DevSecOps的六大支柱：测量、监控、报告和行动

CSA GCR cloud security alliance GREATER CHINA REGION

CSA cloud security alliance

DevSecOps 工作组的官网地址是

https://cloudsecurityalliance. org/research/working-groups/devsecops/

云安全联盟

创立于2009年，作为世界领先的独立、权威国际产业组织，致力于定义和提高业界对云计算和下一代数字技术安全最佳实践的认识和全面发展。

云安全联盟大中华区

在香港注册并在上海登记备案的国际NGO组织，旨在 立足中国，连接全球，推动 中国数字安全技术标准与产业的发展及国际合作。

4 大区
大中华区、美洲区、欧非区、亚太区

180+ 分会
英国、法国、加拿大、旧金山、马来西亚等覆盖50多个国家和地区

2.5K+ 成员单位
世界500强科技公司、安全厂商、中小型企业、研究机构

20W+ 社区专业人员
研究工作组专家、社区志愿者、从业人员、CSA认证学员

前沿研究

#云安全 #AI安全
#零信任 #数据安全
#5G安全 #区块链安全
#量子安全 #物联网安全
#金融安全 #医疗安全
#智能座舱安全
#关键基础设施安全
.....

培训与认证

CCSK 云安全知识认证
CDSP 数据安全认证专家
CAISP人工智能认证专家
CZTP 零信任认证专家
CCPTP 云渗透测试认证专家
CCAK 云计算审计知识认证
.....

会议活动

CSA summit@RSAC
CSA GCR Congress
CSA研讨会
AI For GOOD峰会
.....

评估与认证

AI STR AI安全、可信、负责任认证
STAR 云安全评估认证
CAST 云应用安全可信认证
CNST 云原生安全可信认证
.....

1000+研究成果 **10W+认证学员** **1000W+传播量**

成员单位（部分）

ZEEKR Google Microsoft amazon HUAWEI intel IBM Security cisco ORACLE Meta 阿里云
UCLOUD优刻得 京东 SHECA 上海CA中心 移动云 天翼云 中国电子云 CECLOUD China unicom 中国联通 中国移动 China Mobile 中国电信 CHINA TELECOM ICBC 中国工商银行 LEXIN乐信

企业合作微信号:csagcr 认证培训微信号:CSAlynn 邮箱:info@c-csa.cn

致谢

中文版翻译专家

翻译组成员：

高亚楠　伏伟任　卜宋博　贺志生　江　澎　林艺芳　谢绍志

王　彪

审校组成员：

李　岩　王彪

（以上排名不分先后）

英文版编写专家

主要作者：

Dan Gora

Roupe Sahans

贡献者：

Alex Brown

Daniel Parkin

Michael Roca

审阅者：

Roland M

Julianna Tchebotareva

Timothy Thatcher

Udith W

CSA 分析师：

Josh Buker

CSA 全球员工：

Claire Lehnert

在此感谢以上专家及单位。如译文有不妥当之处，敬请读者联系CSA GCR秘书处给予雅正！联系邮箱research@c-csa.cn；国际云安全联盟CSA公众号。

目录

前言

云安全联盟和 SAFECode 都致力于改善软件安全成状况。2019 年 8 月发表的论文《DevSecOps 的六个支柱》提供了高端方法论和成功实施解决方案，作者们使用这些方法和解决方案可以快速构建软件，并最小化安全相关错误数量。这六个支柱是:

支柱 1：集体责任（2020 年 2 月 20 日发布）

支柱 2：协作与集成（2024 年 2 月 21 日发布）

支柱 3：务实的实现（2022 年 12 月 14 日发布）

支柱 4：建立合规与发展的桥梁（2022 年 2 月 8 日发布）

支柱 5：自动化（2020 年 7 月 6 日发布）

支柱 6：测量、监控、报告和行动（2024 年 5 月发布）

支撑这些支柱的成功解决方案是云安全联盟和 SAFECode 一系列更详细的联合出版物。本文是六个后续出版物的最后一篇。

介绍

DevSecOps 举措的实施和维护可能需要几个月到几年的时间。在考虑人员、流程和工具的大规模变化时，持续测量 DevSecOps 的成功和失败是关键的差异化因素。“你无法管理你不能测量的事物”这句话十分正确，没有可操作的测量标准和可观察性去测量绩效，就无法理解进步，无法复制成功，也无法意识到失败。

测量、监测、报告和行动的能力是任何成功的安全计划的基本特征，以支持有效的决策。在本文中，我们探讨:

- **使数据可观察:**将安全数据和指标转化为可观察的数据
- **基于场景的修正:**安全可观察性概念应用于高绩效和低绩效场景案例
- **通过报告改进:**安全可观察性报告的起点

目标

云安全联盟的 DevSecOps 工作组(WG)在“DevSecOps 的六个支柱”1 中发布了高级别指南，倡导采用新的安全方法。这六个支柱被认为是任何希望实施 DevSecOps 的组织需要重点关注的领域，其中一个支柱是"支柱 6:测量、监测、报告和行动"。

支柱 6 的目标是促进和展示 DevSecOps 安全状况的确切测量。这将使安全和数字领导者能够确定其安全实践的有效性，以及安全如何融入软件开发生命周期。

目标读者

本文件的目标读者包括从事信息安全和信息技术管理和业务职能的人员。其中包括 CISO、CIO、CTO 以及参与以下职能领域的个人:平台工程、DevOps、产品团队、架构、信息安全、治理和合规。

使数据可观察

如果一个系统具备有效且可见的系统状态数据，那么它就是可观察的，这对于追根溯源至关重要。一个系统现状的可观察性测量基于其生生成的数据，如日志、指标和跟踪状态。可观测性旨在了解整体环境中发生的情况，以便检测和解决问题，从而保持系统的高效和可靠性。组织采用可观察性来帮助检测和分析运营、软件开发生命周期、应用程序安全和最终用户体验中事件的影响程度。

日志、指标、分布式跟踪和用户体验的测量是实现可观测性成功的关键支柱:

- **日志**:特定时间发生的离散事件的结构化或非结构化文本记录

- **指标**:是指计数或度量的值，通常在一段时间内进行计算或聚合。指标可以来自各种来源，包括基础设施、主机、服务、云平台和外部来源

- **分布式跟踪**:事务或请求在应用程序中流动的活动，并显示服务如何连接，包括代码级别的详细信息

- **用户体验**:在应用程序上，甚至在生产前环境中，添加具体的、由外向内用户视角的数字化体验，扩展传统的远程可观测性。

脆弱性的可观测性

一旦产品团队具备基于正确工具和工作流程的扫描机制，如静态应用程序安全测试(SAST)、动态应用程序安全检测(DAST)和软件组成分析(SCA)等，识别脆弱性将十分简单。而挑战始于团队持续性开展扫描、修正和报告。

随着时间的推移，脆弱性积压可能会增加，在某些情况下甚至会增加到数以万计，直到变得难以管理和补救。有效的脆弱性管理程序应该在开发生命周期的早期发现并修复漏洞，以减少其平台/产品的总体风险暴露。脆弱性的可观察性要求识别以下三个属性。

1. 平均识别时间

MTTI 是识别脆弱性所花费的时间。较高的 MTTI 表明，在开发生命周期后期发现的脆弱性，将增加补救成本和将问题引入生产环境的风险。通过及早识别脆弱性，开发人员可以防止可利用脆弱性引入生产环境。

在图 2 中，识别时间为脆弱性暴露窗口的开始日期(1 月 10 日)和识别所述脆弱性的日期(1 月 16 日)，因此，识别所需时间为 7 天。对数百个脆弱性进行识别的时间会有所不同，MTTL 是所有漏洞识别值的平均值。

2. 平均补救时间

MTTR 指示识别后修复脆弱性所花费的平均时间。MTTR 有助于跟踪脆弱性是否被识别，是否被解析器团队“遗忘”。与 MTTI 一样，MTTR 可以根据严重性和接受风险的脆弱性数量进行跟踪

图 2 中，补救时间是识别日期(1 月 16 日)和补救日期(2 月 10 日)的差值，因此，补救所需时间为 26 天。与 MTTI 一样，对数百个脆弱性进行补救的时间不同，MTTR 是所有漏洞补救值的平均值。

3. 发展速度与脆弱性趋势相对应

开发速度是否超过了修复脆弱性的能力?跟踪这一趋势有助于确定开发团队是否在开发和运营过程中难以跟上脆弱性补救的步伐，以及补救是否与发展速度保持同步。MTTI/MTTR 可以根据发展速度和严重程度评分进行跟踪。这可以帮助确定开发团队和安全团队在脆弱性管理中是否存在脱节。

在图 2 中，脆弱性暴露窗口(1 月 10 日-2 月 10 日)跨越了两个发布窗口。此值将有助于提供上下文信息，以帮助证明 MTTI 和 MTTR 值的正确性。

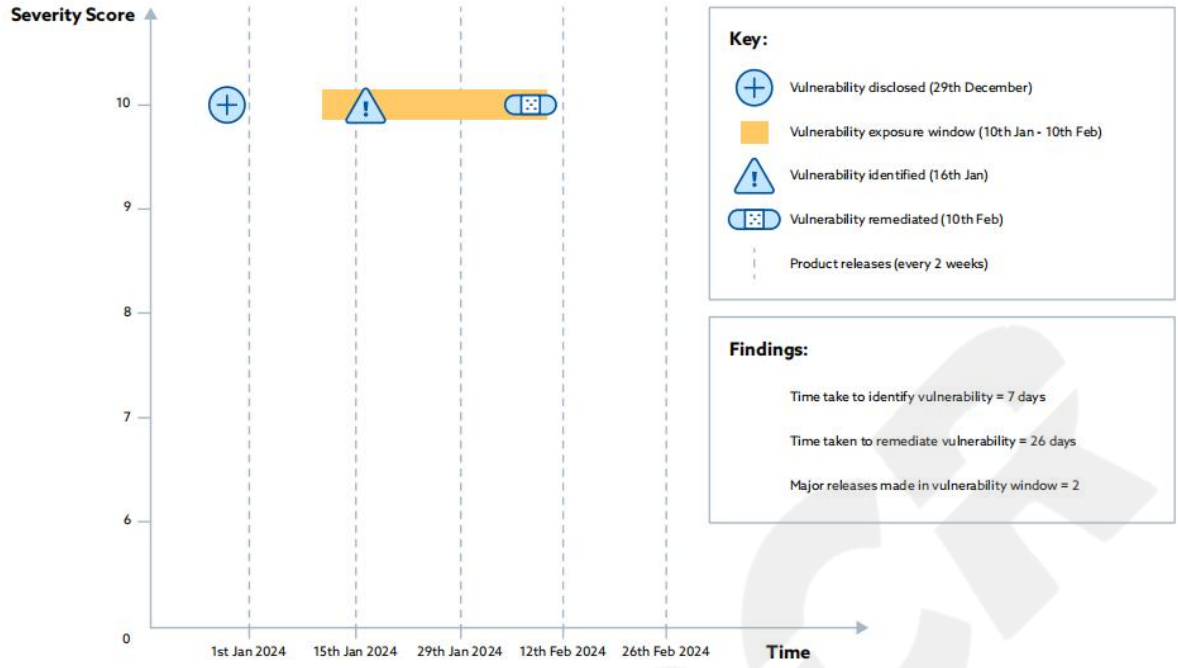


Figure 2: Metrics for vulnerabilities

安全架构观察

以下可观察因素旨在指导开发团队开展可接受程度威胁建模工作及其补救活动。组织可以利用这些改进他们的威胁建模模式和形式。

4. 威胁量

跟踪从威胁建模工作中识别出的威胁数量、威胁类型和相应的控制。将有助于了解是否采用了纵深防御方法。

图 3 演示了对 STRIDE(欺骗、篡改、抵赖、信息泄露、拒绝服务、特权提升)应用于每个威胁主题的控制量。深度防御平均值是每个 STRIDE 威胁类型的控制覆盖率平均值。可以观察到，本例中的信息披露是最普遍的威胁，因为它所映射的控制量最高。

5. 计量控制

针对从威胁建模工作中识别到的威胁，跟踪相应的控制措施。这有助于了解补救活动是否按规模进行了优化。例如，单个控件具有多个用途。

图 4 演示了通常从威胁模型中识别出的控件的示例集，以及每个控制用于解决威胁的用法。这衡量了每个控件的投资回报率(ROI)。在本例中，基于角色的访问控制(RBAC)和安全监控这两项控制所能处理的威胁最多，投资回报率最高，这提示了应该优先考虑相关控制措施。而每个控制措施可平均处理 5 到 6 个威胁用例。

6. 计量安全措施

为应对已识别的脆弱性/威胁而转换为安全措施的控制量。这有助于了解环境是否得到了保护，以防止风险、威胁或脆弱性发生。

图 5 显示了随着时间推移的安全措施的实施情况，强调了针对威胁状况采取预防性安全措施的重要性。

7. 威胁模型燃尽

衡量对威胁的控制状况是威胁建模的一项关键输出。通过跟踪已识别状况与当前实施状况，团队可以评估威胁建模的有效性，验证威胁建模对开发过程的价值赋能状况。

图 5 显示了威胁模型的燃尽情况，通过实施控制，在 5 个冲刺阶段(10 周)内将 25 个未解决的威胁减少到 10 个。这种类型的衡量将证明威胁模型实施是否成功。在这个例子中，在 10 周内实施的控制措施中，有 60%是良好的做法。

8. 采用安全模型

分析安全设计模型的实施情况是评价安全架构成熟度的衡量方法。安全模式是一种可重用的解决方案，可解决软件或系统设计中的常见安全挑战和漏洞。它可以是一份工作文档，也可以是一段可重复使用的代码(模板)。

通常情况下，一个更成熟的组织会有更多的模型和模板，以确保遵循一致的设计/工程实践。完善的模型有助于提高重复性和一致性，减少重复出现的弱点和漏洞的数量。缺乏采用可能表明团队没有意识到这些模型的存在，或者模型并不有效。图表 6 中，模型/模板数量为 20 个，采用率仅为 70%，表明六种模型存在效率低下的问题。

9. 威胁场景与网络风险映射

计量威胁，为风险声明提供说明，即有多少威胁与某个风险声明相对应。与风险不同，威胁是无法缓解的；但是，威胁是可以应对的，并可用于提供风险信息。

图表 6 展示了这一指标。在该示例中，敏感数据暴露风险与 14 个威胁有关；所以，从理论上讲，处理这 14 个威胁能降低敏感数据暴露的风险分数。

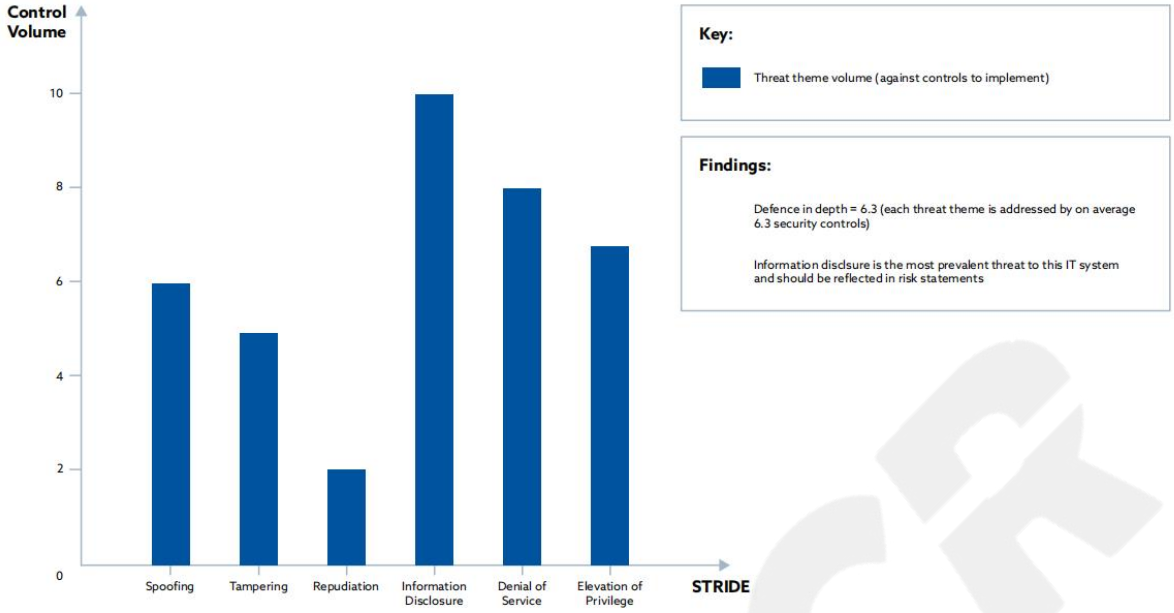


Figure 3: Metrics for control implementation across threat themes

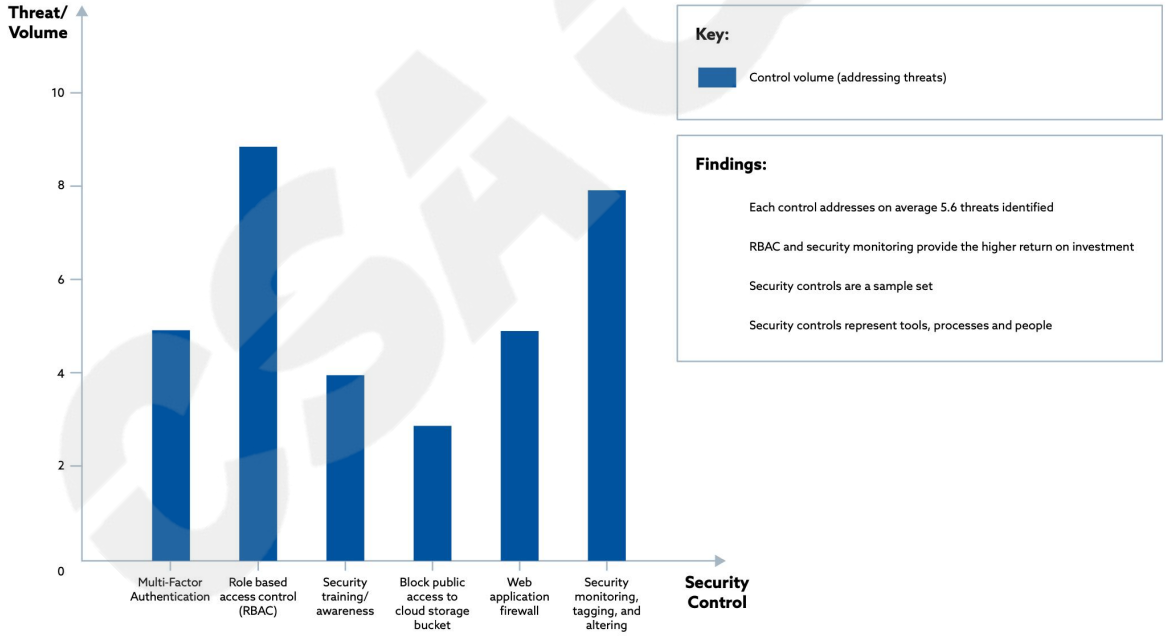


Figure 4: Metrics for security control reuse to address threats

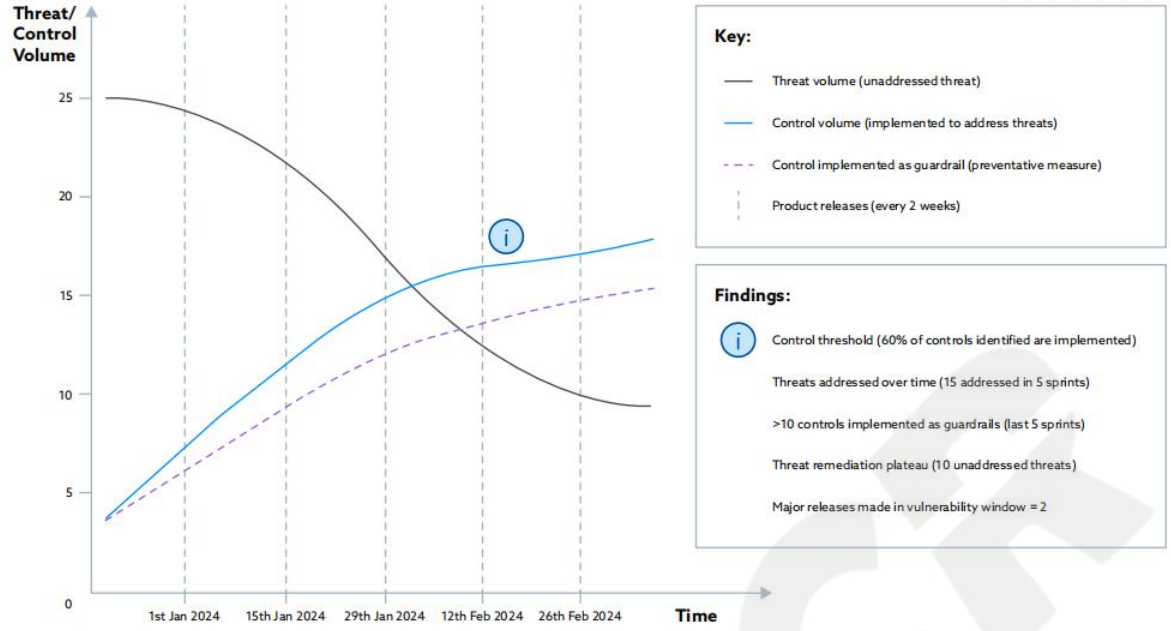


Figure 5: Metrics for control implementation against threats

其他指标：

共有 31/34 个威胁与 3 个风险声明有关联，平均每个风险有 11.3 个威胁。3 个威胁因不属于业务风险而未被计算。

风险 1："敏感数据暴露风险"被 14 个威胁中提及

风险 2："数据完整性受损"被 6 个威胁中提及

风险 3："关键 IT 服务不可用"被 11 个威胁中提及

模型/模板数量为 20 个，整个项目的采用率为 70%。表明有 6 个模式/模板需要修改或删除。

事件响应的可观察性

事件响应的可观察性提供了对警报和检测的洞察力，进而可进行更详细、更翔实的事件后审查。

平均检测时间（MTTD）、平均遏制时间（MTTC）和平均恢复时间（MTTRN）是关键指标，可让企业在安全漏洞导致故障发生时迅速采取行动。对 IT 产业的安全监控可以让企业捕捉到这些指标，让运营团队在事故期间获得必要的数据。

10.平均检测时间（Mean time to detect MTTD）

MTTD 是指团队发现潜在安全事件所需的时间。MTTD 高的组织由于不能及早发现事件，会带来额外的不必要风险。早期检测（识别）意味着团队可以尽早专注于响应和恢复工作，减少对生产的影响。

在图表 7 中，检测（识别）时间为暴露窗口的开始日期（12 月 31 日）和确定所述事件的日期（1 月 1 日），因此识别时间为两天。在数百个事件中，识别/检测时间会有所不同，MTTD 是所有事件检测值的平均值。

11.平均遏制时间（Mean time to contain MTTC）

MTTC 指跟踪控制安全漏洞/事件所需的时间。这有助于了解应对安全事件的速度。遏制是指团队可以开始限制和减少事件的影响；避免对依赖服务造成影响或避免漏洞的横向移动。

在图表 7 中，遏制时间是从检测日期（1 月 1 日）到遏制日期（1 月 6 日），因此遏制时间为 6 天。在数百起事件中，遏制时间会有所不同，MTTC 是所有事件控制值的平均值。

12. 平均恢复时间（Mean time to restore normality MTTRN）

MTTRN 表示事件修复过程中恢复正常所需的平均时间。较慢的 MTTRN 会导致服务中断，影响组织的运作能力或产品的有效工作能力。在某些情况下，较长的 MTTRN 意味着，在恢复带有敏感数据服务时出现了疏忽。

在图表 7 中，恢复正常所需的时间从检测日期（1 月 1 日）开始，到补救所述事件的日期（1 月 25 日）结束，因此恢复正常所需的时间为 25 天。在许多事件中，恢复正常的时间会有所不同；MTTRN 是所有恢复值的平均值。

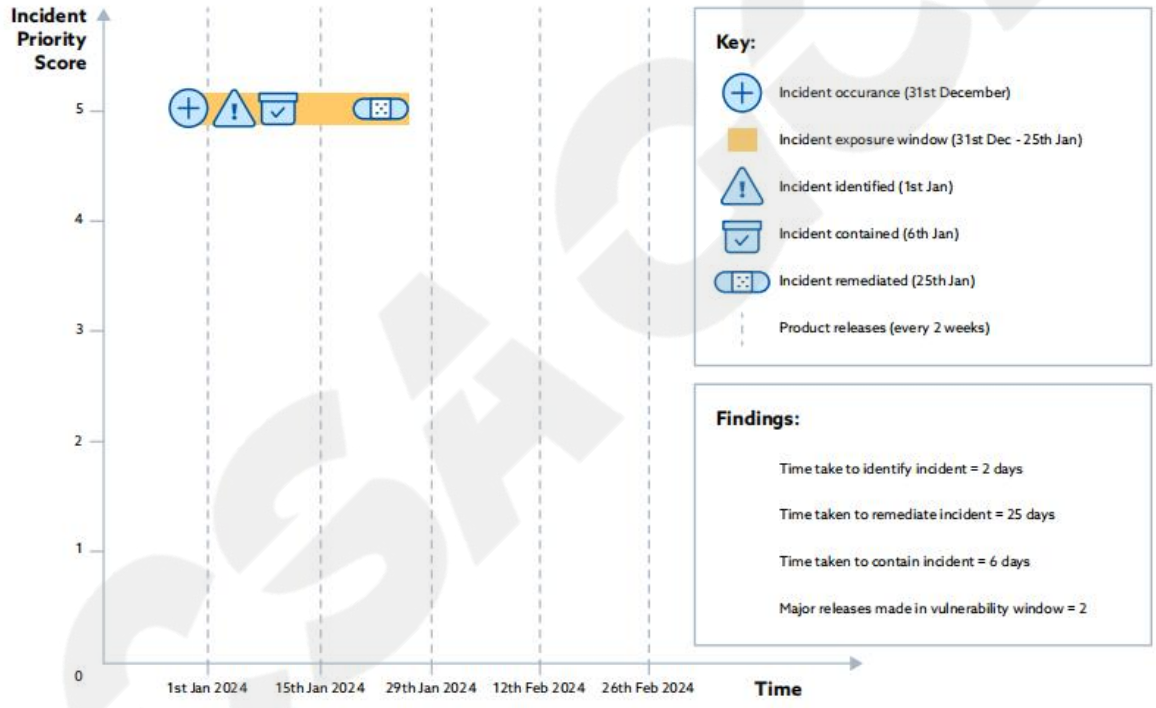


Figure 7: Metrics for incident response

团队间的成熟度比较

理解 DevSecOps 实践对团队效能的影响，需要一种全面的方法来衡量、监控、报告以及根据指标采取行动。

为了提供不同实践成熟度和效能的概览，描述了三个在不同组织中处于不同成熟度水平的团队及其对指标影响的示例情景。情景对比采用了 OWASP 的 DevSecOps 成熟度模型（DSOMM）。团队分别被标记为 Alpha、Beta 和 Charlie，对应低、中、高成熟度及效能水平。通过情景对比，我们获得了这些实践在实际应用中的洞见，以及它们对 DevSecOps 实践的影响。关注点并不在于将团队标定为‘好’或‘坏’，而是突出这些实践在可观察指标以及整体成熟度和效能上的不同影响。

团队“Alpha” – 低成熟度和效能

Alpha 团队在 DevSecOps 实践方面处于较低的成熟度水平。这个团队在 DSOMM 的第 1 至第 2 级之间运作，对相关的 DevSecOps 实践具有基础的理解和采纳。Alpha 团队仍处在适应 DevSecOps 的早期阶段。团队在修复可被利用的漏洞方面反应迟缓，并且在确保设计中内置安全性的方法上缺乏一致性。延迟的事件处理凸显了对更好安全实践和培训的需求。

团队“Beta” – 中等成熟度和效能

Beta 团队在 DSOMM 的第 2 至第 3 级之间运作，展现了他们在 DevSecOps 实践中的中等成熟度。尽管他们已经超越了 DevSecOps 的基础阶段，但他们处理漏洞的方式可以被描述为适度的反应式，而非主动式。在设计阶段嵌入安全性的承诺显示出一定程度的不一致性，偶尔会导致潜在威胁被忽视。此外，虽然他们对事件的响应速度明显快于该领域中成熟度较低的团队，但仍存在改进的空间。

团队“Charlie” – 高成熟度和效能

Charlie 团队在其 DevSecOps 实践方面展现出高级别成熟度和效能，达到了 DSOMM 的第 3 和第 4 级。各项指标体现出高度的 DevSecOps 成熟度。Charlie 团队的实践与指标不仅反映出高水平的技术专业能力，还体现了一种将安全放在首位并在每一个步骤中加以整合的文化。这使得安全性得以在所有阶段中持续嵌入，从而减小了攻击面并提升了安全姿态。尽管事件响应既高效又及时，Charlie 团队仍然保持着持续改进的心态。这保证了事件响应能够适应不断变化的安全威胁。

三个团队之间的安全可观察性

通过对比场景，性能成熟度的影响变得显而易见。这些场景强调了持续测量的重要性，它为明智的决策、成功复制实践以及识别需要改进的领域奠定了基础。它们最终成为组织提升其 DevSecOps 实践和文化的指南。

跨团队的漏洞可观察性

图 8 证实了使用经过的天数对 MTTI 和 MTTR 进行平均和评分的概念，展示了随着时间的推移而取得的改进。**图 9** 代表随时间变化的团队间比较，显示了团队表现的高低。

Alpha 团队的低成熟度实践导致大量漏洞被利用，而 Beta 团队的中等成熟度实践使漏洞数量减少。Charlie 团队凭借其高成熟度实践，拥有最少的可利用漏洞。

MTTI：在新漏洞出现后，Alpha 团队平均需要 20 天才能识别出一个新漏洞。由于建立了每周定期的漏洞扫描，Beta 团队将识别新漏洞的时间缩短到了 15 天。然而，Beta 团队实时扫描生产代码库，不会让相应的漏洞出现在生产环境中。由于持续集成和持续部署（CI/CD）实践，在开发测试环境的源代码库中使用自动化安全扫描，Charlie 团队能够在几天内以较低的 MTTI 检测到漏洞。

MTTR：一旦漏洞被识别出来，Alpha 团队的漏洞往往得不到解决，因此 MTTR 会随着时间的推移而增加。Beta 团队已经建立了一个缓解流程，由于难以及时解决已识别的漏洞，MTTR 降低并稳定在 15 天。在采取有效的补救措施，如开发人员会专门花时间进行漏洞加固，Charlie 团队的 MTTR 随着时间的推移而持续下降。

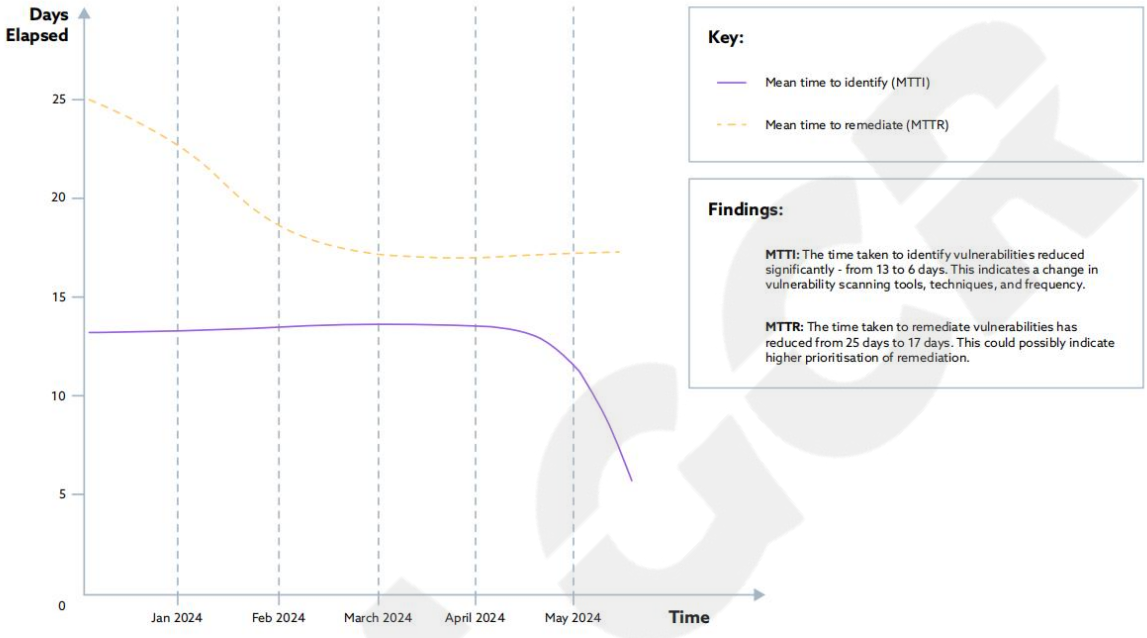


Figure 8: Conceptual observability for exploitable vulnerabilities

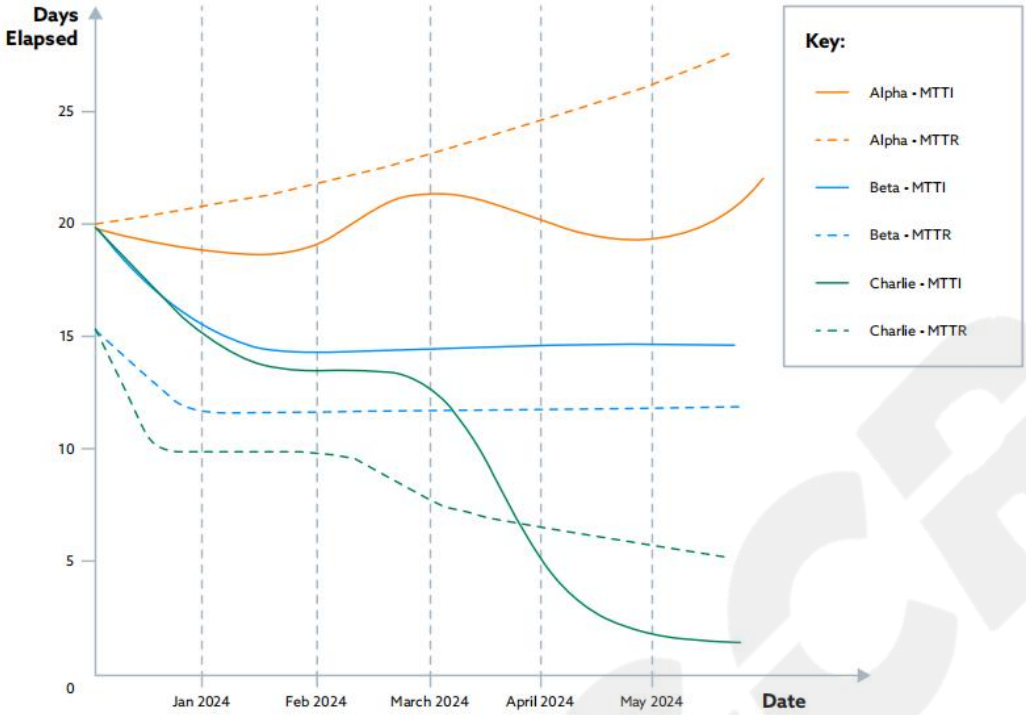


Figure 9: Comparison between teams for exploitable vulnerabilities

跨团队的安全架构可观察性

图 10 证明了通过平均深度防御和控制重用得分的概念，并展示了随时间的改进。**图 11 和图 12** 代表了跨团队随时间的比较，为实施和威胁消减提供了高和低表现的指示。

Alpha 团队缺乏正式的安全架构实践，导致控制措施难以落实，而 Beta 团队的中等实践显示出逐渐改进。Charlie 团队的最佳实践使安全架构可观测性水平高，控制措施在短时期内落实到位。

防御深度和控制复用：Alpha 团队在多次风险评估中经常发现相同的威胁，这表明重复的安全威胁没有策略进行有效缓解。虽然 Beta 团队在评估期间在识别各种威胁方面取得了进展，但他们威胁建模练习的延迟有时会导致漏洞被忽视。Charlie 团队利用一个明确的威胁建模过程来预测和缓解各种威胁，该过程可以在其他开发团队中推广使用，并且定期的回顾会议确保团队从过去的威胁场景中学习并增强其威胁缓解策略。

威胁和控制量：Alpha 团队在设计阶段确定了必要的安全控制，但在生产过程中只实施了其中的一小部分。例如，虽然他们认识到加密的必要性，但只在存储过程中实现加密，在数据传输过程中却没有实现。Beta 团队经常意识到集成安全控制的重要性，例如在存储和传输级别都进行加密。然而，识别和实施之间的延迟，意味着并非所有已识别的控制措施都会立即嵌入到应用程序中。Charlie 的安全控制得到了迅速实施。例如，在确定端到端加密的需求后，团队通过自动化部署脚本来管理、强制执行安全配置，确保它在存储、传输和处理过程中都得到应用。

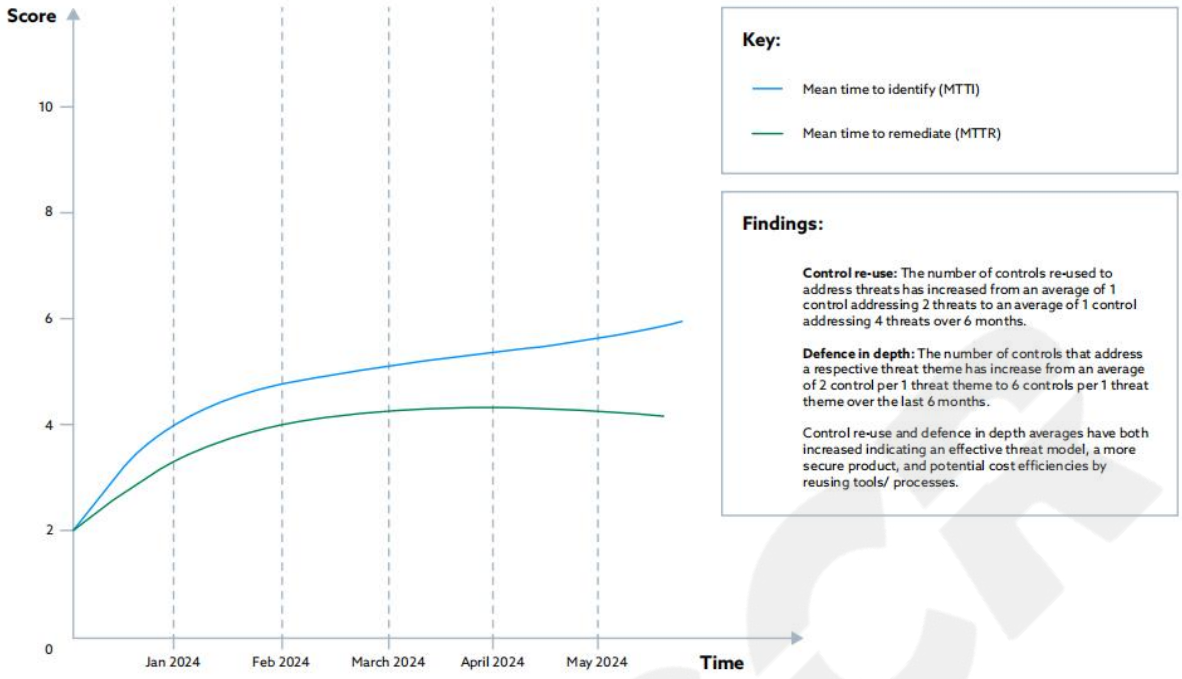


Figure 10: Conceptual observability for security architecture

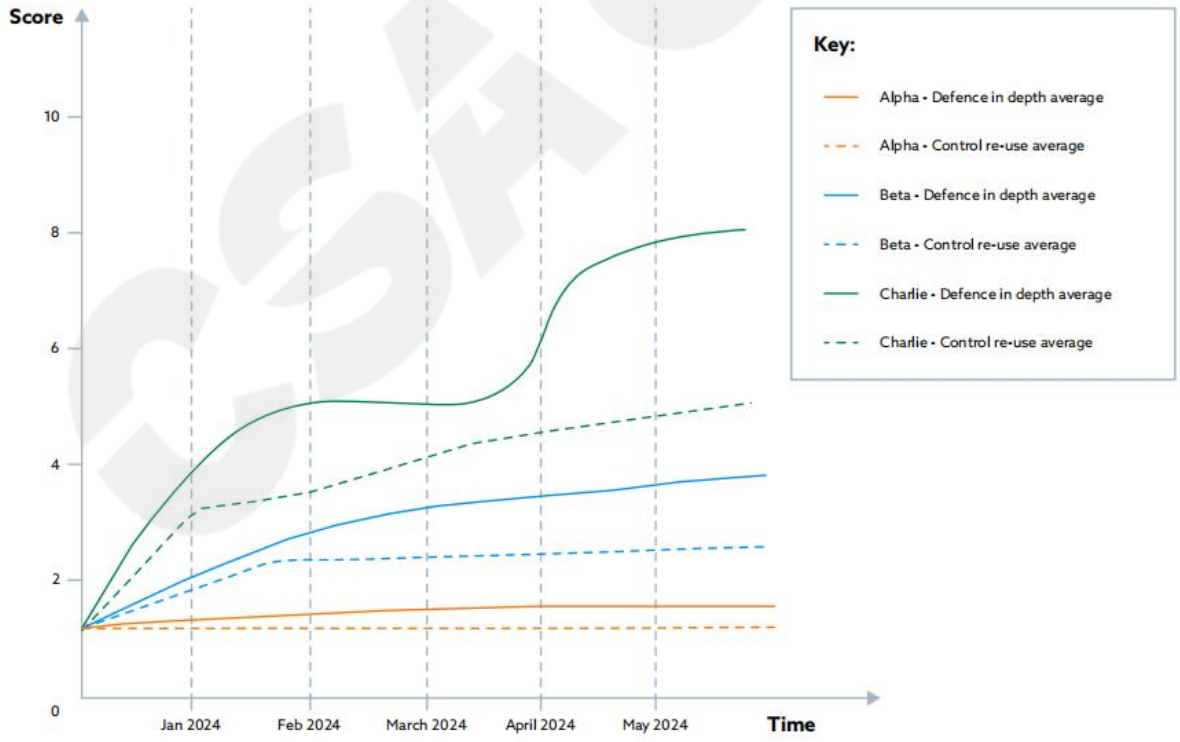


Figure 11: Comparison between teams for security architecture

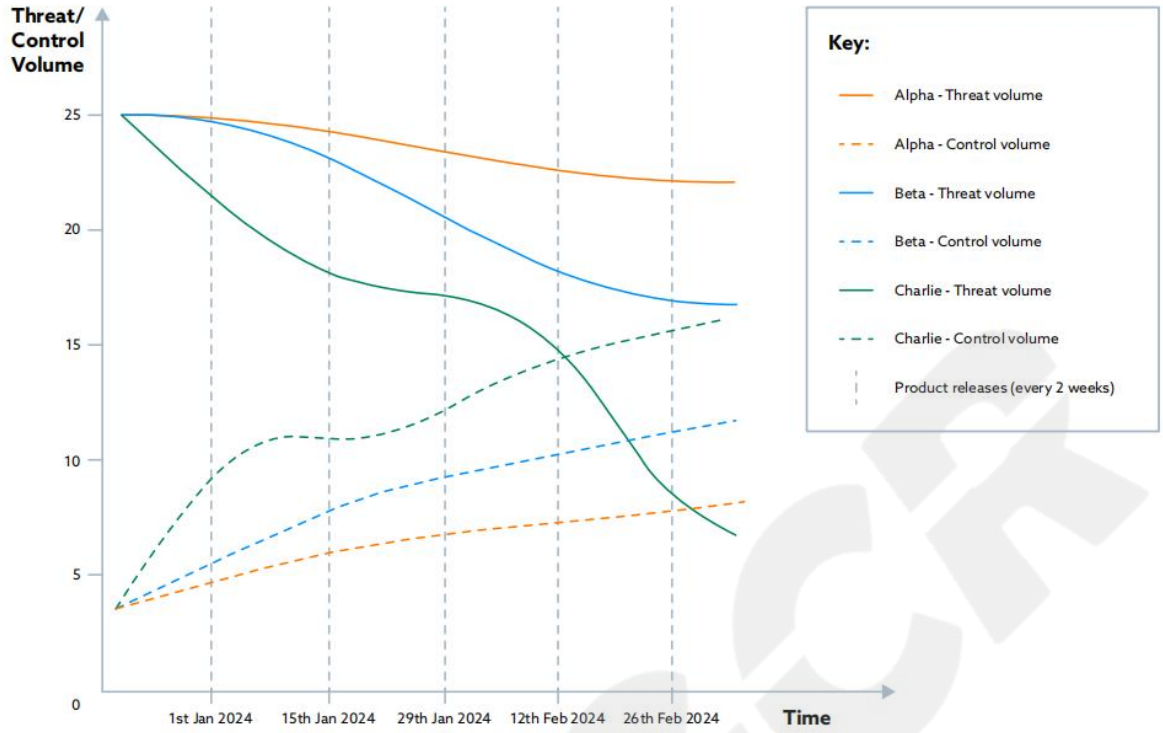


Figure 12: Comparison between teams - threat burndown

跨团队的事件响应可观测性

Alpha 团队缺乏正式的事件响应计划，导致事件响应行动的可观察性较差。Beta 团队的基本计划提高了可观察性，而 Charlie 团队定义明确且定期更新的计划带来了出色的事件响应可观察性。

平均故障发现时间（MTTD）：Alpha 团队缺乏有效的监控工具和实践，导致平均故障发现时间（MTTD）延长，平均为 19 天，并且随着时间的推移而增加。基于已有监控能力，Beta 团队将这一时间缩短到 15 天，并随着时间的推移显示出改进的迹象。然而，由于流程和人员配备不足，出现了改进停滞不前的情况。Charlie 团队拥有一套强大的检测和响应能力以及稳健的流程，这意味着它能够在几天内发现并解决故障，维持较低的 MTTD。

平均故障确认时间（MTTC）+平均故障解决时间（MTTRN）：对于 Alpha 来

说，事件通常需要时间来解决，积压的工作造成了超负荷，导致 MTTC 和 MTTRN 数值随时间增加。Beta 已经建立了支持分类的工具，但受限于随时间持续改进的能力。Charlie 由于有效地使用了工具、人员和流程，能够灵活调整并重新优先处理遏制和恢复活动，持续降低了 MTTC 和 MTTRN。

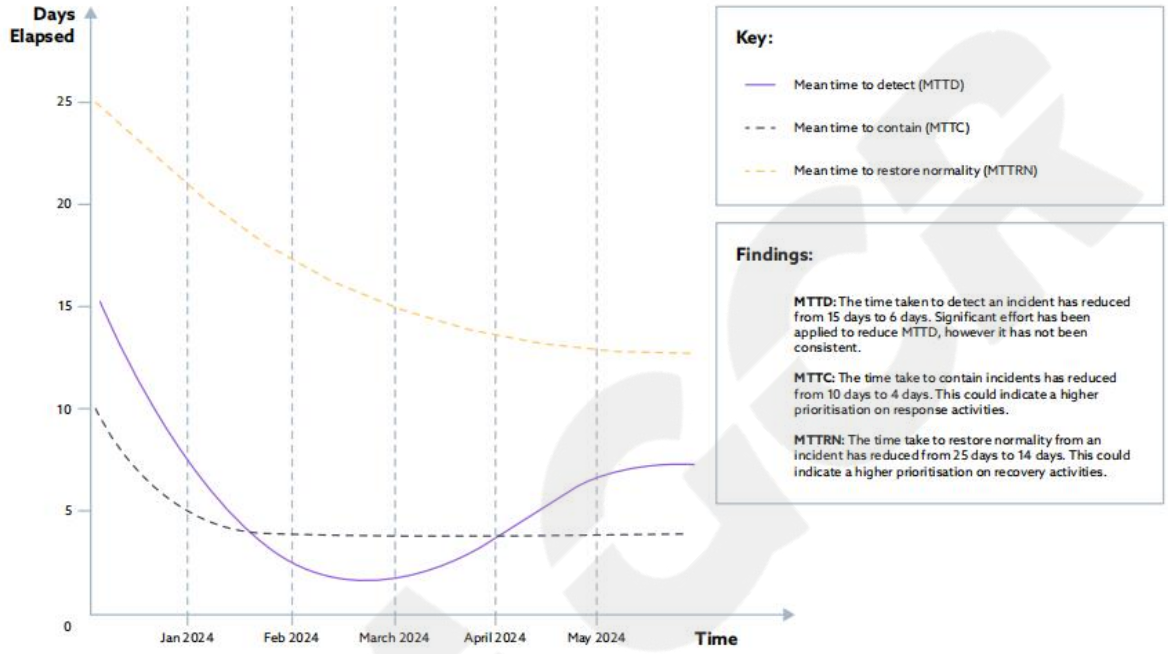


Figure 13: Conceptual observability for incident response

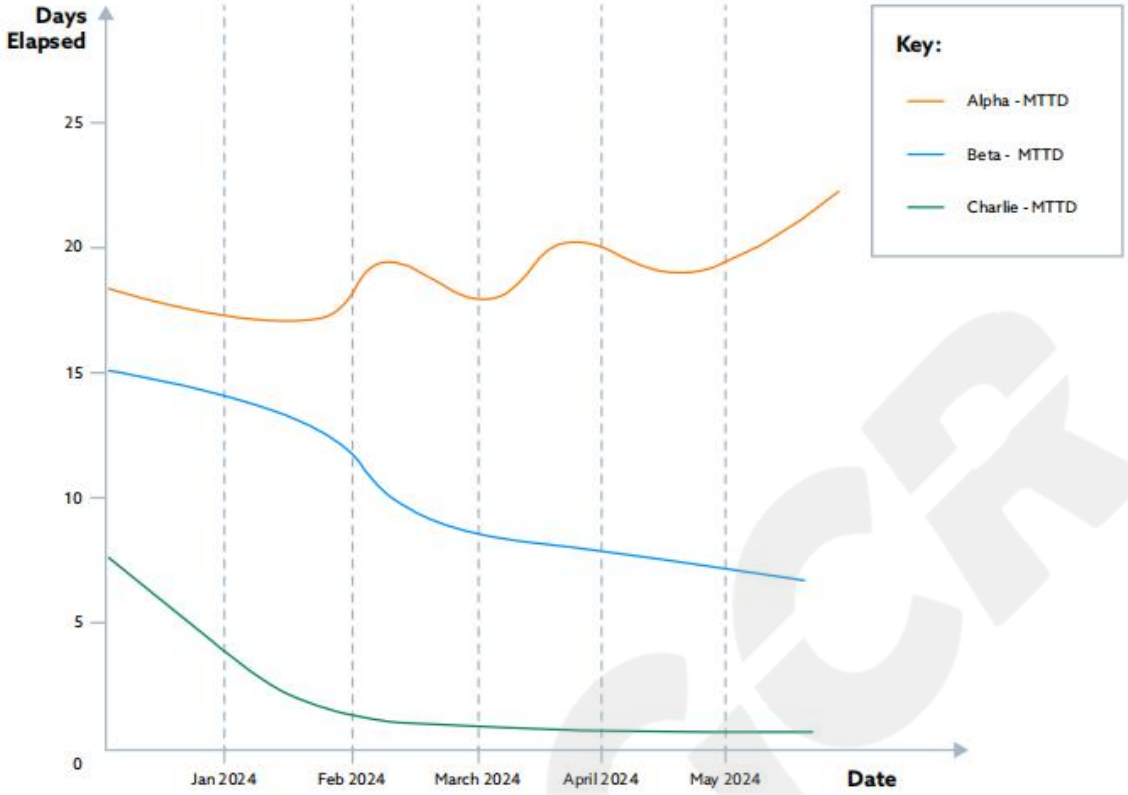


Figure 14: Comparison between teams - MTTD

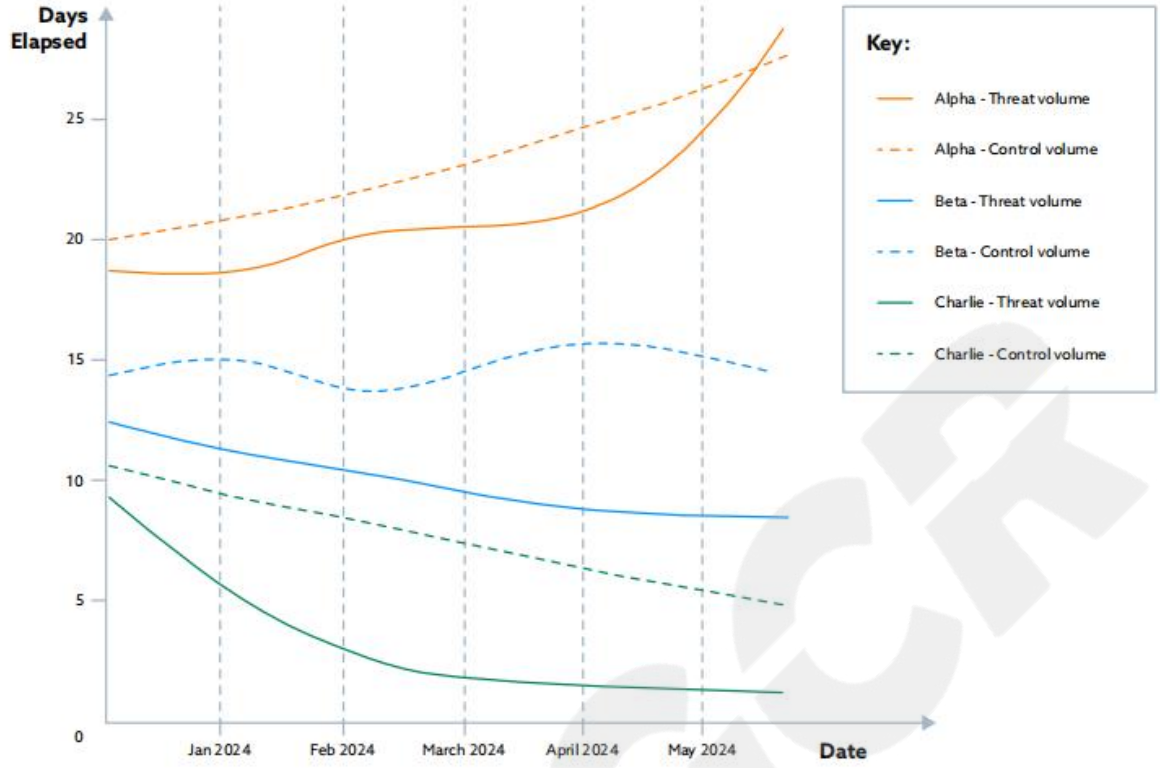


Figure 15: Comparison between teams - MTTC and MTTRN

通过报告提高

报告是组织在嵌入安全性时应该考虑的一个关键方面,因为它不仅提供了当前安全状况的视图,还为高层领导提供了评估和管理风险、安全预算和资源的能力。通过报告实现持续改进,关键在于确定哪些信息对项目团队来说是相关的,哪些应该报告给高层管理。这远不止是简单地分享由安全解决方案提供的漏洞报告那么简单。

这里描述了四个原则及其应用,在构建报告以为组织带来最大收益时应该考虑这些原则,并提供了一条通往报告成功的路线图。

原则 – 使数据可访问和可观察

通过收集和表达有意义的数据,组织能够评估当前的安全状况,并进行趋势分

析和风险缓解。可能获得的好处包括:

- **提高风险意识:**可访问和可观察的数据能直接提高风险意识。当数据易于获取时,团队能更好地识别潜在的风险和漏洞,从而采取主动的风险缓解策略。

- **增加对安全的投资:**可访问的数据使组织能够更有效地分配资源,识别需要采取额外安全措施的领域。这会导致对安全的投资增加,特别是通过持续改进的举措。

- **了解威胁格局和未来规划:**数据的可访问性提供了对威胁格局的洞见,使组织能够为未来的安全挑战做好准备。它使组织能够适应和响应新兴的威胁。

- **加快事故响应活动:**可访问和可观察的数据有助于加快事故响应。当安全事故发生时,可快速获取数据,缩短检测、控制和恢复正常业务运营的时间。

原则 – 突出改进机会

这一原则在报告领域扮演着关键角色,它着眼于确定 DevSecOps 实践中需要改进的领域。它为提升创造了机会。这些机会并不局限于技术层面,还可能涉及改善事故响应计划、员工培训或采用新的安全工具和实践等更广泛的范畴。从这种报告中可获得的好处包括但不限于:

- **增强安全状况:** 直接识别并解决改进领域可以提高安全状况的稳健性。通过主动解决漏洞和弱点,组织可以更好地防御潜在威胁。

- **持续改进:** 识别和解决改进机会,培养组织内部的持续改进文化。团队可以从过去的缺陷中吸取教训,不断改善实践。

- **量身定制的解决方案:**报告使组织能够针对具体挑战和弱点量身定制解决方案。例如,如果发现培训存在差距,可以开发针对性的培训计划来解决这些具体需求。

- **适应变化的威胁:**随着威胁环境的演变,报告使组织能够及时调整安全实践,保持超越新兴威胁的能力。

原则 – 突出变化,推动持续改进

这一原则强调了报告作为推动组织持续改进的重要机制。它强调需要通过学习和发展计划来提高安全意识和文化。实施这一原则通常表现为为员工制定全面的学习和发展(L&D)计划,旨在提升组织的整体安全意识。应用这一原则可带来以下好处:

- **增强学习文化:**通过报告突出变化,增强了组织重视学习的文化。通过定期审查和采取报告洞见,员工被鼓励不断学习新的技能和知识。

- **动态适应安全需求:**随着报告突出组织安全状况的不断变化需求,它能够实现动态适应。这意味着,当发现新的漏洞时,组织能够快速调整其策略和培训计划来解决这些具体领域。

- **提高安全响应能力:**以持续改进为重点,组织变得更敏捷,更能响应威胁。从报告中吸取教训并实施变革,可降低重复事故的可能性,从而提高处理安全事故的总体响应能力,包括平均修复时间(MTTR)、平均检测时间(MTTD)和平均恢复时间(MTTRN)。

- **主动风险管理:**由报告驱动的持续改进使组织从被动转向主动。通过分析趋势和模式,组织可以预测并减轻风险,在其变成安全漏洞之前就得到缓解。

原则 – 鼓励沟通和协作

鼓励通过报告进行沟通和协作的原则是构建统一方法的核心。通过提供组织安全实践和 DevSecOps 活动的概览,报告可以增强理解和合作。这不仅支持当前的安全需求,还为组织未来应对和管理安全挑战做好准备。这种协作性报告方法带来的好处包括:

- **高级领导参与:**传达组织安全状况的报告,使高级领导能够就安全投资和举措做出明智决策。这种参与对于获得必要的资源和关注安全相关事项至关重要。

- **意识和风险识别:**通过持续报告,可以促进对安全风险的广泛理解。这种意识对于及早发现和管理组织内部的潜在安全威胁至关重要。

• **安全文化的发展:**提高沟通交流会自然促进安全文化和意识的提升。随着报告阐明安全问题及其管理情况,它增强了一种安全成为组织文化组成部分的思维方式。

• **团队关系的加强:**报告弥合了开发和安全团队之间的差距,促进了更好的协作。当这些团队在安全目标和实践方面保持一致时,组织就能更有效地应对这些挑战。

• **主动安全措施:**将安全专家纳入开发团队,如报告所述,有利于采取主动安全措施。这种"向左转移"的方法确保在开发过程的早期就考虑安全因素,从而减少潜在漏洞并简化补救工作。

应用这些报告原则

使数据可访问、突出机遇、推动持续改进以及鼓励沟通和协作的这四项原则,对于推动通过报告实现改进至关重要。这些原则共同构成了一个全面的方法,用于测量和改善 DevSecOps 生命周期中的安全性。当这些原则融入报告过程时,组织可将其安全指标从被动数据点转变为主动的安全治理和战略决策工具。

报告路线图

您的旅程指南

我们如何才能将思维转变为“安全+绩效=价值”的思维模式？

1 未来终极目标

通过整个组织的绩效衡量安全性，并使用安全可观测性来管理风险，确定项目/活动的优先级并有效分配预算。

2 当今的挑战

了解您的项目和整个组织面临的挑战 -- 低绩效的根本原因可能不是安全团队。确保您能够持续收集数据。

3 项目目标

没有一种独立的工具可以为您提供安全可观测性，因此请将其视为自己的项目。建立成功标准。如果您的项目旨在快速构建产品，需要平衡其与网络风险（例如数据泄露）的成本，并适当设置您的安全可观测性目标。

4 安全可观测性里程碑

首先通过一个项目建立概念验证以赢得人心。如果您能够展示漏洞管理的可观测性，那么接下来的问题将是我们还能在哪里应用这种方法。根据组织的数据点进行垂直扩展，并利用现有资源（不要只为了展示可观测性而采购工具）。一旦获得倡导和领导支持，就可以横向扩展多个项目。

登山装备

在开始之前，请确保您已配备好数据点/工具，以便为您提供数据。您需要漏洞扫描程序的数据，事件需要记录在 IT 服务管理系统中，并且您的安全架构师应该已经进行了威胁建模。

工作方式

不要各自为政。让组织内部和外部的领导和倡导者都参与进来，包括安全和非安全利益相关者。

结论

在任何数字项目或组织中实施安全性绝非易事。为了获利，企业往往会加快发布功能的速度，以抢占市场份额。基基础设施和应用程序的设计和构建是根据绩效指标（即部署速度、创建功能所需的时间、测试时间）来衡量，其也非常适合衡量业务目标。

相比之下，安全性非常有效地通过合规性而非绩效来衡量成功。改进安全功能的能力往往会使团队/产品变得更加合规。合规性和绩效有时可能是对立的力量，需要微妙的平衡，以避免人们认为“安全是阻碍因素”。

本文提供了实证证据，表明使用绩效指标可以有效评估安全性和 DevSecOps 的三个关键方面。我们计算项目/组织在管理漏洞、构建安全服务和响应事件方面的表现，从而提供安全可观测性。

安全可观测性为企业领导者提供了有效的衡量标准，让他们了解安全效率和性能，最重要的是，还了解了需要进一步投资和变革的地方。

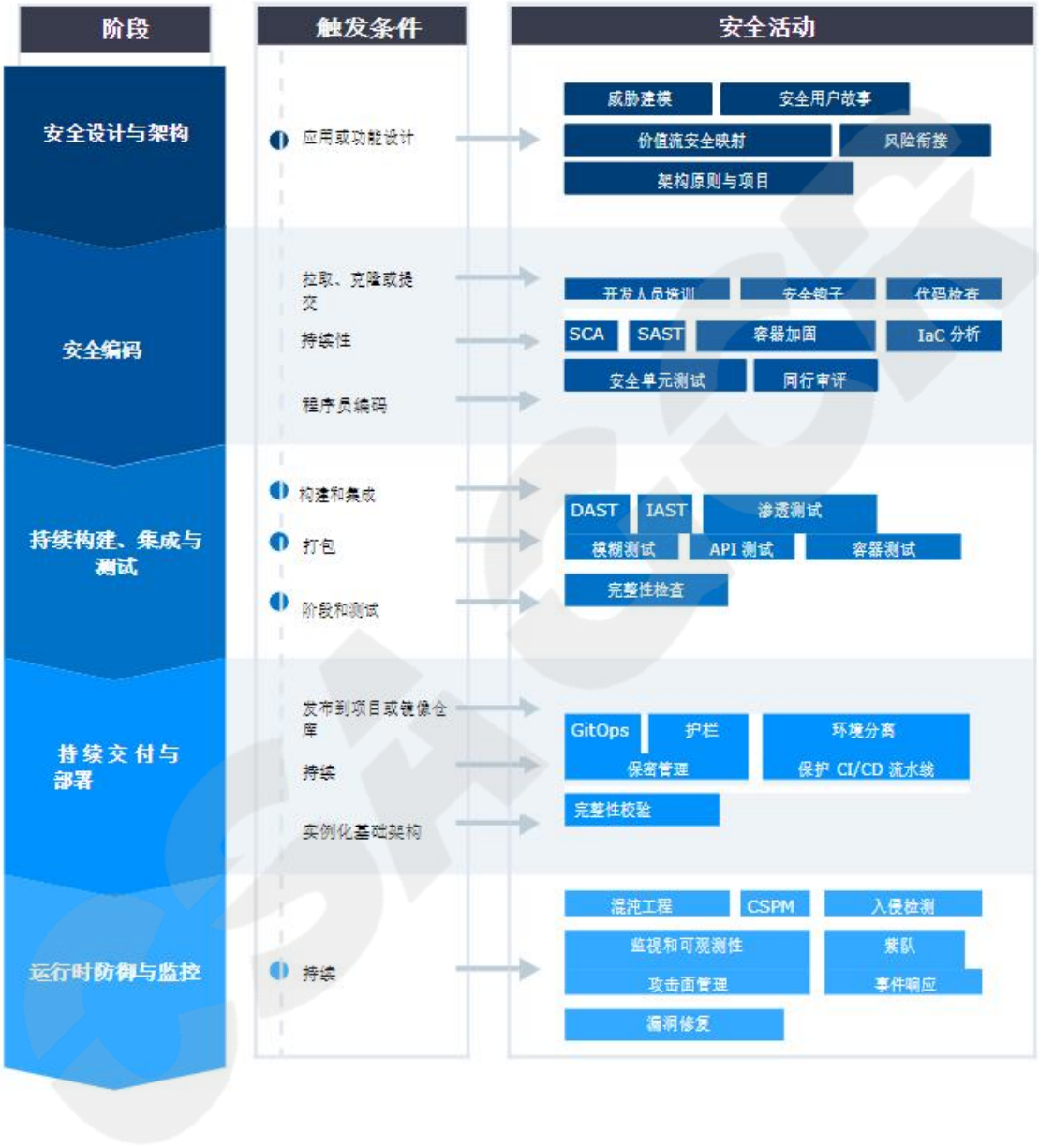
CSA DevSecOps 工作组将继续研究提供安全可观测性见解，并扩展附录 B 中的用例范围。

附录 A：软件生命周期分解

在将安全性纳入软件开发生命周期 (SDLC) 时，组织有多种工具和解决方案可供选择。在“支柱 3：务实的实现”中，我们探讨了将安全融入软件开发生命周期 (SDLC) 的设计、编码、测试、部署和监控阶段的不同安全活动。

使用专注于应用程序开发和平台安全的与框架无关的 DevSecOps 模型，我们分解了图 1 中的各个阶段，其中可以识别指标以提供数据（参见附录 B）。虽然不同定义中的 SDLC 有差异性阶段划分，但五阶段 SDLC（参见图 1：设计、编码、集成和测试、部署和监控）提供了软件开发的通用视图。

我们认识到，工具和解决方案可能难以部署和实施，而且难以扩展。当前人们的普遍看法是，现有解决方案可能无法提供有助于降低安全风险。我们挖掘如何使用 DevSecOps 阶段的度量指标来评价 DevSecOps 相关活动中的安全性能。



附录 B: 衡量

从表 1 中的每个阶段，我们已经确定了一份可能的度量标准的广泛列表，以鼓励整体决策。每个度量标准反映了各行业的产品和安全团队的常见用法。虽然表 1 并不是对最佳实践的比较或反映，但这些是团队在 SDLC 每个阶段可以收集的数据类型。

阶段	衡量标准
设计	• 每个组件，功能和特征的威胁向量 • 完成威胁建模演练的时间 • 应对威胁的成本和时间 • 安全控制/威胁比率 • 重复使用（现有）安全控制的占比 • 产品功能的安全控制占比 • 涉及安全要求和验收标准的公开占比 • 安全故事集 • 安全故事与功能故事的占比 • 完成安全用户故事所需的时间 • 安全工作的平均交付周期 • 开发过程中发现并修复的问题占比 • 测试过程中发现并修复的问题占比 • 设计过程中的风险识别量

	• 设计变更带来的风险缓解量 • 设计过程中风险解决（减轻，转移，接受，避免）的占比 • 在设计过程中识别和记录风险所需的平均时间 • 架构原则工作量 • 产品团队在设计过程中符合架构原则的占比 • 安全架构审查量
编码	• 开发人员/工程师参与安全和隐私培训的占比 • 一段时间内的经验教训数量 • 在预提交时被拒绝的代码更改占比 • 在提交后通知的占比 • 在合并前同行评审的代码更改的占比 • 已进行代码审查的占比 • 已进行代码静态检查的代码量 • 总代码静态检查错误的数量 • 修复静态检查错误所需的时间 • 符合组织许可/通过 SCA 扫描的开源组件占比 • 发现的许可问题数量 • 生产中使用的产品未解决的许可问题数量

	• 最新依赖项的占比 • 具有已知漏洞的依赖项数量 • 更新依赖项所需的平均时间 • 已扫描的 IAC 代码占比 • 已扫描的源代码占比 • 跨代码库的检查量（按语言分类） • 由于安全配置错误被拒绝的 IAC 部署量 • 发现的安全问题数量 • SAST 误报率的占比 • 处理从 SAST/IAC 扫描中发现的安全问题的安全票据数量 • 加固的容器数量 • 在生产中使用的加固容器的占比 • 加固容器更新的平均频率 • 容器不合规覆盖的数量（根据 CIS 基准或其他标准） • 与安全相关的单元测试结果的占比 • 安全单元测试的吞吐时间 • 成功和失败测试的通过率 • 进行同行评审的安全漏洞代码和架构更改的占比

测试	•通过 DAST、IAST 或持续渗透测试在运行时检测到的漏洞数量 •进行不同类型的运行时测试（如 DAST、IAST 或连续渗透测试）的平均前置时间 •检测到的高/严重漏洞的数量 •DAST/IAST 误报率的占比 •DAST/IAST 对应用功能测试覆盖的平均值 •完成 DAST、IAST 和渗透测试等各自运行时测试所需的时间 •高/严重运行时漏洞与中、低和信息性漏洞的比例 •响应无效、意外或格式错误的测试数量 •安全运行时测试期间 API 覆盖的占比 •容器镜像中的漏洞数量 •涉及容器的安全事件数量 •应用于容器的安全控制数量 •完整性检查的通过率占比
部署	•部署频率和成功率 •将基础设施部署到生产环境所需的平均时间 •识别配置偏差所需的平均时间 •修正配置偏差所需的平均时间

	• 修复配置偏差所需的手动干预步骤平均数 • 应用于云资源的护栏数量 • 转换为护栏的策略声明数量（例如，访问控制和资源管理） • 每个环境的安全事件数量 • 环境隔离测试的数量 • 机密轮换前的平均生命周期 • 存储在 CI/CD 管道中的机密数量 • 管道上触发安全扫描所用的平均时间 • 部署到生产环境所用的平均时间 • 管道部署失败的占比 • 使用的加固/批准镜像的占比 • 镜像上安全配置覆盖的占比（例如，CIS 基准第 1 级 vs. 第 2 级）
监控	• 平均恢复（Recover）时间（MTTR） • 变更失败率 • 用户服务正常运行时间 • 发布后识别的安全改进机会数量 • 检测入侵的平均时间 • 响应入侵的平均时间

	• 根据 STRIDE 用例记录的资源占比 • 来自攻击模拟（ATT&CK）的攻击向量数量 • 符合组织安全策略的外部端点数量 • 生产相关事件与非生产相关事件的占比 • 识别、调查和解决漏洞的平均时间 • 按产品分类的事件数量 • 平均修复（Remediate）时间（MTTR） • 在生产中修复的漏洞占比 • 修复或补丁的平均工作量（t-shirt 大小） • 与威胁模型直接相关的已修复漏洞占比

参考资料

Sources: 5 6

Cloud Security Alliance,The Six Pillars of DevSecOps: Pillar 1, Collective Responsibility, February 2020, Available @https://cloudsecurityalliance.org/artifacts/devsecops-collective-responsibility/

Cloud Security Alliance, The Six Pillars of DevSecOps: Pillar 2, Collaboration and Integration, February 2024, Available @ https://cloudsecurityalliance.org/artifacts/devsecops-automation/

Cloud Security Alliance, The Six Pillars of DevSecOps: Pillar 3, Pragmatic Implementation, September 2022, Available @https://cloudsecurityalliance.org/artifacts/devsecops-automation/

Cloud Security Alliance, The Six Pillars of DevSecOps: Pillar 4, Bridging Compliance and Development, February 2022, Available @https://cloudsecurityalliance.org/artifacts/devsecops-pillar-4-bridging-compliance-and-development/

Cloud Security Alliance, The Six Pillars of DevSecOps: Pillar 5, Automation, July 2020, Available @ https://cloudsecurityalliance.org/artifacts/devsecops-automation/

Cloud Security Alliance, Information Security Management through Reflexive Security: Six Pillars in the Integration of Security, Development and Operations, August 2019, Available @ https://cloudsecurityalliance.org/artifacts/information-security-management-through-reflexive-security/

Cloud Security Alliance, The Six Pillars of DevSecOps, Achieving Reflexive Security Through Integration of Security, Development and Operations, August 2019, Available @https://cloudsecurityalliance.org/artifacts/six-pillars-of-devsecops/

The Forrester New Wave™: Cybersecurity Risk Rating Solutions, Q4 2018, Available

@https://go.forrester.com/blogs/examine-the-cybersecurity-risk-ratings-market-with-forresters-new-wave/

Forrester Research on DevOps Quality Metrics that Matter, Tricentis Sponsored 75 Common Metrics Ranked by Industry Experts, 2021 Available @,

https://www.tricentis.com/resources/forrester-research-on-devops-quality-metrics/

A Measurement Companion to the CIS Critical Security Controls (Version 6) October 2015, Available

@ https://CIS-Critical-Security-Controls-VER-6.0-10.15.2015.pdf

Measures and Metrics in Corporate Security, Security Executive Council, 2011, Available @ http:// council.com/content/measures_metrics_mini_200801.pdf

Carnegie Mellon University, Software Engineering Institute: The Current State of DevSecOps Metrics, March 29, 2021, Available @ https://insights.sei.cmu.edu/blog/the-current-state-of devsecops-metrics/

GSA Tech Guides, DevSecOps Guide, July 2021, Available @ https://tech.gsa.gov/guides/dev_sec_ ops_guide/

Information Technology Laboratory / Software and Systems Division, Software Quality Group, Metrics and Measures, available @ https://www.nist.gov/itl/ssd/software-quality-group/metrics-and-measures

OWASP DevSecOps Maturity Model DSOMM, DSOMM (owasp.org)

Model, Governance, Education and Guidance, Training and Awareness, Training and Awareness (owaspsamm.org)

CSA GCR cloud security alliance GREATER CHINA REGION ®

Cloud Security Alliance Greater China Region

扫码获取更多报告

云安全联盟大中华区
官网：https://c-csa.cn 电话：0755-86548359
地址：上海市浦东新区东育路255弄5号3FB30（前滩国际经济组织集聚区）/
广东省深圳市南山区粤海街道北京航空航天大厦1号楼905