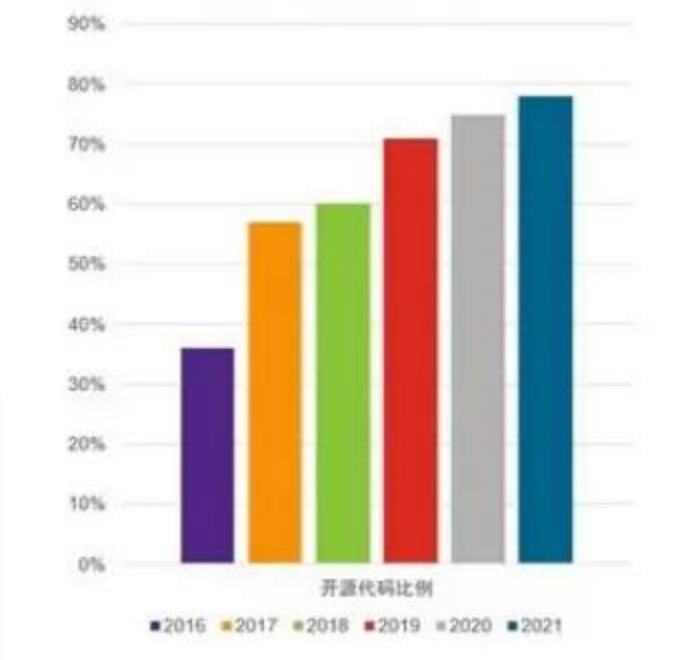

GOTC 2023
全球开源技术峰会
THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE
OPEN SOURCE, INTO THE FUTURE
「聚焦开源安全专题论坛」专场
基于代码疫苗技术的开源软件供应链安全治理
董毅 2023年05月28日


GOTC 2023
全球开源技术峰会
THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE
OPEN SOURCE, INTO THE FUTURE
01 开源风险

开源组件是软件中的主要成分

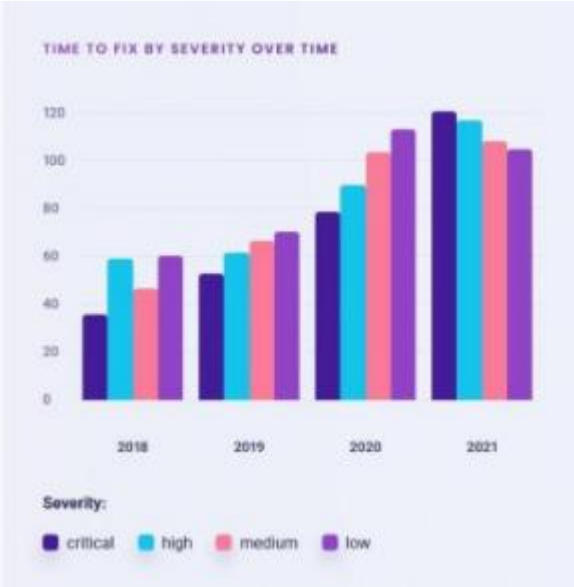
- 2021年开源代码的比例已经高达78%，计算机硬件和半导体、网络、能源与清洁科技、物联网这四大行业均100%地使用到了开源软件，开源在促进全球的软件创新方面发挥着越来越重要的作用。

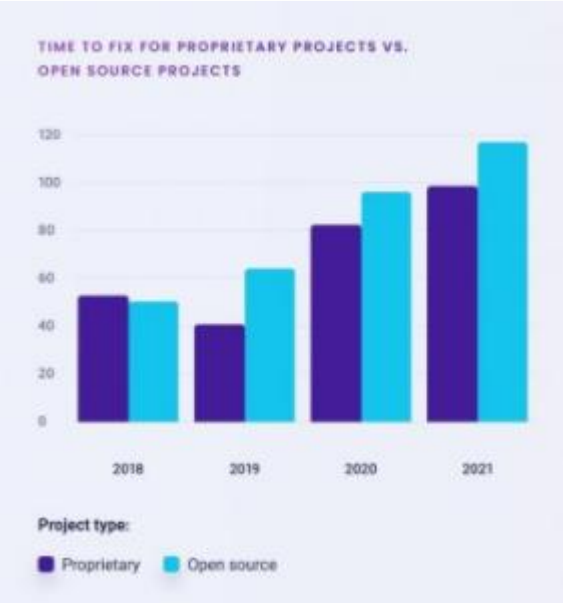


《2022年开源安全和风险分析》——新思科技

开源组件漏洞频发且难以修复

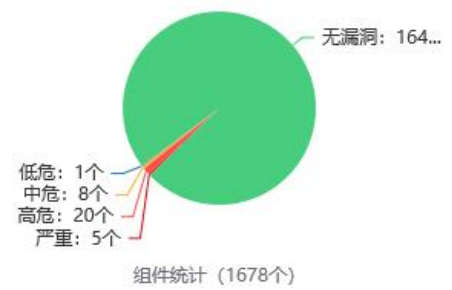
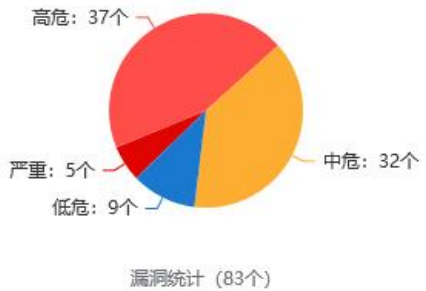
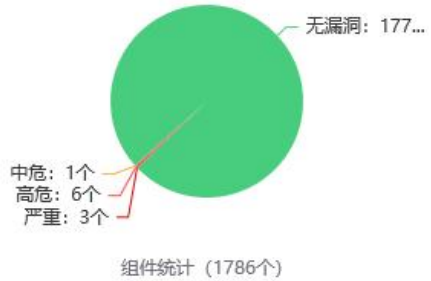
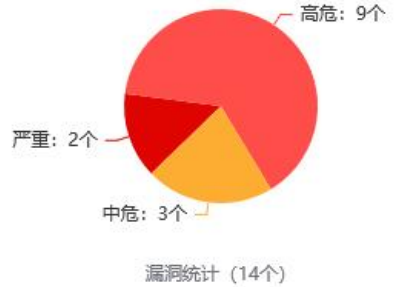
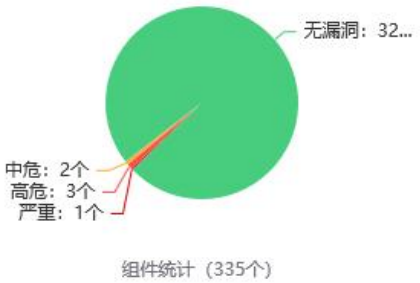
- 一个应用程序开发项目平均有 49 个漏洞和 80 个直接依赖项。
- 修复开源项目漏洞所需的时间也在稳步增加。早在 2018 年，修复安全漏洞平均需要 49 天。2021年，开发一个补丁大约需要 110 天。





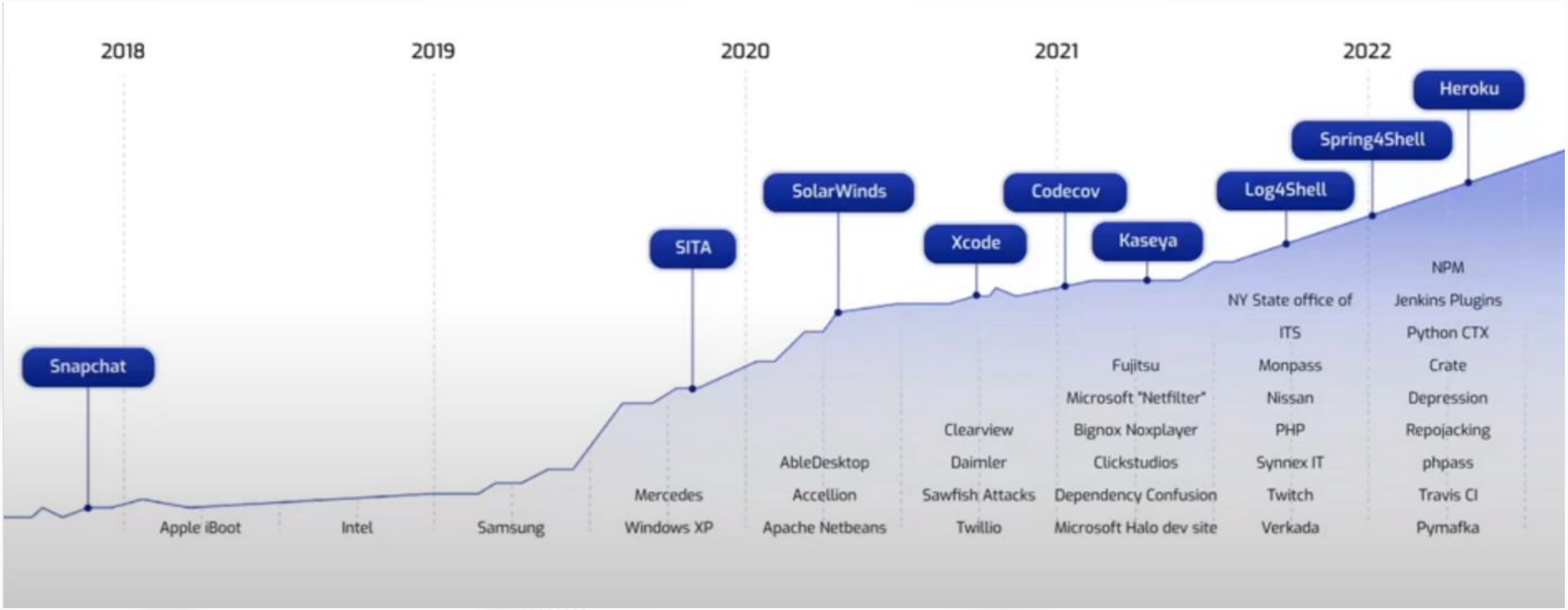
《The State of Open-Source Security》——Snyk&Linux基金会

开源应用漏洞数

	OpenSCA扫描结果	
h********r-main		
p********s-main		
c**********a-main		

严峻的开源供应链安全风险

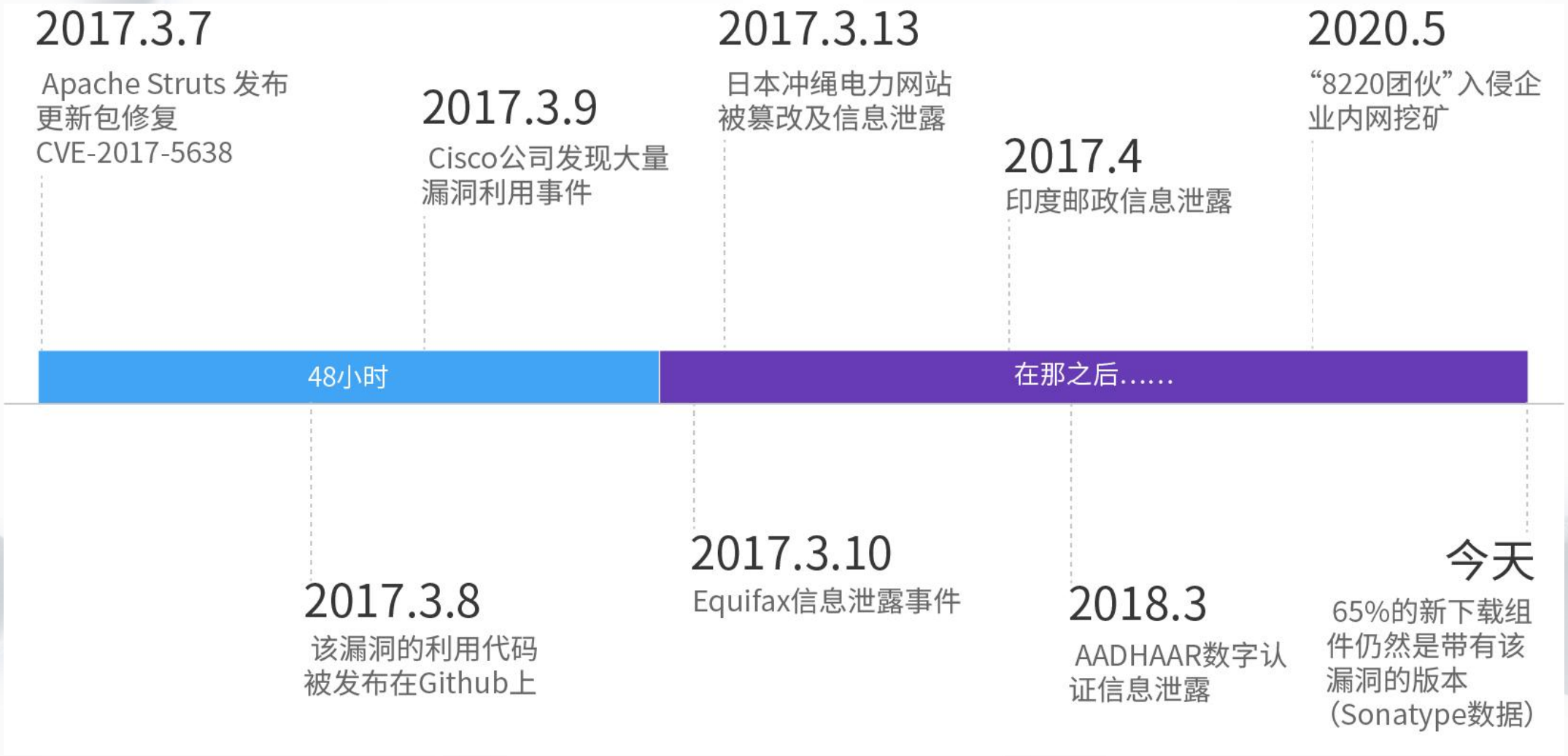
“到2025年，全球45%的组织会受到软件供应链攻击，比2021年增长三倍”——Gartner



2017 Struts2 S2-045（CVE-2017-5638）

- Apache Struts 2被曝存在远程命令执行漏洞，漏洞编号S2-045，CVE编号CVE-2017-5638，在使用基于Jakarta插件的文件上传功能时，有可能存在远程命令执行，导致系统被恶意用户利用。
- 恶意用户可在上传文件时通过修改HTTP请求头中的Content-Type值来触发该漏洞，进而执行系统命令。

S2-045时间轴

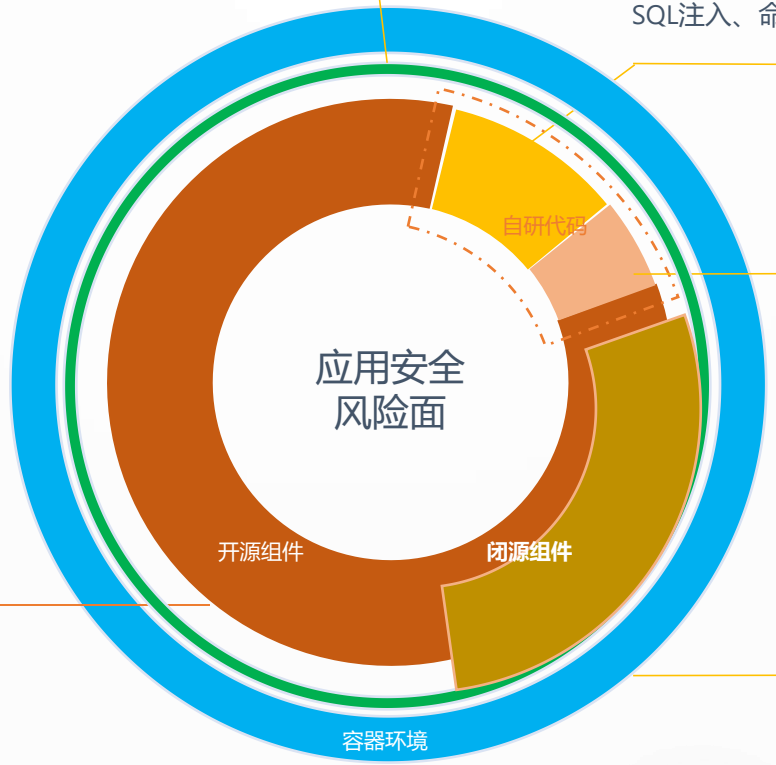

2017.3.7
Apache Struts 发布更新包修复CVE-2017-5638
2017.3.9
Cisco公司发现大量漏洞利用事件
2017.3.13
日本冲绳电力网站被篡改及信息泄露
2017.4
印度邮政信息泄露
2020.5
“8220团伙”入侵企业内网挖矿
48小时
在那之后……
2017.3.8
该漏洞的利用代码被发布在Github上
2017.3.10
Equifax信息泄露事件
2018.3
AADHAAR数字认证信息泄露
今天
65%的新下载组件仍然是带有该漏洞的版本（Sonatype数据）

关注新场景下的开源治理

信息技术演进

- 开发模式：瀑布 > 敏捷 > DevOps
- 应用架构：大型系统 > SOA > 微服务
- 基础设施：数据中心 > 托管服务器 > 云基础设施
- 服务器：物理机 > 虚拟化 > 容器化

聚焦到应用系统风险源头



API安全性

失效的用户认证、安全性、错误配置、注入等

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

合规需求、安全配置

未能满足安全合规、未建立安全基线、敏感数据泄漏

容器环境镜像风险

软件漏洞、恶意程序、敏感信息泄漏、不安全配置、仓库漏洞、不可信镜像

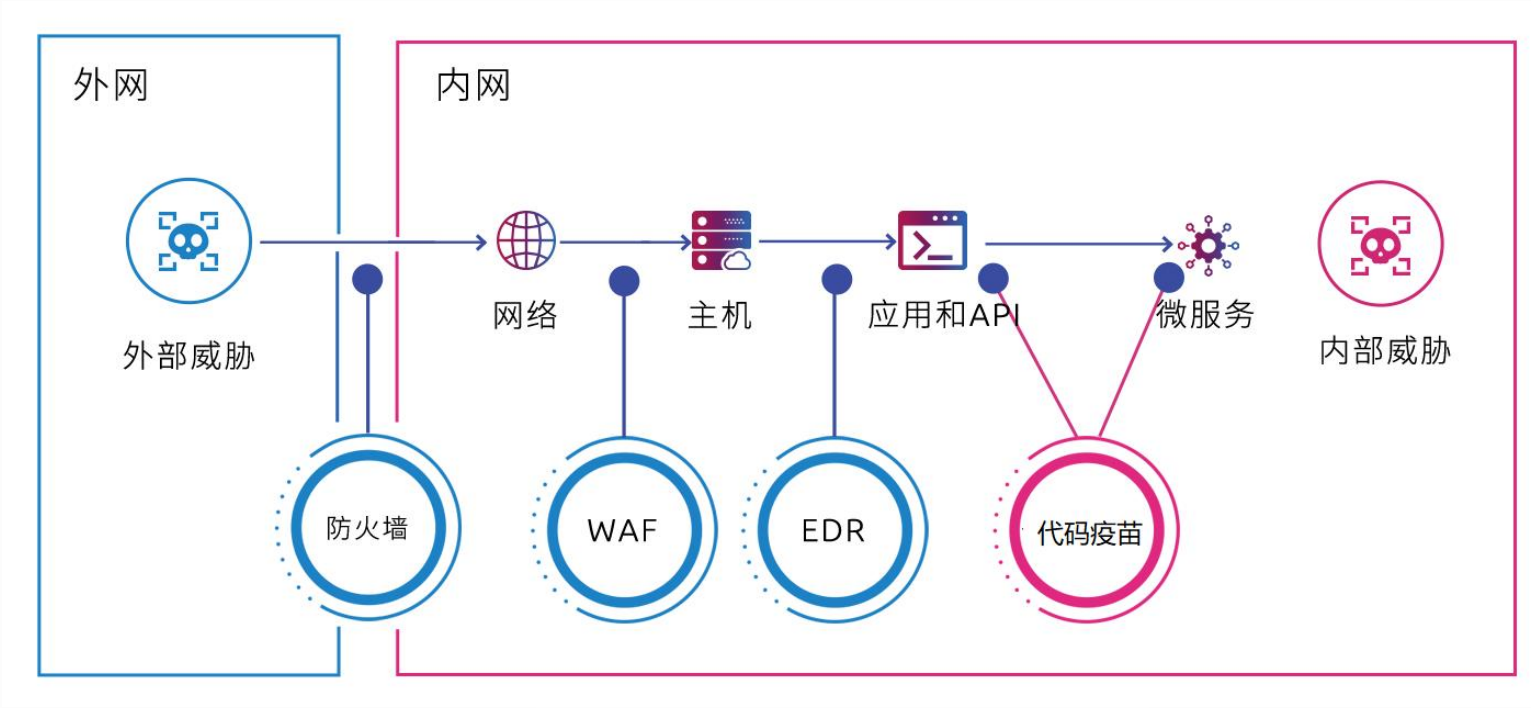
第三方组件

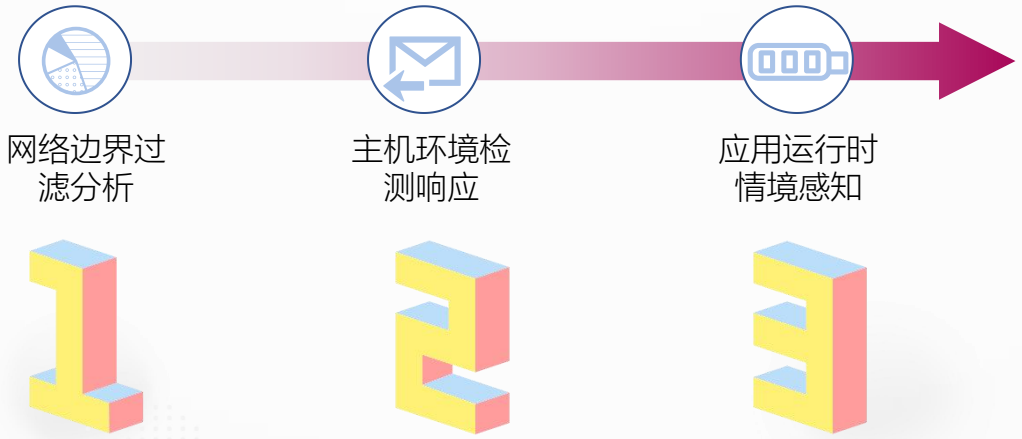
开源组件/闭源组件

CNNVD、CNVD、CVE等
开源许可风险


GOTC 2023
全球开源技术峰会
THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE
OPEN SOURCE, INTO THE FUTURE
02 代码疫苗技术

数字安全技术的三次关键演进





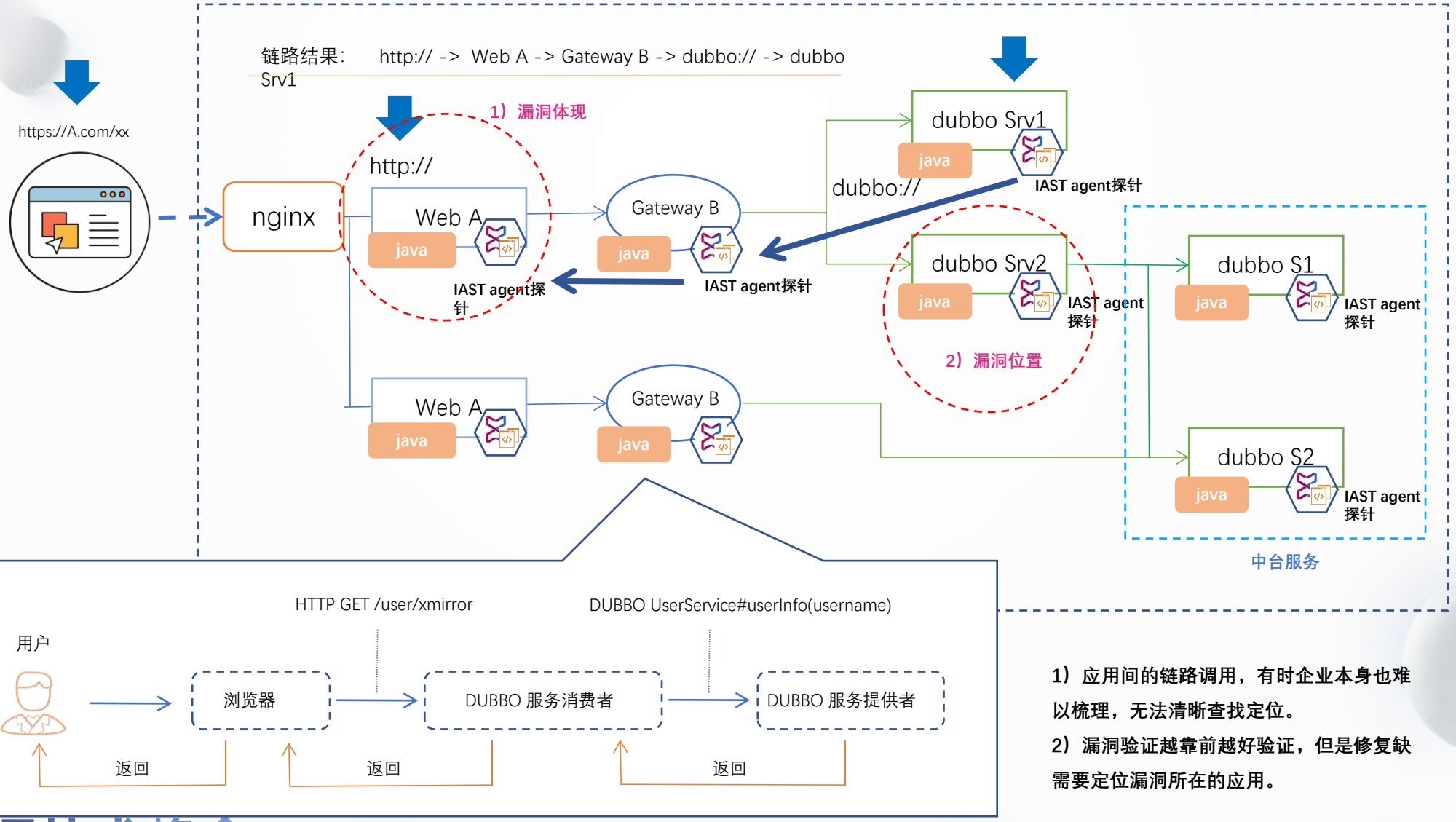
代码疫苗技术

一种新型的应用内探针技术，统一融合IAST、OSS、RASP、DAR、API、APM等安全能力，一个探针解决应用长期面临的安全漏洞、数据泄漏、运行异常、0day攻击等风险，减轻多探针运维压力的同时，为应用植入代码疫苗，实现应用与安全共生。

网络层安全 > 主机层安全 > 应用层安全

- 实时检测，不需要代码安全专家来逐行分析源代码
- 无需对原有代码逻辑进行修改，自免疫安全威胁
- 检测过程无脏数据，自动化基础度高，无需维护复杂的策略和规则
- **覆盖应用自研代码、第三方开源/闭源组件、数据安全**，并提供**积极检测与响应**能力
- 统一应用Agent安全探针，减轻多探针的运维压力

东西向流量-漏洞链路追踪



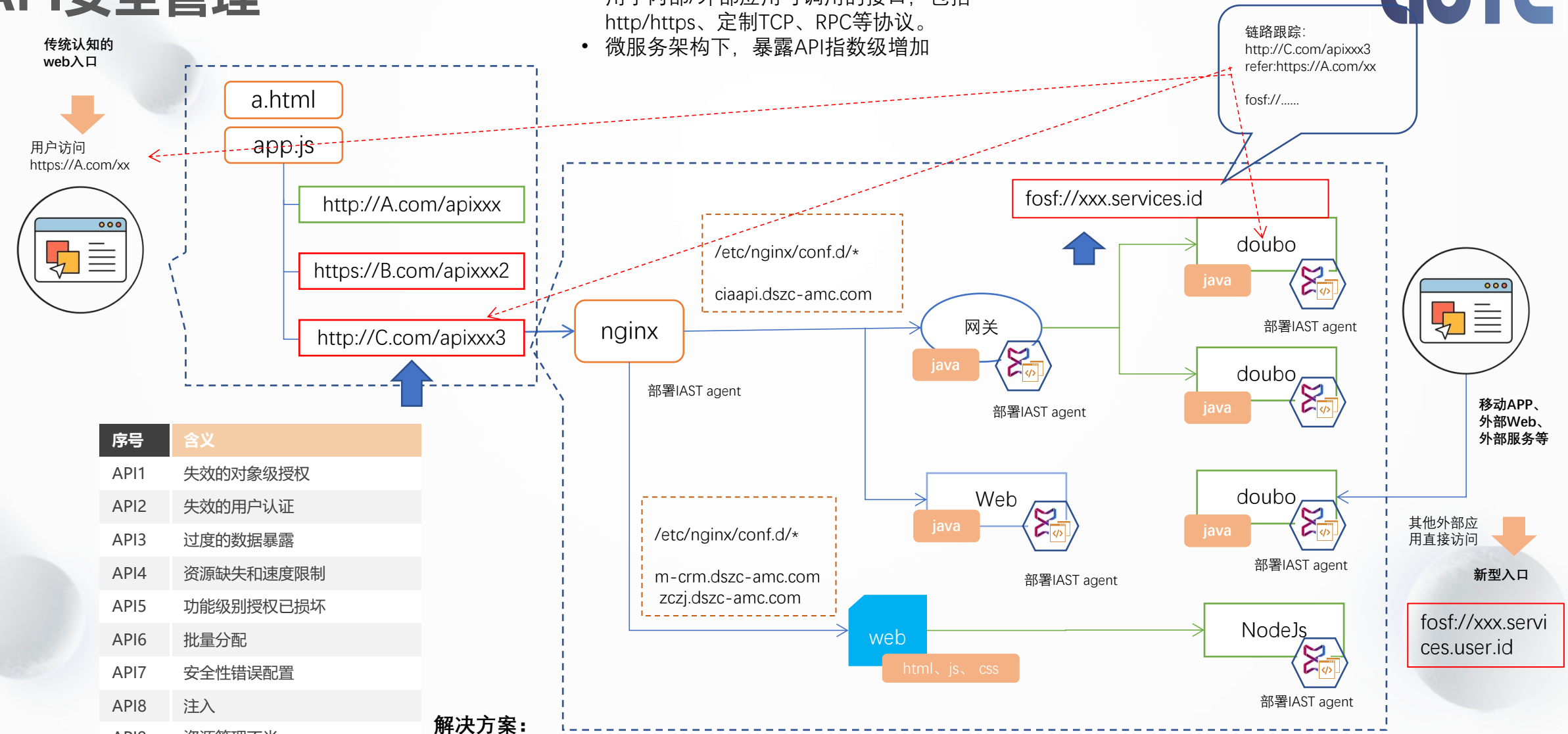
1）应用间的链路调用，有时企业本身也难以梳理，无法清晰查找定位。

2）漏洞验证越靠前越好验证，但是修复缺需要定位漏洞所在的应用。

API安全管理

什么是API?

- 用于内部/外部应用可调用的接口，包括http/https、定制TCP、RPC等协议。
- 微服务架构下，暴露API指数级增加



序号	含义
API1	失效的对象级授权
API2	失效的用户认证
API3	过度的数据暴露
API4	资源缺失和速度限制
API5	功能级别授权已损坏
API6	批量分配
API7	安全性错误配置
API8	注入
API9	资源管理不当
API10	日志和监控不足

解决方案：

- API资产梳理（暴露面、风险分析）
- API链路调用威胁阻断
- OWASP API安全 TOP 10（权限控制、注入等）

数据安全检测


```
{
  "category": "SQL_Injection",
  "event": "Data Exfiltration",
  "severity": "high",
  "timestamp": "2021-01-01 00:00:00",
  "query": "SELECT name, password FROM user name='' or 1=1'",
  "statement_type": "SELECT",
  "table": "user",
  "columns": "['name', 'password']",
  "session_id": "xxxxxxxxxx",
  "filename": "AccountInfo.java:18",
  "call_stack": "org.apache.tomcat.dbcp.dbcp2.DelegatingConnection.prepar
  "os": "Mac OS X 10.15.7",
  "ip": "192.168.172.1",
  "hostname": "node-01.xmirror.com",
  "url": "http://test.case/vuln/account?name='%20OR%201=1"
}
```


splunk>

elastic

- 基于真实攻击事件生成数据，您需要的所有内容均包含在同一条日志记录中。
- 提供多种维度的图、表协助您的团队进行数据分析。
- 也可对接至其他 SIEM 平台或自研风险度量平台与您的其它数据汇总分析

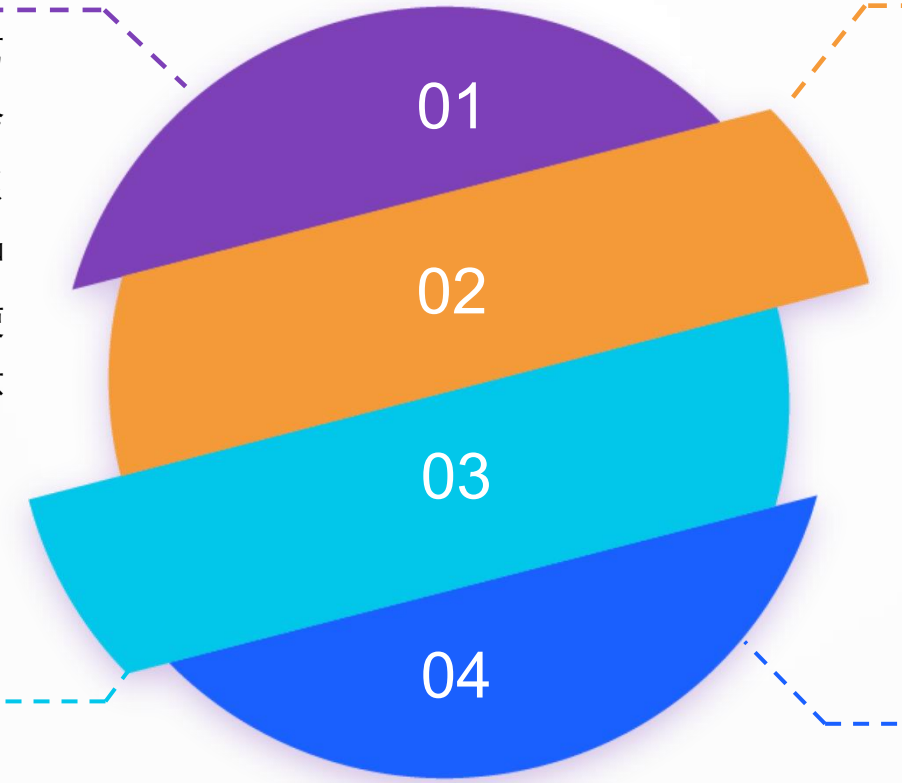
个人基本资料	个人姓名、生日、性别、民族、国籍、家庭关系、住址、个人电话号码、电子邮件地址等
个人身份信息	身份证、军官证、护照、驾驶证、工作证、出入证、社保卡、居住证等
个人生物识别信息	个人基因、指纹、声纹、掌纹、耳廓、虹膜、面部识别特征等
网络身份标识信息	个人信息主体账号、IP 地址、个人数字证书等
个人健康生理信息	个人因生病医治等产生的相关记录，如病症、住院志、医嘱单、检验报告、手术及麻醉记录、护理记录、用药记录、药物食物过敏信息、生育信息、以往病史、诊治情况、家族病史、现病史、传染病史等，以及与个人身体健康状况相关的信息，如体重、身高、肺活量等
个人教育工作信息	个人职业、职位、工作单位、学历、学位、教育经历、工作经历、培训记录、成绩单等
个人财产信息	银行账户、鉴别信息(口令)、存款信息（包括资金数量、支付收款记录等）、房产信息、信贷记录、征信信息、交易和消费记录、流水记录等，以及虚拟货币、虚拟交易、游戏类兑换码等虚拟财产信息
个人通信信息	通信记录和内容、短信、彩信、电子邮件，以及描述个人通信的数据（通常称为元数据）等
联系人信息	通讯录、好友列表、群列表、电子邮件地址列表等
个人上网记录	指通过日志储存的个人信息主体操作记录，包括网站浏览记录、软件使用记录、点击记录、收藏列表等
个人常用设备信息	指包括硬件序列号、设备 MAC 地址、软件列表、唯一设备识别码（如 IMEI/Android ID/IDFA/OpenUDID/GUID/SIM 卡 IMSI 信息等）等在内的描述个人常用设备基本情况的信息
个人位置信息	包括行踪轨迹、精准定位信息、住宿信息、经纬度等
其他信息	婚史、宗教信仰、性取向、未公开的违法犯罪记录等


GOTC 2023
全球开源技术峰会
THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE
OPEN SOURCE, INTO THE FUTURE
03 落地方案

开源软件供应链安全治理痛点

1. 理不清

企业不清楚在系统中使用了多少第三方开源组件。开源组件通常又会依赖其它更多组件，多级依赖关系使得整个组件结构更加复杂，这种结构的安全性对于应用的研发和使用来说，很多时候也是未知的，不可控的



2. 看不见

企业在使用开源组件的过程中，不知道它们中有的已产生过安全漏洞和知识产权风险。很多企业会使用非常老的组件和软件，其中很多爆发过安全漏洞，但没有及时去更新。对于这些已知漏洞的风险隐患，企业无法获悉，这种不可见性增加了危险系数

3. 找不到

企业在开源组件出现漏洞的时候，无法快速地定位受影响的组件以及评估影响的范围

4. 治不了

当企业明确漏洞影响的范围以及受影响的组件并定位到具体项目后，就需要进行相关治理工作，对组件进行相应的评估、缓解和修复

落地要点

源头检测

开发测试：将SCA工具对接到DevOps流程里，对编译构建环节卡点，保障软件构建时所依赖组件的安全性，确保不引入存在重大漏洞的组件；使用基于插桩技术的IAST工具，在功能测试的同时，检测是否存在高危漏洞风险，并展示漏洞触发数据流，便于修复指导

出厂免疫

积极防御：针对今后随时可能爆发的未知0DAY漏洞，推荐使用RASP应用自防御能力，针对该类漏洞的攻击利用方式精准有效的防护。它可以通过应用的函数行为分析、上下文情境感知及热补丁技术有效阻断绝大部分RCE类未知漏洞攻击

持续运营

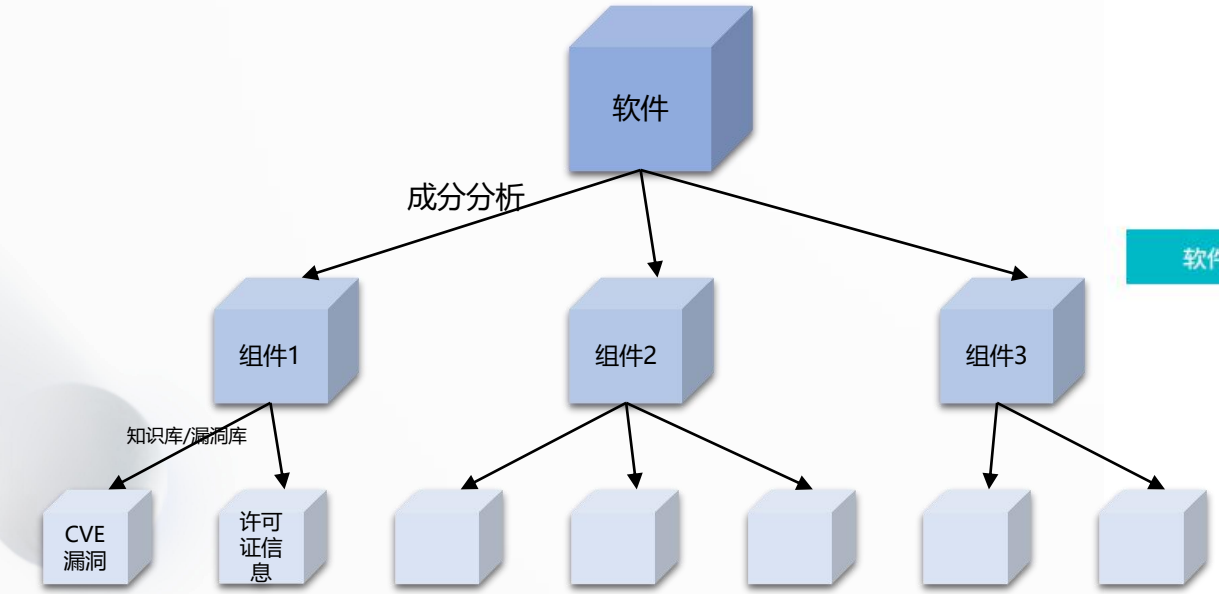
安全运营：常态化使用和运营安全可信的制品库，通过SCA和SBOM持续为每个应用程序构建详细的软件物料清单，全面洞察每个应用软件的组件情况。RASP配合开源漏洞情报，第一时间发现并处理开源漏洞风险

SCA——解决“理不清”

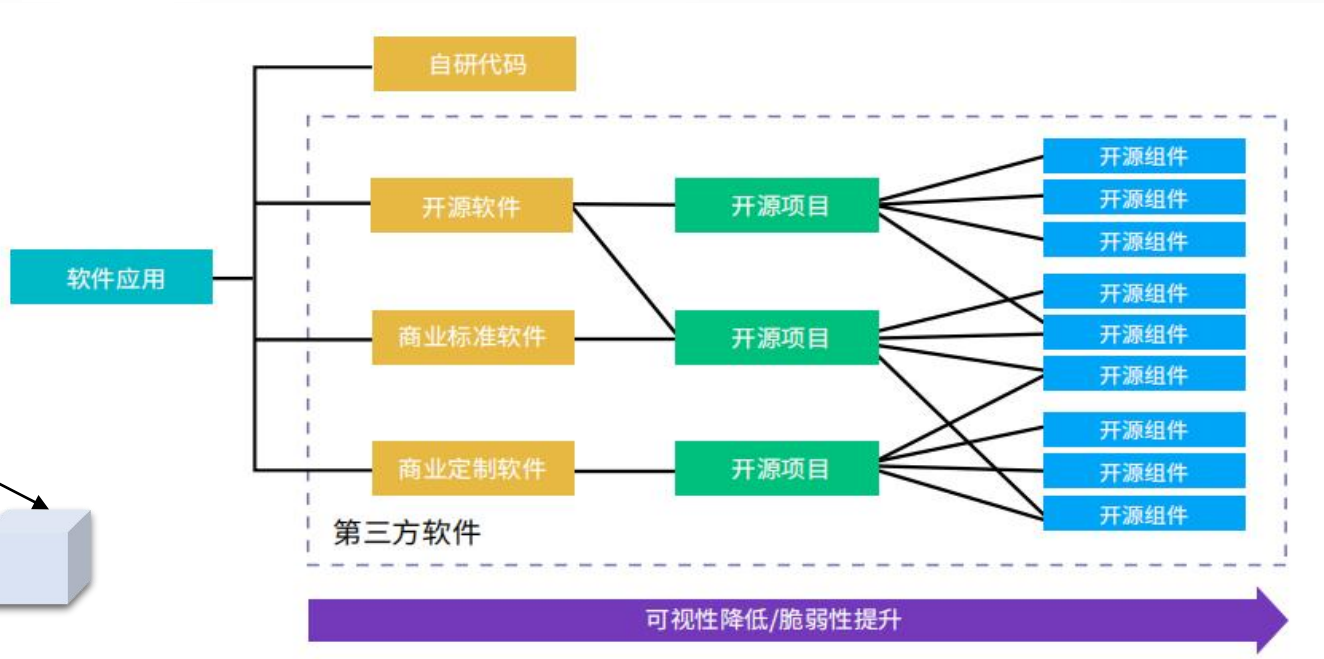
我的软件中包含什么组件和已知风险?

不知道 软件使用了哪些开源组件	不知道 开源组件有哪些风险	不知道 怎么安全使用开源组件

SCA（Software Composition Analysis）软件成分分析，通过检测软件许可证、依赖项以及代码库中的已知漏洞和潜在漏洞来分析开源组件，使 DevOps 能够管理其安全风险和许可证合规性。已经成为安全合规风险管控和安全态势感知必不可少的能力。



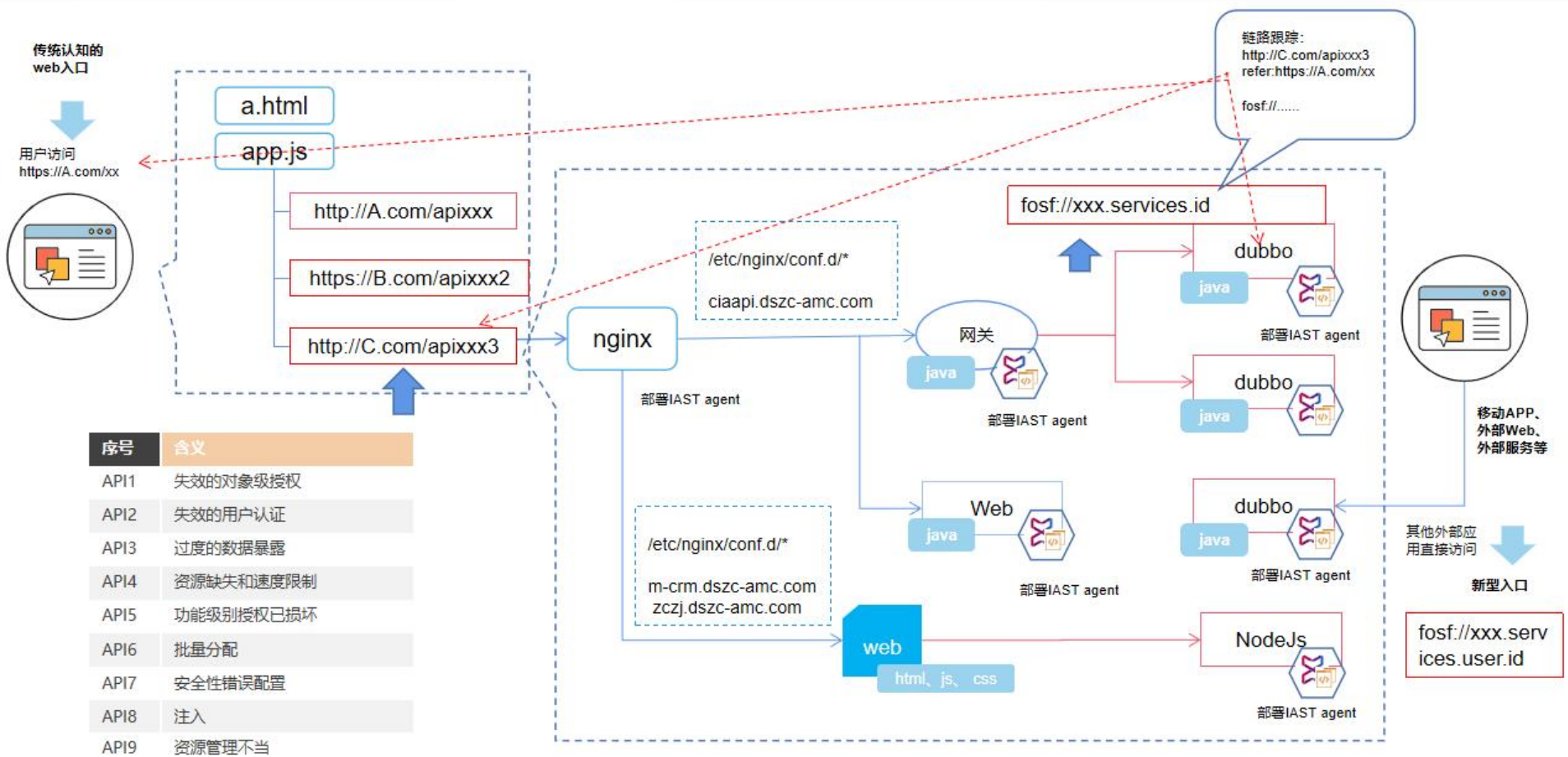
第三方组件是软件的“组成成分”



IAST——解决“看不见”

哪些已知风险是会切实造成危害的?

交互式应用程序安全测试（IAST）是 Gartner 公司提出的一种应用程序安全测试方案。通过代理和在服务端部署的 Agent 程序，收集、监控 Web 应用程序运行时请求数据、函数执行，并与扫描器端进行实时交互，高效、准确的识别安全漏洞，同时可准确确定漏洞所在的代码文件、行数、函数及参数。

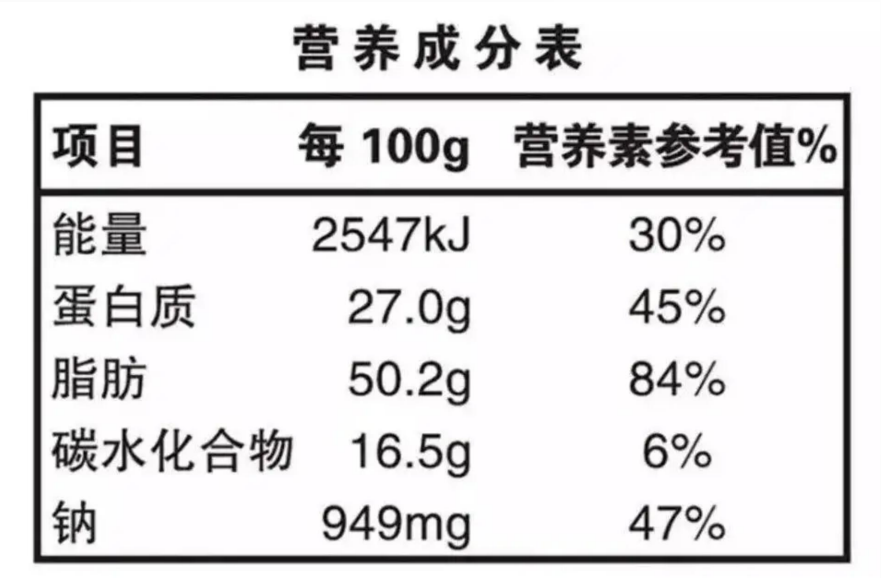


序号	含义
API1	失效的对象级授权
API2	失效的用户认证
API3	过度的数据暴露
API4	资源缺失和速度限制
API5	功能级别授权已损坏
API6	批量分配
API7	安全性错误配置
API8	注入
API9	资源管理不当
API10	日志和监控不足

开源漏洞情报+SBOM——解决“找不到”

哪些应用和组件会受到0day的影响?

- 软件物料清单（SBOM, Software Bill Of Material）是代码库中所有开放源代码和第三方组件的列表。
- SBOM能够列出管理这些组件的许可证，代码库中使用的组件的版本及其补丁程序状态。



营养成分表

项目	每 100g	营养素参考值%
能量	2547kJ	30%
蛋白质	27.0g	45%
脂肪	50.2g	84%
碳水化合物	16.5g	6%
钠	949mg	47%

属性	SPDX	CycloneDX	SWID
作者姓名	(2.8) Creator:	metadata/authors/author	<Entity> @role (tag Creator), @name
时间戳	(2.9) Created:	metadata/timestamp	<Meta>
供应商名称	(3.5) PackageSupplier:	Supplier publisher	(softwareCreator/ publisher), @name
组件名称	(3.1) PackageName:	name	<softwareIdentity> @name
版本字符串	(3.3) PackageVersion:	version	<softwareIdentity> @version
组件哈希值	(3.10) PackageChecksum: (3.9) PackageVerificat ionCode:	Hash “alg”	<Payload>/../<File> @[hash-algorithm]:hash
唯一标识符	(2.5)SPDX Document Namespace (3.2) SPDXID:	bom/serialNumber component/bomref	<softwareIdentity> @tagID
关系	(7.1) Relationship: DE- SCRIBES CONTAINS	(Inherent in nestedas- sembly/subassembly and/or dependency graphs)	<Link> @rel, @href

“到 2025 年，60% 负责开发关键基础设施相关软件的组织将在其软件工程实践中强制使用标准化的 SBOM，比 2022 年（不到 20%）大幅上升。”

——《Innovation Insight for SBOMs》，Gartner

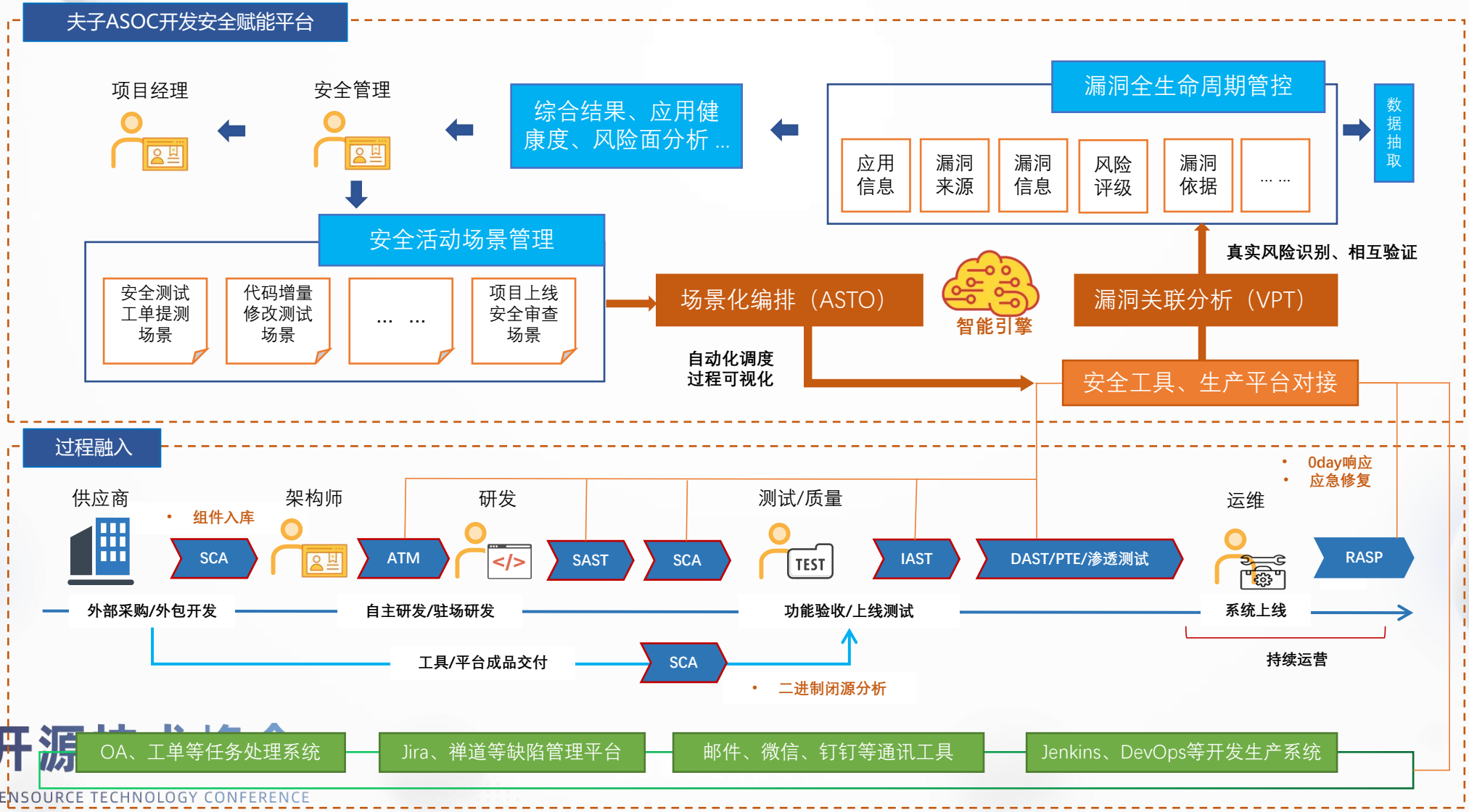
RASP——解决“治不了”

快速消除0day风险面并拦截异常行为

RASP（Runtime Application Self-Protection）是一种“新型应用安全保护技术”，它将保护程序像疫苗一样注入到应用程序中，应用程序融为一体，它可以检测从应用程序到系统的所有调用，能实时检测和阻断安全攻击，使应用程序具备自我保护能力，当应用程序遭受到实际攻击伤害，就可以自动对其进行防御。

ASOC平台技术架构

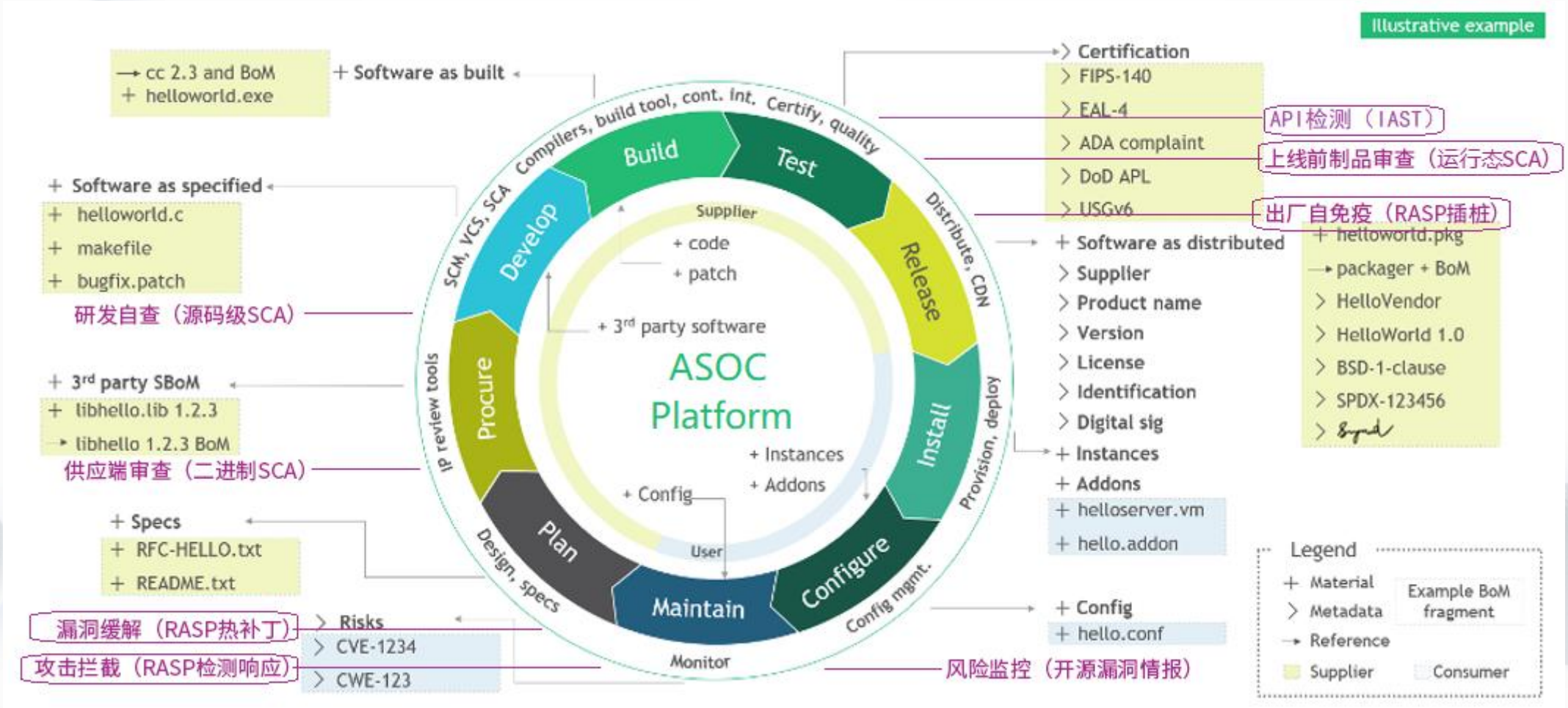
ASOC（application security orchestration and correlation，应用安全编排与关联），消除DevSecOps安全活动冗余工作，安全人员更加聚焦真实问题，以整体视角管理安全开发体系，并加速应用发布效率。




GOTC 2023
全球开源技术峰会
THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE
OPEN SOURCE, INTO THE FUTURE
04 体系构建

基于代码疫苗技术，构建开源应用风险治理体系

SCA+IAST+RASP+漏洞情报



GOTC

THANKS

全球开源技术峰会

THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE