Akamai

白皮书

实现 API 持续安全的 5 个步骤

报告内容

API 安全挑战

应用程序编程接口 (API) 有助于推动创新和提高收入，由这些 API 形成的生态系统不断发展壮大，并融入到企业推出的各项数字或云计划中。但问题也接踵而来，API 已迅速成为一种主要的攻击媒介。

攻击者意识到，API 可能成为直通公司最敏感数据的快捷通道。相关的漏洞很多，包括投入生产环境的 API 往往存在错误配置、缺少身份验证控制措施以及意外暴露在互联网上等等，而所有这些问题都会给攻击者带来可乘之机。

为什么在向最终用户公开发布应用程序之前没有发现和修复这些 API 漏洞呢？让我们来一探究竟并探讨下一步要何去何从。

您可能听说过这样一句话——“企业的发展离不开软件”。对于当今的企业而言，更应该说是“企业中的每个业务部门都是一个独立的应用程序开发商，都在你争我抢地满足客户需求”。这句话听起来有些刺耳，但却是不争的事实。

尽管由集中式 IT 部门进行的传统部署仍然存在，但很多企业都见证了由各业务部门的计划掀起的创新浪潮，并且推动他们创新的不是经权衡的流程，而是迫在眉睫的商业目标。为了快速行动并抓住市场机遇，企业可能无法充分考虑安全影响，从而导致出现所谓的网络安全债务。中心 IT 部门以外的开发人员（有些甚至直接对接业务部门）可能在很短时间内就开发出一款新的应用程序、网站工具和生成式 AI 增强的服务，然后绕过内部控制措施直接上线。在很多情况下，安全团队都无法了解这些项目，因此无法全面评估风险。

多年来，众所周知的攻击媒介一直存在这种变化，导致行业专家迫切希望各公司增强对勒索软件的防御能力、确保密码安全等。但是，很多公司并未将保护 API 视为当务之急。这会导致出现严重问题，因为 API 已融入到企业创建的每个应用程序和每项在线服务之中，它们不断地交换着数据，却很少受到充分的保护。

财务模式

如今，预算已从集中式 IT 支出转为各业务部门的运营开支，但预算流程并没有随之变化以反映安全风险的升高。业务部门可能并不完全清楚应该向安全领域分配多少资金，因此常常导致没有分配专门的预算来确保数据受到保护。

云迁移

应用程序正在迁移到公有云和私有云环境。数据和工作负载的迁移导致复杂性增加、控制力下降，并将第三方引入环境中。企业需要实施额外的安全实践来减少这些风险因素，但可能并不具备有效实施这些实践所需的技术、经验或资源。

容器化

转向微服务会导致攻击面呈指数级扩大。这些实例可能会快速实例化，然后崩溃。这是一个高度动态的环境，而很多企业所依赖的传统工具适合更静态的环境，因此很难确保它的安全。这是 API 无处不在并充满风险的又一佐证。如今基于容器和基于微服务的应用程序架构依赖于多个 API 才能正常工作。即使企业拥有 API 清单，也知道其整个环境中 API 的确切数量，他们往往也不知道哪些 API 会返回敏感数据。

敏捷开发

企业希望开发人员能够迅速地推出新的应用程序、服务和功能，这也会带来很大的风险。为了应对最后期限的压力，开发团队可能会使用持续集成和持续交付 (CI/CD) 方法及自动化功能来进行开发、集成和测试，确保高效完成工作。

但随之而来的风险由谁负责？转向 DevOps 意味着会更频繁地进行超出安全团队控制范围的代码更改。越来越多的企业正在转变为采用左移模式进行应用程序整体开发。这一步在方向上没有问题。但也需要将“尽早测试，频繁测试”的思维方式应用于应用程序内的 API，并且企业还有大量的工作需要做。

您应该从何处着手？持续交付需要持续地保护安全。下面将介绍五个步骤，以帮助您学会在企业持续高速创新的同时如何开始利用全面、不间断的 API 保护功能。

实现 API 持续安全的 5 个步骤

1. 培养持续安全文化

了解和管理 API 安全并非易事。这要求您的领导团队在整个企业中培养安全文化，尤其需要在整个软件开发生命周期中落实安全文化。如果您已在培养这种安全文化方面取得进展，下一步就要利用它解决 API 的复杂性问题以及 API 带来的运营风险。这将提升监测能力、治理和协作能力。

您的企业可以通过以下实用措施来建立和维持长久的安全文化：

- **分散安全团队。**让专家深入到开发小组和产品线中，以提升监测能力和治理能力。根据这些专家提供的背景信息制定更灵活的策略，并且落地实施。
- **确保安全团队参与所有数字发布活动**，不仅仅以制定策略的方式参与，而且从每项服务推出开始就积极参与。业务部门负责人、其团队以及开发人员应与安全人员建立畅通的沟通渠道。
- **指定安全专员。**确定业务部门内的安全倡导者，以帮助建立和维系对高效开展工作至关重要的关系。指定专门的安全专员有助于不断强化安全信息，并让跨职能团队相互负责。
- **让每个人都参与其中。**安全培训必不可少，并且不仅仅面向开发人员和工程师。参与软件开发过程及其他工作的每个人都应该参加安全培训。

2. 评估 API 安全态势

很多企业都低估了其 API 资产的规模。拥有清单的企业可能不仅可能遗漏大量的 API，而且可能不知道哪些 API 会构成高风险。通过创建完整、准确的清单，您便可以评估整个 API 攻击面。

以下建议将帮助您全面了解自己的 API 安全态势：

- **建立完整的清单。**清晰、准确地了解贵企业的潜在风险敞口以及它在所有 API 和 Web 应用程序中的真实情况。使用 API 发现工具，全面查找所有 API 并将其加入清单中，包括：
  - 影子 API
  - 僵尸 API
  - 休眠 API
- **识别每个 API 及其风险。**了解与每个 API 交互的敏感数据的类型、每个 API 的路由方式、关联的物理资源以及它所属的业务部门或应用程序。
- **检查安全团队资源分配。**分析每个 AppSec 团队成员必须维护的 API 的数量。确定是否需要更多的技术或培训以保持或改善态势。

3. 修复、自动化和集成

企业需要了解 API 访问权限、使用情况和行为。但 API 很复杂，难以进行分析。了解 API 安全形势的复杂性通常需要完成解析日志、提取目录数据、审查配置、测试安全性以及评估设备配置等过程。如果没有适当的工具，修复工作可能会非常繁重，因为这项工作在技术上具有挑战性或者需要花费大量的时间和精力。但是，修复往往能够以自动化或半自动化方式完成，从而消除已知的漏洞并减轻直接风险，之后便不需要或很少需要人工干预。

下面几点可帮助您防范攻击并解决配置错误的问题：

- **与现有 IT 工作流程管理系统相集成。**您需要确保在发现问题后将它们分配给适当的团队进行处理。集成应触发自动化工作流程来解决企业内的所有 API 问题。
- **分阶段进行自动修复。**最初，在执行新的修复操作之前，都需要依靠人工来审批这些操作。确保与各业务部门相互配合以实现半自动化的修复。只向开发人员抱怨代码很糟糕并不能解决问题。他们需要具有指导性和有用的见解。否则，您就是在浪费自己和开发人员的时间。当发现问题反复出现时，可采用全自动化来加快修复速度。
- **监控恶意行为。**利用以往关于 API 利用手段的知识来确定异常行为，从而发掘攻击者的意图。采用自动化或半自动化响应来抵御攻击。
- **与现有的安全信息与事件管理 (SIEM) 系统相集成。**此集成可确保规模较大的团队能够使用 API 安全数据。

4. 将 API 安全防护左移

就 API 开发而言，不仅要进行测试，还要*考虑何时*测试 API。传统模式会将测试安排在更靠近部署阶段的位置，这很重要但远远不够，并且可能会导致出现严重漏洞。“左移” 是一种将各种任务移至开发过程早期阶段的方法。通过将安全和测试融入 API 开发中的每个步骤，左移方法可以确保开发人员能够在 API 的整个生命周期内进行漏洞监控。这使企业能够夯实 API 安全根基，然后在此基础之上加快创新速度和增强竞争优势。

以下建议有助于您顺利采用左移 API 测试：

- **明确目标。**由于左移需要进行企业和文化变革，因此管理层应该先明确测试过程的目标，以确保引入开发周期的任何新工具或流程都适用于该团队现有的开发和测试方法。
- **了解供应链。**先清楚您的企业如何以及在何处开发应用程序和软件，然后再制定全面的左移安全计划。供应链的安全风险状况主要取决于供应链中其他各方的安全能力。这也有助于您的开发人员确定可能需要将测试更早地设在生命周期的哪个阶段。
- **自动化安全流程。**当开发团队启动微服务时，需要确保团队内安插的安全专家从一开始就使用 API 安全工具来帮助监控随容器化出现的风险。
- **使用一致的工具。**建议安全团队采用开发团队的主要界面，并调整为开发人员首选的工具、工作台和语言。例如，和常规的用户案例一样，可以将漏洞和发现输入到新应用程序功能要求的产品待办事项中。
- **让 AppSec 团队成为创新源泉。**利用持续交付原则开发可以抵御风险的安全特定微服务，为您的企业树立差异化优势。

5. 持续测试

正如我们刚才确定的，由于左移安全方法会将测试移到时间表的左侧，因此团队可以在生命周期的更早阶段执行测试。相比之下，右移方法要利用真实用户和场景进行测试，这在开发环境中无法实现。右移测试可以在生产环境中进行测试，并通过收集应用程序用户的反馈和评价来改善用户体验，从而确保现实世界中的软件稳定性和性能。事实上，这两种方法各有优缺点。为了最大限度地降低潜在风险，企业需要进行持续测试。

您的企业可以使用以下提示来培养和保持长久存在的安全文化：

- **主动进行 API 测试。**在 API 软件开发生命周期中，应当通过 API 安全测试修复生产前和生产后的所有潜在问题。每个 API 在部署前后都要验证其完整性。
- **持续监控 API 流量。**跟踪 API 使用并分析 API 流量元数据。实时流量分析可识别新的 API 和现有 API 中的变化。该分析过程必须自动完成、可重复且可操作。
- **监控漏洞和错误配置。**测试应当随着开发活动持续、同步地进行，而且需要客户、开发人员和测试人员之间保持持续沟通。它需要识别问题，以便在问题被利用之前进行修复。分析上的延误会让黑客有更多时间来利用漏洞。还有一点非常重要，务必及时报告策略或功能变更并更新 SIEM 系统。
- **对 API 流量进行记录。**一定要对 API 流量进行记录，以在需要针对特定 API 密钥、令牌、IP 地址和用户身份创建取证报告时参考。

总结

对于很多企业来说，API 安全威胁是真实存在的危险。API 往往不受管理，是传统工具监测之下的漏网之鱼，并且一直存在错误配置、缺少身份验证和代码编写错误等问题。这导致不受管理的 API 成为攻击者的主要目标，并可能出现 API 已被入侵而企业未能觉察的情况。

解决此问题的方法是持续提供安全保护，并在 API 创建和进入生产环境时将安全机制融入到开发人员的流程和 API 本身中。企业应该牢记以下四条最佳实践：

1. 培养让 AppSec 专家深入企业工程团队的新型文化
2. 发现 API 的完整清单，以掌握企业的 API 安全风险状况
3. 确定修复措施的优先顺序，尽可能自动完成修复，并将 API 安全无缝集成到当前的应用程序安全系统中
4. 保持持续警惕并进行持续 API 测试，以确保快速识别并抵御新的漏洞

虽然这种方法需要新的思维方式、不同的流程和跨团队协作，但这些都是可以克服的挑战。

阅读更多内容，详细了解 API 攻击方法、常见 API 漏洞以及如何保护您的企业。

预约定制化 Akamai API Security 演示，了解我们如何为您提供帮助。

Akamai

Akamai Security 可为推动业务发展的应用程序提供全方位安全防护，而且不影响性能或客户体验。诚邀您与我们合作，利用我们规模庞大的全球平台以及出色的威胁监测能力，防范、检测和抵御网络威胁，帮助您建立品牌信任度并实现您的愿景。如需详细了解 Akamai 的云计算、安全和内容交付解决方案，请访问 akamai.com 和 akamai.com/blog，或者扫描下方二维码，关注我们的微信公众号。发布时间：2024 年 10 月。

扫码关注 - 获取最新云计算、云安全与CDN前沿资讯