布者，API开发者门户，API网关，API密钥管理器，API流量管理器等模块组成
- 通过插件的形式，提供负载均衡、日志审计、身份验证、速率限制、协议转换等功能

5.4API安全治理-API网关

开源API网关

◆WSO2 API管理平台架构



5.4API安全治理-API网关

开源API网关

- WSO2 API管理平台构成
  - 由API发布者，API开发者门户，API网关，API密钥管理器，API流量管理器等模块组成
  - API提供者通过API发布者模块定义和管理API
  - API使用者通过API开发者门户网站发现和使用API
  - API网关、API密钥管理器、API流量管理器等模块为API服务安全、便捷地使用提供强大的功能保护

5.4API安全治理-API网关

开源API网关

- WSO2 API管理平台安全特性
  - 身份认证：支持Http Basic，证书/密钥，OAuth2.0，JWT等认证方式
  - 授权与访问控制：提供基于范围和基于XACML的细粒度访问控制机制
  - API审核：集成API安全平台42Crunch，提供安全审核功能
  - API威胁保护：提供基于正则威胁，JSON威胁，XML威胁的防护
  - 限流：支持多谢限流策略，比如吞吐量、IP地址和范围，HTTP请求头等

5.4API安全治理-数据隐私

数据隐私法律框架

- 国内法律法规
  - 中华人民共和国网络安全法，简称网安法，2017-06-01起施行
  - 制订时间相对较早，主要是侧重于网络安全，第一次明确提出网络运管者对个人信息的全生命周期保护
  - 第四章第四十一条：网络运营者收集、使用个人信息，应当遵循合法、正当、必要的原则，公开收集、使用规则，明示收集、使用信息的目的、方式和范围，并经被收集者同意。网络运营者不得收集与其提供的服务无关的个人信息，不得违反法律、行政法规的规定和双方的约定收集、使用个人信息，并应当依照法律、行政法规的规定和与用户的约定，处理其保存的个人信息。

5.4API安全治理-数据隐私

数据隐私法律框架

- 国内法律法规
  - 中华人民共和国数据安全法，简称数安法，2021-09-01起施行
  - 宏观地确立数据保护的相关制度，并对数据处理者提出相关的义务要求，它的范围更广，包括个人信息与非个人信息
  - 第三章，数据安全制度
  - 第四章，数据安全保护义务
  - 第五章，政务数据安全与开放

5.4API安全治理-数据隐私

数据隐私法律框架

- 国内法律法规
  - 中华人民共和国个人信息保护法，简称个保法，2021-11-01起施行
  - 侧重于保护个人信息，但是相较于《数据安全法》，《个人信息保护法》提出了更多具体的场景
  - 第二章，个人信息处理规则
  - 第三章，个人信息跨境提供的规则
  - 第四章，个人在个人信息处理活动中的权利
  - 第五章，个人信息处理者的义务
  - 第六章，履行个人信息保护职责的部门

5.4API安全治理-数据隐私

数据隐私法律框架

- 国外法律法规
  - 欧盟《通用数据保护条例》GDPR，2018-05-25起施行
  - （欧盟）有史以来最为严格的网络数据管理法规
  - 适用范围极为广泛，任何收集、传输、保留或处理涉及到欧盟所有成员国内的个人信息的机构组织均受该条例的约束。
  - 基本原则：合法公开透明原则，目的限定原则，数据最小化原则，准确原则，有限留存原则，完整保密原则
  - 主体权利：知情权，访问权，更正权，被遗忘权，处理限制权，反对权，，可携带权，通知义务和自动决策等

5.4API安全治理-数据隐私

数据隐私的含义

- 国标《信息安全技术-个人信息安全规范》
- 个人信息
  - 以电子或者其他方式记录的能够单独或者与其他信息结合识别特定自然人身份或者反映特定自然人活动情况的各种信息。
  - 包括姓名、出生日期、身份证件号码、个人生物识别信息、住址、通信通讯联系方式、通信记录和内容、账号密码、财产信息、征信信息、行踪轨迹、住宿信息、健康生理信息、交易信息等

5.4API安全治理-数据隐私

数据隐私的含义

- 个人敏感信息
  - 一旦泄露、非法提供或滥用可能危害人身和财产安全，极易导致个人名誉、身心健康受到损害或歧视性待遇等的个人信息。
  - 包括身份证件号码、个人生物识别信息、银行账户、通信记录和内容、财产信息、征信信息、行踪轨迹、住宿信息、健康生理信息、交易信息、14 岁以下（含）儿童的个人信息等。

5.4API安全治理-数据隐私

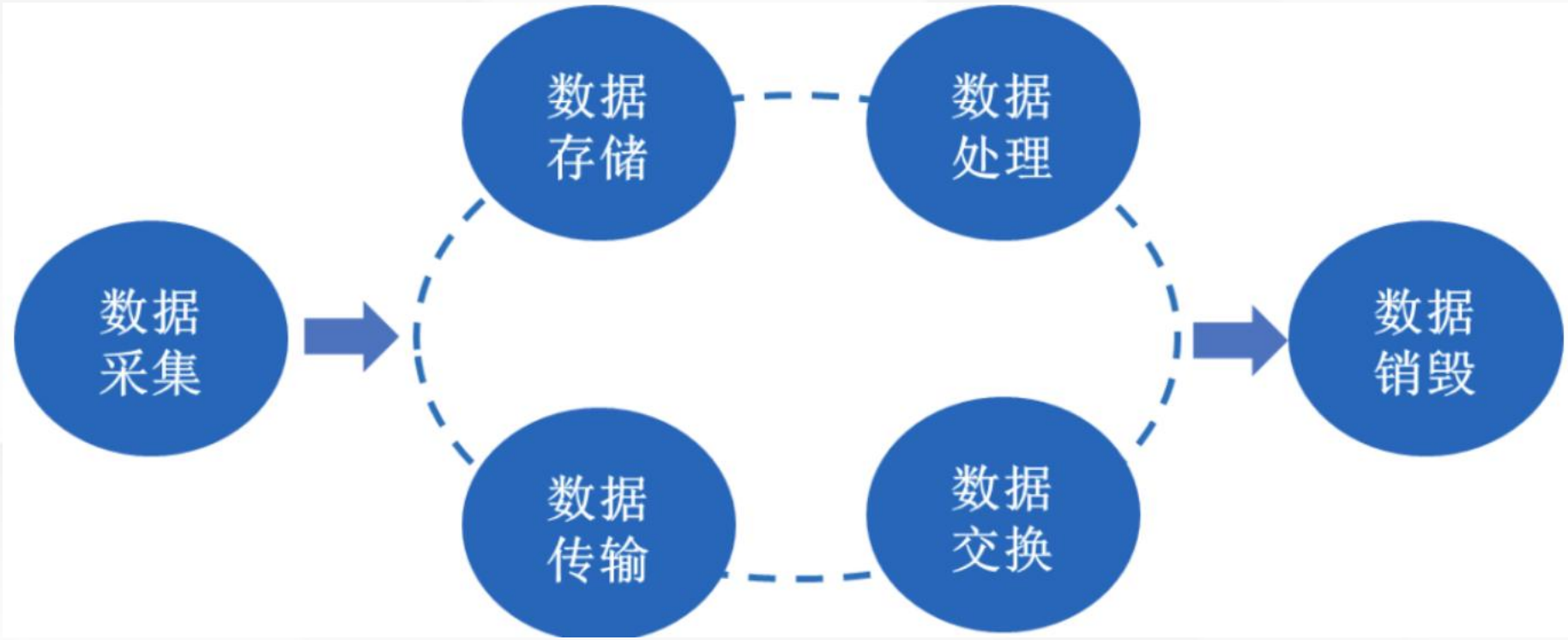
数据生命周期

- 当在API中开展数据隐私保护时，重点是保护个人信息，尤其是个人敏感信息的安全
- 需要从数据的收集、存储、使用、共享、转让、公开披露等信息处理各个环节中关注安全性

5.4API安全治理-数据隐私

数据生命周期

◆ 国标《信息安全技术-数据安全能力成熟度模型》



5.4API安全治理-数据隐私

数据生命周期各阶段安全

- 数据采集安全
  - 数据分类分级
  - 数据采集和获取
  - 数据源鉴别及记录
  - 数据质量管理

5.4API安全治理-数据隐私

数据生命周期各阶段安全

- 数据传输安全
  - 数据传输加密
  - 网络可用性管理

5.4API安全治理-数据隐私

数据生命周期各阶段安全

- 数据存储安全
  - 存储介质安全
  - 逻辑存储安全
  - 数据备份与恢复

5.4API安全治理-数据隐私

数据生命周期各阶段安全

- 数据处理安全
  - 数据脱敏处理
  - 数据分析安全
  - 数据正当使用
  - 数据处理环境安全
  - 数据导入导出安全

5.4API安全治理-数据隐私

数据生命周期各阶段安全

- 数据交换安全
  - 数据共享安全
  - 数据发布安全
  - 数据接口安全

5.4API安全治理-数据隐私

数据生命周期各阶段安全

- 数据销毁安全
  - 数据销毁处置
  - 介质销毁处置

5.4API安全治理-数据隐私

数据生命周期各阶段安全

- 通用安全
  - 数据安全策略规划
  - 组织人员管理
  - 数据资产管理
  - 元数据管理
  - 安全事件应急

QA