AWS Well-Architected Framework

安全性支柱

安全性支柱: AWS Well-Architected Framework

Table of Contents

安全性支柱 – AWS Well-Architected Framework

发布日期： 2023 年 12 月 6 日 (文档修订)

本白皮书主要介绍 AWS Well-Architected Framework 的安全性支柱。它提供了指导，以帮助您在安全 AWS 工作负载的设计、交付和维护过程中应用最佳实践和最新建议。

简介

此 AWS Well-Architected Framework 能够帮助您理解您在 AWS 上构建工作负载时所做的决策。通过使用此框架，您将了解有关在云中设计和运行可靠、安全、高效、经济实惠且可持续的工作负载的最新架构最佳实践。它提供了一种方法，使您能够根据最佳实践持续衡量工作负载，并确定需要改进的方面。我们相信，拥有架构完善的工作负载能够大大提高实现业务成功的可能性。

该框架基于六大支柱：

- 卓越运营
- 安全性
- 可靠性
- 性能效率
- 成本优化
- 可持续性

本白皮书重点介绍安全性支柱。这样可以帮助您遵循最新的 AWS 建议，从而满足您的业务和法规要求。本白皮书面向技术岗位的人员，例如首席技术官（CTO）、首席信息安全官（CSO/CISO）、架构师、开发人员和运营团队成员。

阅读本白皮书后，您将了解可在设计注重安全的云架构时使用的 AWS 最新建议和策略。本白皮书不提供实施细节或架构模式，但会针对此类信息提供适当资源参考。通过采用本白皮书中的实践，您可以构建能够保护您的数据和系统、控制访问并自动响应安全事件的架构。

安全基础知识

安全性支柱描述了如何利用云技术来保护数据、系统和资产，以改善您的安全状况。本白皮书提供了有关在 AWS 上构建安全工作负载的深度最佳实践指导。

设计原则

在云领域，有很多原则可帮助您提高工作负载的安全性：

- 健壮的身份验证体系： 实施最小权限原则，并通过对每一次与 AWS 资源之间的交互进行适当授权来强制执行职责分离。集中进行身份管理，并努力消除对长期静态凭证的依赖。
- 保持可追溯性： 实时监控和审计对环境执行的操作和更改并发送警报。为系统集成日志和指标收集功能，以自动调查并采取措施。
- 在所有层面应用安全措施： 利用多种安全控制措施实现深度防御。应用到所有层面（例如网络边缘、VPC、负载均衡、每个实例和计算服务、操作系统、应用程序和代码）。
- 自动实施安全最佳实践： 借助基于软件的自动化安全机制，您能够以更为快速且更具成本效益的方式实现安全扩展。创建安全架构，包括实施可在版本控制模板中以代码形式定义和管理的控制措施。
- 保护动态数据和静态数据：将您的数据按敏感程度进行分类，并采用加密、令牌和访问控制等机制（如适用）。
- 限制对数据的访问： 使用相关机制和工具来减少和消除直接访问或人工处理数据的需求。这样可以降低处理敏感数据时数据处理不当、被修改以及人为错误的风险。
- 做好应对安全性事件的准备： 制定符合您组织要求的事件管理和调查策略和流程，做好应对事件的准备工作。开展事件响应模拟演练并使用具有自动化功能的工具来提高检测、调查和恢复的速度。

定义

云中的安全包含七个方面：

- 安全基础知识
- 身份与访问权限管理
- 检测
- 基础设施保护
- 数据保护
- 事件响应

- 应用程序安全性

责任共担

安全性和合规性是 AWS 与客户的共同责任。这种共担模式可以减轻客户的运维负担，因为 AWS 运行、管理和控制从主机操作系统和虚拟化层组件，一直到服务运营所在物理设施的安全性。客户负责管理来宾操作系统（包括更新和安全补丁）、其他关联应用程序软件以及 AWS 提供的安全组防火墙的配置。客户应慎重选择服务，因为他们所承担的责任因他们使用的服务、服务与其 IT 环境的集成以及适用法律法规而各异。这一责任共担的性质还提供了支持部署的灵活性和客户控制能力。如下图所示，责任的这种区分通常称为云“本身的”安全与云“中的”安全。

AWS 负责“云本身的安全” – AWS 负责保护运行 AWS 云中提供的所有服务的基础设施。此基础设施由运行 AWS 云服务的硬件、软件、网络和设施组成。

客户负责“云中的安全” – 客户的责任将由客户选择的 AWS 云服务决定。这决定了客户必须执行的作为其安全责任一部分的配置工作量。例如，Amazon Elastic Compute Cloud（Amazon EC2）等服务被归类为基础设施即服务（IaaS），因此，这需要客户执行所有必要的安全配置和管理任务。部署 Amazon EC2 实例的客户负责管理来宾操作系统（包括更新和安全补丁）、客户在实例上安装的任何应用程序软件或实用程序，以及每个实例上由 AWS 提供的防火墙（称为安全组）的配置。对于抽象服务（例如 Amazon S3 和 Amazon DynamoDB），AWS 运行基础设施层、操作系统和平台，而客户访问端点以存储和检索数据。客户负责管理他们的数据（包括加密选项）、对其资产进行分类以及使用 IAM 工具应用适当的权限。

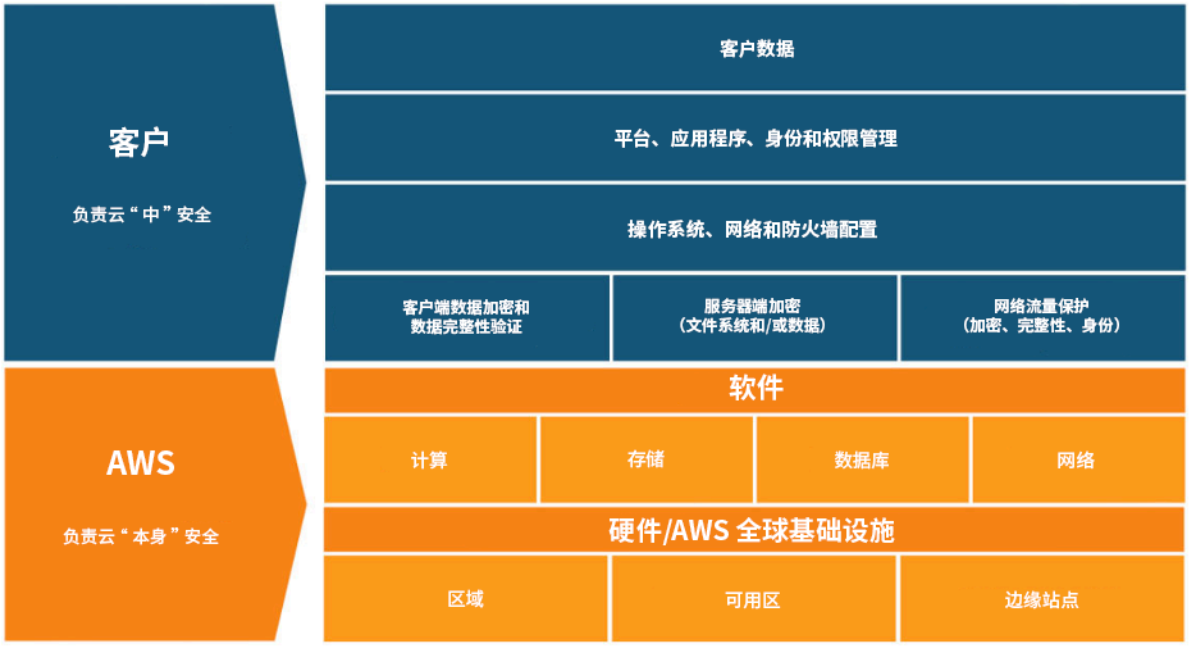


图 1：AWS 责任共担模式。

此客户/AWS 责任共担模式还扩展到 IT 控制措施方面。正如 AWS 与客户共同行使控制 IT 环境的责任一样，管理、运营和验证 IT 控制措施的责任也是由双方共同承担。AWS 负责管理与部署在 AWS 环境中的物理基础设施相关的控制措施（之此管理工作此前可能由客户承担），从而帮助客户缓解操作控制措施的负担。因为每个客户在 AWS 中的部署均不相同，所以客户可以藉此将管理特定 IT 控制措施的责任移交到 AWS，从而形成一个新型分布式控制环境。然后客户可以使用 AWS 控制和合规性文档来根据需要执行控制评估与验证程序。以下是由 AWS 和/或 AWS 客户管理的控制措施示例。

继承的控制措施 – 客户完全从 AWS 继承的控制措施。

- 物理和环境控制

共享的控制措施 – 应用于基础设施层和客户层的控制措施，但在单独的上下文中或从单独的视角应用。在共享的控制措施中，AWS 提供了对基础设施的要求，客户必须对其使用 AWS 服务实施自己的控制措施。示例包括：

- 补丁管理 – AWS 负责修补和修复基础设施中的缺陷，而客户负责修补他们的来宾操作系统和应用程序。
- 配置管理 – AWS 维护其基础设施设备的配置，而客户负责配置他们自己的来宾操作系统、数据库和应用程序。
- 意识与培训 – AWS 培训 AWS 的员工，而客户必须对自己的员工进行培训。

客户特定的 – 完全由客户负责的控制措施，基于客户在 AWS 服务中部署的应用程序。示例包括：

- 服务和通信保护或区安全，这可能需要客户在特定的安全环境中路由数据或将数据分区。

治理

安全治理是总体方法的一部分，旨在通过制定策略和控制目标来帮助管理风险，从而帮助实现业务目标。通过遵循安全控制目标的分层方法来实施风险管理，其中每一层均基于前一层而构建。了解 AWS 责任共担模式是您的基础层。这方面的知识阐明了您需对客户承担的责任以及您从 AWS 得到了什么。一项很好的资源是 AWS Artifact，它使您能够按需访问 AWS 安全与合规报告以及选定的在线协议。

在下一层满足您的大部分控制目标。在该层提供了平台范围的功能。例如，该层包括 AWS 账户分配过程、与身份提供程序（例如 AWS IAM Identity Center）的集成以及常见的检测性控制。平台治理过程的一些输出也位于该层。在您希望开始使用新的 AWS 服务时，更新 AWS Organizations 服务中的服

务控制策略（SCP）可为服务的初次使用提供防护机制。您可以使用其他 SCP 来实施常见的安全控制目标，这通常称为安全不变量。这些是您应用于多个账户、组织单位或整个 AWS 组织的控制目标或配置。典型示例是限制运行基础设施的区域，或防止停用检测性控制措施。该中间层还包含编码策略，例如配置规则或签入管道。

顶层是产品团队满足控制目标的地方，这是因为实施是在产品团队控制的应用程序中完成的。这可能是在应用程序中实施输入验证或确保身份在各项微服务之间正确传递。尽管产品团队负责配置，他们也仍能从中间层继承一些功能。

无论您在何处实施控制措施，目标都是一致的，即管理风险。一系列风险管理框架将应用于特定的行业、区域或技术。您的主要目标：根据可能性和后果来强调风险。这是 固有风险访问 AWS 资源。紧接着，您可以定义控制目标，降低可能性和/或减少后果。随后，采用适当的控制措施后，您可以查看可能产生哪些风险。这是 剩余风险访问 AWS 资源。控制目标可应用于一个或多个工作负载。下图显示了一个典型的风险矩阵。可能性基于以前发生事件的频率，而后果基于事件的财务、声誉和时间成本。

可能性	风险等级				
极有可能	低	中	高	严重	严重
有可能	低	中	中	高	严重
可能	低	低	中	中	高
不可能	低	低	中	中	高
极不可能	低	低		中	高
结果	转换时间	低	中	高	严重

图 2：风险等级可能性矩阵

AWS 账户管理和分离

我们建议您根据职能、合规性要求或一组通用控制措施，在单独的账户和组账户中组织工作负载，而不是完全沿用您企业的报告结构。在 AWS 中，账户是硬性边界。例如，强烈建议执行账户级分离，以使生产工作负载与开发和测试工作负载分离。

集中管理账户： AWS Organizations 自动创建和管理 AWS 账户，并在创建这些账户之后控制它们。当您通过 AWS Organizations 创建账户时，请务必考虑使用您的电子邮件地址，因为这将是允许重置密码的根用户。Organizations 允许您根据工作负载的要求和用途， 将账户分组为代表不同环境的组织部门 (OU)。

集中设置控制： 在适当的级别，只允许特定的服务、区域和服务操作，以控制您的 AWS 账户能够执行的操作。AWS Organizations 允许您使用服务控制策略（SCP），在组织、组织部门或账户级别应用权限防护机制，此操作适用于所有 AWS Identity and Access Management (IAM) 用户和角色。例如，您可以利用 SCP 禁止用户从您未明确允许的区域启动资源。AWS Control Tower 能够以一种简化的方式设置和管理多个账户。它会自动在您的 AWS Organization 中设置账户、自动预置、应用 防护机制（包括预防和检测），并提供一个控制面板以使您获得可见性。

集中配置服务和资源： AWS Organizations 可帮助您配置 AWS 服务 ，这些服务将应用于您的所有账户。例如，您可以使用 AWS CloudTrail配置集中日志记录功能，以记录在您的组织中执行的所有操作，并禁止成员账户停用日志记录功能。您也可以使用 AWS Config集中聚合您定义的规则的数据，以便能够审计您的工作负载是否合规，并快速对变化做出反应。AWS CloudFormation StackSets 允许您在组织中跨账户和 OU 集中管理 AWS CloudFormation 堆栈。这样，您就可以自动预置一个新账户，以满足您的安全要求。

使用安全服务的委托管理功能，将用于管理的账户与组织计费（管理）账户分隔开。多项 AWS 服务（例如，GuardDuty、Security Hub 和 AWS Config）支持与 AWS Organizations 的集成，包括为管理功能指定特定的账户。

最佳实践

- SEC01-BP01 通过使用账户来分隔工作负载
- SEC01-BP02 保护账户根用户和属性

SEC01-BP01 通过使用账户来分隔工作负载

通过采取多账户策略，在环境（如生产、开发和测试）和工作负载之间建立共同的防护机制和隔离措施。强烈建议在账户层面进行分离管理，这样可为安全性、账单和访问提供强大的隔离边界。

期望结果：形成一种账户结构，可将云运维、无关工作负载和环境隔离到单独的账户中，从而提高整个云基础设施的安全性。

常见反模式：

- 将多个相互毫无关联，具有不同数据敏感度级别的工作负载放入同一账户中。
- 组织单位（OU）结构界定不清。

建立此最佳实践的好处：

- 即使不该访问的工作负载无意中被访问了，影响范围也会缩小。
- 能够对访问 AWS 服务、资源和区域进行集中治理。
- 可集中管理策略和安全服务，维护云基础设施的安全性。
- 使账户创建和维护流程自动化。
- 集中审计基础设施状况，从而满足法规遵从性和监管要求。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

AWS 账户 提供安全隔离边界，使不同敏感度的工作负载或资源相互分离。AWS 提供相应工具，以多账户策略来大规模管理云工作负载，从而利用此隔离边界。如要获得有关 AWS 多账户策略的概念、模式和实施的指导，请参阅使用多个账户组织 AWS 环境。

如果您需要集中管理多个 AWS 账户，您的账户应基于组织单位（OU）层，建立层次结构。然后可以建立安全控制机制，并将其应用于 OU 和成员账户，从而为组织内的成员账户建立一致的预防性控制机制。安全控制机制是层层继承的，使您能够筛选位于 OU 层次结构较低层次的成员账户的可用权限。优秀的架构设计将能够利用这种层层继承的特性，减少设置安全策略，降低复杂性，并使每个成员账户的安全控制效果达到预期。

采用 AWS Organizations 和 AWS Control Tower 这两种服务，可在您的 AWS 环境中实施和管理多账户结构。AWS Organizations 使得您能够将账户建立成由一个或多个 OU 层定义的层次结构形式，每个 OU 均可包含若干成员账户。服务控制策略（SCP）使组织管理员能够对成员账户建立精细的预防性控制，而 AWS Config 可用于建立对成员账户的主动式和检测性控制。许多 AWS 服务与 AWS Organizations 集成，以提供委派型的管理控制，并在组织内的所有成员账户中执行特定于服务的任务。

AWS Control Tower 位于 AWS Organizations 之上，为具有登录区的多账户 AWS 环境提供了一键式最佳实践设置。登录区是 Control Tower 多账户环境的入口处。与 AWS Organizations 相比，采用 Control Tower 具有若干好处。这三种好处可以改进账户治理状况：

- 将强制安全防护机制集成于系统中，可自动应用于准入组织的账户。
- 有多种防护机制可供选择，还能开启或关闭给定 OU 组的防护机制。
- AWS Control Tower Account Factory 可在组织内部自动部署账户，设置好预先批准的基准和配置选项。

实施步骤

1. 设计组织单位结构：设计良好的组织单位结构减少了创建和维护服务控制策略及其他安全控制机制所需的管理负担。您的组织单位结构应与您的业务需求、数据敏感度和工作负载结构看齐。
2. 为多账户环境创建登录区：登录区提供了一致的安全性和基础设施基础，您的组织可以从中快速开发、启动和部署工作负载。您可以使用定制的登录区或 AWS Control Tower 来编排您的环境。
3. 建立防护机制：通过登录区为您的环境实施一致的安全防护机制。AWS Control Tower 提供了可部署的必选和可选控制机制的列表。实施 Control Tower 时会自动部署必选控制机制。查看高度推荐和可选控制机制的列表，并实施适合您需求的控制机制。
4. 限制访问新添加的区域：对于新的 AWS 区域，诸如用户和角色之类的 IAM 资源将仅传播到您指定的区域。可以在使用 Control Tower 时通过控制台执行此操作，也可以通过调整 AWS Organizations 中的 IAM 权限策略执行此操作。
5. 考虑使用 AWS CloudFormation StackSets：StackSets 可帮助您通过已批准的模板将资源（包括 IAM 策略、角色和组）部署到不同的 AWS 账户 和区域中。

资源

相关最佳实践：

- SEC02-BP04 依赖集中式身份提供程序

相关文档：

- AWS Control Tower
- AWS 安全性审计指导原则
- IAM 最佳实践

- 使用 CloudFormation StackSets 跨多个 AWS 账户 和区域预置资源
- 组织常见问题解答
- AWS Organizations 术语和概念
- AWS Organizations 多账户环境中的服务控制策略的最佳实践
- AWS 账户管理参考指南
- 使用多个账户组织 AWS 环境

相关视频：

- 利用自动化和监管，支持大规模采用 AWS
- 架构完善的安全性最佳实践
- 使用 AWS Control Tower 构建和监管多个账户
- 为现有组织启用 Control Tower

相关研讨会：

- Control Tower 沉浸日

SEC01-BP02 保护账户根用户和属性

根用户是 AWS 账户 中权限最高的用户，对账户内的所有资源具有完全管理访问权限，在某些情况下不受安全策略的约束。禁用对根用户的编程访问，为根用户建立适当的控制机制，并避免日常使用根用户。这样有助于降低无意中暴露根凭证以及随后破坏云环境的风险。

期望结果：保护根用户有助于减少因滥用根用户凭证而导致意外或故意损坏的可能性。建立检测性控制机制也可以在有人使用根用户执行操作时向适当人员发出警报。

常见反模式：

- 使用根用户执行各种任务，而非仅在必要时使用根用户凭证。
- 忽略定期测试应急计划，以验证关键基础设施、流程和人员在紧急情况下的运作情况。
- 只考虑典型的账户登录流程，而没有考虑或测试替代的账户恢复方法。
- 因为 DNS、电子邮件服务器和电话提供商要用于账户恢复流程，就不将其作为关键安全边界的一部分进行处理。

建立此最佳实践的好处：保护对根用户的访问可以建立信心，使您账户中的操作受到控制和审计。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

AWS 提供许多有助于保护账户安全的工具。但由于其中一些措施默认情况下未启用，因此您必须采取直接行动来实施它们。请将这些建议视为确保 AWS 账户 安全的基本步骤。实施这些步骤时，务必建立一个持续评估和监控安全控制机制的过程，这非常重要。

当您首次创建 AWS 账户 时，最初使用的是一个对账户中所有 AWS 服务和资源有完全访问权限的身份。该身份称为 AWS 账户 根用户。您可以使用创建账户时使用的电子邮件地址和密码，以根用户的身份登录。由于授予 AWS 根用户的访问权限较高，您必须仅将 AWS 根用户用于执行特别需要它的任务。必须严格保护根用户登录凭证，并且应始终为 AWS 账户 根用户启用多重身份验证（MFA）。

除了使用用户名、密码和多重身份验证（MFA）设备登录根用户的常规身份验证流程外，还可以使用账户恢复流程登录您的 AWS 账户 根用户，该用户可以访问与您的账户关联的电子邮件地址和电话号码。因此，保护发送恢复电子邮件的根用户电子邮件账户和保护与该账户关联的电话号码同样重要。还应考虑潜在的循环依赖关系，其中与根用户关联的电子邮件地址托管在同一 AWS 账户 的电子邮件服务器或域名服务（DNS）资源上。

使用 AWS Organizations 时，有多个 AWS 账户（均有一个根用户）。将一个账户指定为管理账户，然后可以在管理账户下面添加几层成员账户。优先保护管理账户的根用户，然后解决成员账户根用户问题。保护管理账户根用户的策略可能与保护成员账户根用户的策略不同，您可以对成员账户根用户建立预防性安全控制机制。

实施步骤

建议使用以下实施步骤为根用户建立控制机制。在适用情况下，建议与 CIS AWS Foundations Benchmark 版本 1.4.0 交叉引用。除了这些步骤外，请参阅 AWS 最佳实践指导以确保您的 AWS 账户和资源安全。

预防性控制机制

1. 为账户设置准确的联系信息。
    a. 该信息用于丢失的密码恢复流程、丢失的 MFA 设备账户恢复流程，以及与您的团队进行关键的安全相关通信。
    b. 使用企业域托管的电子邮件地址（最好是通讯组列表）作为根用户的电子邮件地址。使用通讯组列表而不是个人的电子邮件账户可提供额外的冗余和连续性，以便在很长一段时间内访问根账户。

c. 联系信息上所列的电话号码应该是为此目的而设置的专用安全电话的号码。电话号码不应列出或与任何人共享。

2. 不要为根用户创建访问密钥。如果存在访问密钥，请将其删除（CIS 1.4）。

   a. 消除根用户的任何长期编程凭证（访问密钥和私有密钥）。

   b. 如果已存在根用户访问密钥，您应将使用这些密钥的进程转换为使用 AWS Identity and Access Management（IAM）角色的临时访问密钥，然后删除根用户访问密钥。

3. 确定是否需要为根用户存储凭证。

   a. 如果您使用 AWS Organizations 创建新的成员账户，则新成员账户上根用户的初始密码将设置为一个不向您公开的随机值。如果需要，请考虑使用 AWS 组织管理账户的密码重置流程来访问成员账户。

   b. 对于独立 AWS 账户 或管理 AWS 组织账户，请考虑为根用户创建并安全地存储凭证。为根用户启用 MFA。

4. 在 AWS 多账户环境中，为成员账户根用户启用预防性控制机制。

   a. 考虑为成员账户启用不允许为根用户创建根访问密钥预防性防护机制。

   b. 考虑为成员账户启用不允许以根用户身份执行操作预防性防护机制。

5. 如果需要根用户凭证，请执行以下操作：

   a. 使用复杂密码。

   b. 为根用户启用多重身份验证（MFA），特别是 AWS Organizations 管理（付款人）账户（CIS 1.5）。

   c. 考虑使用硬件 MFA 设备以提高弹性和安全性，因为一次性使用的设备可以减少包含 MFA 代码的设备被重复用于其他用途的可能性。验证是否定期更换由电池供电的硬件 MFA 设备。（CIS 1.6）

      - 要为根用户配置 MFA，请遵循启用虚拟 MFA 或硬件 MFA 设备的说明。

   d. 考虑注册多个 MFA 设备用于备份。 每个账户最多允许 8 个 MFA 设备。

      - 请注意，为根用户注册多个 MFA 设备将自动禁用在 MFA 设备丢失情况下恢复账户的流程。

   e. 安全地存储密码，如果以电子方式存储密码，则考虑循环依赖关系。不要以需要访问同一 AWS 账户 才能获得密码的方式存储密码。

6. 可选：考虑为根用户制定定期密码轮换计划。

   - 凭证管理最佳实践取决于您的监管和政策要求。受 MFA 保护的根用户并不依赖密码作为单重身份验证。
   - 定期更改根用户密码可降低无意中暴露的密码被滥用的风险。

检测性控制

- 创建警报以检测根凭证的使用（CIS 1.7）。启用 Amazon GuardDuty 将通过 RootCredentialUsage 调查结果对根用户 API 凭证的使用进行监控和发出警报。
- 评估并实施 AWS Well-Architected 安全性支柱合规包 AWS Config 中包含的检测性控制机制，或者如果使用 AWS Control Tower，则评估并实施 Control Tower 内强烈建议的控制机制。

运营指南

- 确定组织中应该有权访问根用户凭证的人员。
  - 采用双人规则，以便不会出现一个人就能够访问所有必要凭证和 MFA 来获得根用户访问权限的情况。
  - 验证组织（而不是个人）对与账户关联的电话号码和电子邮件别名（用于密码重置和 MFA 重置流程）保持控制。
- 仅在例外情况下使用根用户（CIS 1.7）。
  - 不得使用 AWS 根用户执行日常任务，即使是管理任务也不可以。仅以根用户身份登录，以执行需要根用户的 AWS 任务。所有其他操作都应由代入适当角色的其他用户执行。
- 定期检查对根用户的访问是否正常，以便在出现需要使用根用户凭证的紧急情况之前对过程进行测试。
- 定期检查与账户关联的电子邮件地址以及备用联系人下列出的电子邮件地址是否有效。监控这些电子邮件收件箱，查看您可能从中收到的安全通知 <abuse@amazon.com>。还要确保与该账户相关的任何电话号码都有效。
- 准备事件响应程序，以应对根账户滥用情况。请参阅 AWS 安全事件响应指南以及《安全性支柱》白皮书“事件响应”部分中的最佳实践，了解有关为 AWS 账户构建事件响应策略的更多信息。

资源

相关最佳实践：

- SEC01-BP01 通过使用账户来分隔工作负载
- SEC02-BP01 使用强大的登录机制
- SEC03-BP02 授予最低访问权限
- SEC03-BP03 建立紧急访问流程
- SEC10-BP05 预置访问权限

相关文档：

- AWS Control Tower
- AWS 安全性审计指导原则
- IAM 最佳实践
- Amazon GuardDuty – 根凭证使用情况警报
- 通过 CloudTrail 监控根凭证使用情况的分步指导
- 获准与 AWS 一起使用的 MFA 令牌
- 在 AWS 上实施 Break Glass 访问
- AWS 账户中需要改进的十大安全项目
- 发现我的 AWS 账户 中存在未经授权的活动时该怎么办？

相关视频：

- 利用自动化和监管，支持大规模采用 AWS
- 架构完善的安全性最佳实践
- 限制使用来自 AWS re:inforce 2022 的 AWS 根凭证 – AWS IAM 安全最佳实践

相关示例和实验室：

- 实验室：AWS 账户 和根用户

安全地操作您的工作负载

安全地操作工作负载涵盖了从设计、构建、运行到持续改进的整个工作负载生命周期。为了增强您在云中安全运营的能力，其中一种方法是采用组织化的方法进行治理。治理是采用一致的方法来指导决策，而不是完全依赖于相关人员做出良好判断。您的治理模式和流程决定了您如何回答以下问题：“我如何知道给定工作负载的控制目标已实现并且适用于该工作负载？”采用一致的方法来制定决策可以加快部署工作负载，并帮助提高组织中的安全能力标准。

为了安全地操作您的工作负载，您必须对安全性的各个方面应用总体最佳实践。采用您在组织和工作负载层面的卓越运营中定义的要求和流程，并将它们应用到各个方面。及时了解最新的 AWS、行业建议以及威胁情报信息可帮助您改进您的威胁模型和控制目标。实现安全流程、测试和验证的自动化可帮助您扩展安全运营。

利用自动化，可以实现流程的一致性和可重复性。虽然每个人都擅长做很多事情，但肯定不能始终如一地重复做同一件事而不出错。即使编写了良好的运行手册，您也会面临人员无法始终如一地执行重复任务的风险。当人员承担多种责任并且必须对不熟悉的提醒做出响应时尤为如此。不过，自动化每次都会以相同的方式响应。通过自动化部署应用程序是最佳选择。可以先测试运行部署的代码，然后将该代码用于执行部署。这增加了变更过程中的信心，同时降低了变更失败的风险。

要验证配置是否达到您的控制目标，请首先在非生产环境中测试自动化和部署的应用程序。这样一来，您就可以测试自动化，证明它正确地执行了所有步骤，还可以在开发和部署周期中获得早期反馈，从而减少返工。要降低出现部署错误的几率，请通过代码而不是人员来进行配置更改。如果您需要重新部署应用程序，可以利用自动化更轻松地完成此操作。在定义其他控制目标时，您可以轻松地将它们添加到所有工作负载的自动化中。

让各个工作负载负责人使用常见功能和共享组件来节省时间，而不是将精力放在实现特定于其工作负载的安全性上。多个团队可使用的服务的一些示例包括 AWS 账户创建过程、人员的集中化身份、通用日志记录配置以及 AMI 和容器基础映像创建。此方法可以帮助构建者缩短工作负载周期并始终如一地达到安全控制目标。当团队的一致性更高时，您可以验证控制目标，并向利益相关方更好地报告您的控制态势和风险状况。

最佳实践

- SEC01-BP03 识别并验证控制目标
- SEC01-BP04 及时了解最新的安全威胁
- SEC01-BP05 及时了解最新的安全建议
- SEC01-BP06 在管道中自动测试和验证安全控制措施
- SEC01-BP07 使用威胁模型识别威胁并确定缓解措施的优先级
- SEC01-BP08 定期评估和实施新的安全服务和功能

SEC01-BP03 识别并验证控制目标

根据您的合规性要求以及从威胁模型中发现的风险，获得并验证您需要应用于工作负载的控制目标和控制措施。持续验证控制目标和控制措施可帮助您衡量风险缓解措施的有效性。

未建立这种最佳实践的情况下暴露的风险等级： 高

实施指导

- 确定合规性要求：了解您的工作负载必须符合的组织、法律和合规性要求。
- 确定 AWS 合规性资源：确定 AWS 帮助您实现合规性的资源。

- https://aws.amazon.com/compliance/
- https://aws.amazon.com/artifact/

资源

相关文档：

- AWS 安全性审计指导原则
- 安全公告

相关视频：

- AWS Security Hub：管理安全警报和自动执行合规性检查
- 架构完善的安全性最佳实践

SEC01-BP04 及时了解最新的安全威胁

通过及时了解最新的安全威胁，帮助您定义并实施适当的控制措施，识别攻击媒介。使用 AWS Managed Services 可以更轻松地接收 AWS 账户中意外或异常行为的通知。在您的安全信息流程中，使用 AWS 合作伙伴工具或第三方威胁信息源进行调查。此 通用漏洞披露（CVE，Common Vulnerabilities and Exposures）列表 包含公开披露的网络安全漏洞，可供您用于掌握最新信息。

未建立此最佳实践暴露的风险等级： 高

实施指导

- 订阅威胁情报来源：定期查看来自多个来源、与您在工作负载中所用技术相关的威胁情报信息。
  - 通用漏洞披露列表
- 考虑使用 AWS Shield Advanced 服务：如果您的工作负载可通过互联网访问，则该服务可让您近乎实时地了解情报来源。

资源

相关文档：

- AWS 安全性审计指导原则
- AWS Shield

- 安全公告

相关视频：

- 架构完善的安全性最佳实践

SEC01-BP05 及时了解最新的安全建议

及时了解最新的 AWS 和行业安全建议，以改善您的工作负载安全状况。AWS 安全公告 包含有关安全性和隐私通知的重要信息。

未建立此最佳实践暴露的风险等级： 高

实施指导

- 关注 AWS 更新：订阅或定期查看新建议、提示与诀窍。
  - AWS Well-Architected 实验室
  - AWS 安全性博客
  - AWS 服务文档
- 订阅行业新闻：定期查看来自多个来源、与您在工作负载中所用技术相关的新闻动态。
  - 示例：通用漏洞披露列表

资源

相关文档：

- 安全公告

相关视频：

- 架构完善的安全性最佳实践

SEC01-BP06 在管道中自动测试和验证安全控制措施

为安全机制建立可靠的基准和模板，并将其作为构建、管道和流程的一部分进行测试和验证。利用工具和自动化功能，持续测试并验证所有的安全控制措施。例如，对机器镜像和基础设施即代码模板等项目

进行扫描，以发现安全漏洞、异常以及与每个阶段的既定基准的偏差。AWS CloudFormation Guard 可帮助您验证 CloudFormation 模板是否安全，为您节省时间并减少配置错误风险。

减少引入到生产环境中的安全性错误配置的数量至关重要 — 在构建过程中，可以执行的质量控制和可以减少的缺陷越多越好。设计持续集成和持续部署 (CI/CD) 管道，以便尽可能测试安全问题。CI/CD 管道提供了在构建和交付的每个阶段增强安全性的机会。还必须确保 CI/CD 安全工具始终是最新版本，以减轻不断变化的威胁。

跟踪对工作负载配置进行的更改，帮助您进行合规性审计、更改管理以及可能适用于您的调查。您可以使用 AWS Config 记录和评估您的 AWS 和第三方资源。这使您可以依据规则及合规包（合规包是带有补救操作的规则集合），连续审计和评估您的整体合规情况。

更改跟踪应包括计划更改，计划更改可能是组织更改控制流程（有时也称作 MACD，即移动、添加、更改、删除（Move, Add, Change, Delete））、临时更改或意外更改（如意外事件）的一部分。更改可能出现在基础设施中，但也可能涉及其他类别，如代码存储库中的更改、机器镜像和应用程序清单更改、流程和策略更改或文档更改。

未建立此最佳实践暴露的风险等级： 中

实施指导

- 自动管理配置：使用配置管理服务或工具自动实施安全配置并对其进行验证。
  - AWS Systems Manager
  - AWS CloudFormation
  - 在 AWS 上设置 CI/CD 管道

资源

相关文档：

- 如何使用服务控制策略来设置您在 AWS Organization 中的跨账户权限防护机制

相关视频：

- 使用 AWS Organizations 管理多账户 AWS 环境
- 架构完善的安全性最佳实践

SEC01-BP07 使用威胁模型识别威胁并确定缓解措施的优先级

执行威胁建模，以识别并维护一个针对工作负载的潜在威胁和相关缓解措施的最新登记表。确定威胁优先级并调整安全控制缓解措施，以进行防范、检测和响应。根据您的工作负载以及不断变化的安全环境，重新审视和维护此登记表。

在未建立这种最佳实践的情况下暴露的风险等级： 高

实施指导

什么是威胁建模？

“威胁建模可识别、沟通和理解威胁及缓解措施，用于保护重要资产。” – 开放 Web 应用程序安全项目（OWASP）应用程序威胁建模

为什么应该进行威胁建模？

系统是复杂的，并且随着时间的推移会变得越来越复杂，功能越来越强大，从而提供更多业务价值，提高客户满意度和参与度。这意味着 IT 设计决策需要考虑数量不断增加的使用案例。由于复杂性和使用案例排列组合的数量过多，非结构化方法将通常无法有效地发现和减轻威胁。因此，您需要采用一种系统方法来列举对系统的潜在威胁，制定缓解措施并确定这些缓解措施的优先级，以确保贵组织利用有限资源，在改善系统的整体安全状况方面发挥巨大作用。

威胁建模旨在提供这种系统方法，目的是在设计过程的早期发现和解决问题。此时与生命周期的后期相比，缓解措施的成本和工作量相对较低。这种方法与左移 安全实践的行业原则相一致。最终，威胁建模将与组织的风险管理流程相互集成，并通过使用威胁驱动的方法，帮助您决定实施哪些控制措施。

应在什么时候进行威胁建模？

应在工作负载的生命周期中尽早开始进行威胁建模，这使您能够更灵活地处理已识别的威胁。就像软件漏洞一样，越早发现威胁，解决它们就越经济高效。威胁模型是一个动态文档，应随着工作负载的变化而不断发展。随着时间的推移（包括当发生重大变更、威胁形势发生变化或您采用新功能或服务时），重新审视您的威胁模型。

实施步骤

我们如何进行威胁建模？

可以采用许多不同的方式来进行威胁建模。就像编程语言一样，每种方式都有优点和缺点，您应该选择最适合自己的方式。一种方法是从 Shostack 的威胁建模 4 问题框架开始，该框架提出开放式问题，为您的威胁建模工作提供结构：

1. 正在做什么？

   该问题旨在帮助您了解所构建的系统并达成一致意见，以及了解与安全性相关的系统细节。创建模型或图表是回答该问题的常用方法，因为这可帮助您对所构建的内容进行可视化，例如使用数据流图。写下关于系统的假设和重要细节也有助于定义范围内的内容。这使每个参与威胁建模的人员都能专注于同一件事，避免因在超出范围的主题（包括过时的系统版本）上走弯路而浪费时间。例如，如果您要构建一个 Web 应用程序，那么可能不值得花时间为浏览器客户端操作系统可信引导顺序进行威胁建模，因为您无法在设计中影响这一点。

2. 会出现什么问题？

   该问题可帮助您识别系统存在的威胁。威胁是指具有不必要影响并可能影响系统安全的意外或故意行为或事件。如果不清楚哪里可能出现问题，您就无从应对。

   对于可能出现的问题，并没有一个规范的列表。创建此列表需要团队中的所有个人和参与威胁建模工作的相关角色集思广益并展开协作。您可以通过使用识别威胁的模型（如 STRIDE）来帮助集思广益，该模型建议了不同的评估类别：欺骗、篡改、抵赖、信息披露、拒绝服务和权限提升。此外，您可能希望通过回顾现有的列表和研究来帮助集思广益，寻找灵感，其中包括 OWASP Top 10、HiTrust 威胁目录和贵组织自己的威胁目录。

3. 我们要怎么做？

   与前面的问题一样，我们不可能得到包含所有缓解措施的规范清单。这一步骤需要考虑的是上一步中确定的威胁、威胁行动者和要改进的领域。

   安全性和合规性是您与 AWS 的共同责任。重要的是要明白，当您问“我们要怎么做？”时，您也在问“谁负责做这件事？”。了解您和 AWS 之间的责任平衡有助于将威胁建模工作的范围限定在您控制的缓解措施范围内，这些缓解措施通常是 AWS 服务配置选项和您自己的系统特定缓解措施的组合。

   对于共担责任中 AWS 应承担的部分，您会发现 AWS 服务在许多合规计划的范围内。这些计划可帮助您理解 AWS 用以维持云安全性与合规性的可靠控制机制。AWS 客户可以从 AWS Artifact 下载这些计划的审计报告。

   无论您使用哪项 AWS 服务，客户始终要承担一部分责任，并且与这些责任相一致的缓解措施应包含在威胁模型中。对于 AWS 服务自身的安全控制缓解措施，您需要考虑跨域实施安全控制措施，包括身份和访问管理（身份验证和授权）、数据保护（静态和传输中）、基础设施安全性、日志记录和监控等域。每项 AWS 服务的文档都包含一个专门的安全章节，里面提供了关于安全控制机制的指导，可视为缓解措施。重要的是，需要考虑您正在编写的代码及其代码依赖关系，并思考可以

设置哪些控制机制来应对这些威胁。这些控制机制可以是输入验证、会话处理和边界处理等内容。通常，大多数漏洞都是在自定义代码中引入，因此请重点关注这一领域。

4. 我们做得好吗？

   该问题旨在随着时间的推移，让您的团队和组织提高威胁模型的质量并加快执行威胁建模的速度。通过将实践、学习、教学和回顾相结合可以取得这些改进。要想深入了解并亲身体验，建议您和您的团队完成“适合构建者的威胁建模方式”培训课程或研讨会。此外，如果您正在寻找如何将威胁建模集成到组织的应用程序开发生命周期中的指导，请参阅 AWS 安全博客上的如何处理威胁建模一文。

Threat Composer

为了协助和指导您执行威胁建模，可以考虑使用 Threat Composer 工具，该工具旨在缩短威胁建模时实现价值的时间。该工具有便于您执行以下操作：

- 撰写符合威胁语法、能够在自然非线性工作流程中发挥作用的有用威胁语句
- 生成人类可读的威胁模型
- 生成机器可读的威胁模型，允许您将威胁模型视为代码
- 使用 Insights 控制面板协助您快速确定质量和覆盖范围有待改进的方面

如需更多参考，请访问 Threat Composer 并切换到系统定义的示例工作区。

资源

相关最佳实践：

- SEC01-BP03 识别并验证控制目标
- SEC01-BP04 及时了解最新的安全威胁
- SEC01-BP05 及时了解最新的安全建议
- SEC01-BP08 定期评估和实施新的安全服务和功能

相关文档：

- 如何处理威胁建模（AWS 安全性博客）
- NIST：以数据为中心的系统威胁建模指南

相关视频：

- 2021 AWS ANZ 峰会 - 如何处理威胁建模
- 2022 AWS ANZ 峰会 - 扩展安全性 – 优化以实现快速而安全的交付

相关培训：

- 适合构建者的威胁建模方式 – AWS Skill Builder 自控进度的虚拟培训
- 适合构建者的威胁建模方式 – AWS 研讨会

相关工具：

- Threat Composer

SEC01-BP08 定期评估和实施新的安全服务和功能

评估并实施 AWS 和 AWS 合作伙伴提供的安全服务和功能，以改善您的工作负载安全状况。AWS 安全博客重点介绍新的 AWS 服务和功能、实施指导和常规安全指南。AWS 的最新内容 是一个很好的工具，可帮助您随时了解所有新的 AWS 功能、服务和公告。

未建立这种最佳实践的情况下暴露的风险等级： 低

实施指导

- 规划定期审核：创建审核活动日历，包括遵守合规性要求、评估新的 AWS 安全功能和服务，以及及时了解行业最新动态。
- 发现 AWS 服务和功能：发现适用于您使用的服务的安全功能，并在新功能发布时查看这些功能。
  - AWS 安全性博客
  - AWS 安全公告
  - AWS 服务文档
- 定义 AWS 服务上线流程：定义用于上线新 AWS 服务的流程。包括您如何评估新 AWS 服务的功能，以及针对工作负载的合规性要求。
- 测试新的服务和功能：当有新的服务和功能发布时，在与生产环境非常相似的非生产环境中对其进行测试。
- 实施其他防御机制：实施自动化机制来保护您的工作负载，并探索可用选项。
  - 按照 AWS Config 规则 修正不合规的 AWS 资源

资源

相关视频：

- 架构完善的安全性最佳实践

身份与访问权限管理

要使用 AWS 服务，您必须向用户和应用程序授予对您 AWS 账户中资源的访问权限。当您在 AWS 上运行更多的工作负载时，您需要实施强大的身份管理和权限，以确保适当的人员在适当的条件下有权访问适当的资源。AWS 提供了大量功能选择，帮助您管理人员和机器身份及其权限。这些功能的最佳实践分为两个主要领域。

主题

身份管理

在访问和运行安全的 AWS 工作负载时，您需要管理两种类型的身份。

- 人员身份：管理员、开发人员、操作员以及应用程序的使用者需要拥有身份，才能访问您的 AWS 环境和应用程序。他们可能是您的组织成员或与您协作的外部用户，也可能是通过 Web 浏览器、客户端应用程序、移动应用程序或交互式命令行工具与您的 AWS 资源交互的用户。
- 机器身份：您的工作负载应用程序、操作工具和组件需要拥有身份，才能向 AWS 服务发出请求以执行某种操作，例如读取数据。这些身份包括在 AWS 环境中运行的机器，例如 Amazon EC2 实例或 AWS Lambda 函数。您还可以管理需要访问权限的外部各方的机器身份。此外，您可能还有位于 AWS 之外的机器需要访问 AWS 环境。

最佳实践

- SEC02-BP01 使用强大的登录机制
- SEC02-BP02 使用临时凭证
- SEC02-BP03 安全地存储和使用密钥
- SEC02-BP04 依赖集中式身份提供程序
- SEC02-BP05 定期审计和轮换凭证
- SEC02-BP06 利用用户组和属性

SEC02-BP01 使用强大的登录机制

当不使用多重身份验证（MFA）等机制时，登录（使用登录凭证的身份验证）可能会带来风险，特别是在登录凭证被无意泄露或很容易猜到的情况下。使用强大的登录机制，通过要求使用 MFA 和强密码策略来降低这些风险。

期望结果：通过为 AWS Identity and Access Management（IAM）用户、AWS 账户根用户、AWS IAM Identity Center（AWS Single Sign-On 的后继者）和第三方身份提供者使用强大的登录机制，降低意外访问 AWS 中凭证的风险。这意味着需要 MFA，强制执行强密码策略，并检测异常登录行为。

常见反模式：

- 没有为您的身份执行强密码策略，包括复杂密码和 MFA。
- 在不同的用户之间共享相同的凭证。
- 不对可疑的登录使用检测性控制。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

人类身份可通过多种方式登录 AWS。在向 AWS 进行身份验证时，AWS 最佳做法是寻找使用联合身份验证（直接联合或使用 AWS IAM Identity Center）的集中式身份提供者进行验证。在这种情况下，您应与您的身份提供者或 Microsoft Active Directory 建立安全登录过程。

第一次开设 AWS 账户 时，您会从 AWS 账户 根用户开始。您应仅使用账户根用户为您的用户（以及为需要根用户的任务）设置访问权限。开设 AWS 账户 后立即为账户根用户启用 MFA，并使用 AWS 最佳实践指南保护根用户，这一点非常重要。

如果您在 AWS IAM Identity Center 中创建用户，请确保该服务中的登录过程安全。对于消费者身份，您可以使用 Amazon Cognito user pools 并保护该服务中的登录过程，或者使用 Amazon Cognito user pools 支持的身份提供者之一。

如果您使用 AWS Identity and Access Management（IAM）用户，则将使用 IAM 保护登录过程。

无论采用何种登录方法，执行强登录策略都非常关键。

实施步骤

以下是一般的强登录建议。应根据您的公司策略或使用 NIST 800-63 等标准，对您配置的实际设置进行设置。

- 要求使用 MFA。对于人类身份和工作负载，要求使用 MFA 是 IAM 最佳实践。启用 MFA 提供了一层额外的安全保障，这会要求用户提供登录凭证和一次性密码（OTP），或从硬件设备加密验证和生成的字符串。
- 强制执行最小密码长度，这是密码强度的主要因素。
- 强制执行密码复杂性，使密码更难以猜到。
- 允许用户更改自己的密码。
- 创建个人身份而不是共享凭证。通过创建个人身份，您可以为每个用户提供一组唯一的安全凭证。个人用户可以审计每个用户的活动。

IAM Identity Center 建议：

- IAM Identity Center 在使用默认目录时提供了预定义的密码策略，该策略确定了密码长度、复杂性和重用要求。
- 当身份源为默认目录、AWS Managed Microsoft AD 或 AD Connector 时，启用 MFA 并为 MFA 配置“背景认知”或“始终开启”设置。
- 允许用户注册自己的 MFA 设备。

Amazon Cognito user pools 目录建议：

- 配置密码长度设置。
- 对于用户，要求使用 MFA。
- 使用 Amazon Cognito user pools 高级安全设置实现自适应身份验证（可阻止可疑登录）等功能。

IAM 用户建议：

- 最好是使用 IAM Identity Center 或直接联合。然而，您可能需要 IAM 用户。在这种情况下，为 IAM 用户设置密码策略。您可以使用密码策略来定义诸如最小长度、密码是否需要非字母字符之类的要求。
- 创建 IAM 策略来强制执行 MFA 登录，以允许用户管理其自己的密码和 MFA 设备。

资源

相关最佳实践：

- SEC02-BP03 安全地存储和使用密钥

- SEC02-BP04 依赖集中式身份提供程序
- SEC03-BP08 在组织内安全地共享资源

相关文档：

- AWS IAM Identity Center（AWS Single Sign-On 的后继者）密码策略
- IAM 用户密码策略
- 设置 AWS 账户 根用户密码
- Amazon Cognito 密码策略
- AWS 凭证
- IAM 安全最佳实践

相关视频：

- 使用 AWS IAM Identity Center 大规模管理用户权限
- 在每个层面掌握身份

SEC02-BP02 使用临时凭证

进行任何类型的身份验证时，最好使用临时凭证而不是长期凭证，以降低或消除诸如凭证被无意泄露、共享或被盗之类的风险。

期望结果：为了降低长期凭证的风险，请尽可能对人类和机器身份使用临时凭证。长期凭证会带来许多风险，例如，它们能够以代码的形式上传到公有 GitHub 存储库。使用临时凭证可以显著降低凭证被泄露的几率。

常见反模式：

- 开发人员使用 IAM users 的长期访问密钥，而不是使用联合身份验证从 CLI 获得临时凭证。
- 开发人员在他们的代码中嵌入长期访问密钥，并将该代码上传到公有 Git 存储库。
- 开发人员在移动应用程序中嵌入长期访问密钥，然后在应用商店中提供这些密钥。
- 用户与其他用户共享长期访问密钥，或员工离开公司时仍持有长期访问密钥。
- 当可以使用临时凭证时，对机器身份使用长期访问密钥。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

对所有 AWS API 和 CLI 请求使用临时安全凭证，而不是长期凭证。在几乎所有情况下，对 AWS 服务的 API 和 CLI 请求都必须使用 AWS 访问密钥进行签名。这些请求可以使用临时凭证或长期凭证进行签名。只有在使用 IAM 用户或 AWS 账户 根用户时，才应该使用长期凭证（也称为长期访问密钥）。当您联合到 AWS 或通过其他方法代入 IAM 角色时，系统将生成临时凭证。即使您使用登录凭证访问 AWS Management Console，系统也会生成临时凭证供您调用 AWS 服务。需要用到长期凭证的情况很少，您可以使用临时凭证完成几乎所有任务。

尽量不要使用长期凭证，多用临时凭证，并且还应该减少采取 IAM 用户形式，多用联合身份验证和 IAM 角色形式进行访问。虽然 IAM 过去常使用用户来访问人类和机器身份，但我们现在建议不要使用它们，以避免使用长期访问密钥所带来的风险。

实施步骤

对于员工、管理员、开发人员、操作员和客户等人类身份：

- 您应该使用集中式身份提供者的服务，并要求人类用户配合使用联合身份验证与身份提供者两种方法，以使用临时凭证访问 AWS。可以通过直接联合到每个 AWS 账户，或使用 AWS IAM Identity Center（AWS IAM Identity Center 的后继者）和您选择的身份提供者，对用户进行联合身份验证。与使用 IAM 用户相比，联合身份验证除了消除使用长期凭证的情况之外，还具有许多优势。您的用户也可以从直接联合的命令行或通过使用 IAM Identity Center 来请求获得临时凭证。这意味着能够大幅减少需要使用 IAM 用户或用户长期凭证的情况。
- 授予软件即服务（SaaS）提供商等第三方访问 AWS 账户 中资源的权限时，您可以使用跨账户角色和基于资源的策略。
- 如果您需要批准消费者或客户申请访问您的 AWS 资源，可以使用 Amazon Cognito 身份池或 Amazon Cognito user pools 提供临时凭证。通过 IAM 角色配置凭证权限。 对于访客用户未通过身份验证的情况，您还可以定义一个拥有有限权限的单独 IAM 角色。

对于机器身份，您就可能需要使用长期凭证了。在这些情况下，您应该要求工作负载使用具有 IAM 角色的临时凭证来访问 AWS。

- 对于 Amazon Elastic Compute Cloud（Amazon EC2），您可以使用适用于 Amazon EC2 的角色。
- AWS Lambda 使您能够配置 Lambda 执行角色，以授权此服务利用临时凭证执行 AWS 操作。AWS 服务还有许多其他类似的模型，可以使用 IAM 角色授予临时凭证。
- 对于 IoT 设备，您可以使用 AWS IoT Core 凭证提供程序来请求临时凭证。

- 对于需要访问 AWS 资源的本地系统或在 AWS 之外运行的系统，您可以使用 IAM Roles Anywhere。

某些情况下不能选择临时凭证，此时您可能需要使用长期凭证。在这些情况下，可以定期审计和轮换凭证，并定期轮换访问密钥以解决需要使用长期凭证的情况。诸如使用 WordPress 插件和第三方 AWS 客户端等情况，都可能需要用到长期凭证。在必须使用长期凭证的情况下，或者对于并非 AWS 访问密钥的凭证（如数据库登录），您可以使用一种专门用于处理密钥管理的服务，比如 AWS Secrets Manager。借助 Secrets Manager，您可以使用支持的服务轻松管理、轮换和安全地存储加密密钥。有关轮换长期凭证的更多信息，请参阅轮换访问密钥。

资源

相关最佳实践：

- SEC02-BP03 安全地存储和使用密钥
- SEC02-BP04 依赖集中式身份提供程序
- SEC03-BP08 在组织内安全地共享资源

相关文档：

- 临时安全凭证
- AWS 凭证
- IAM 安全最佳实践
- IAM 角色
- IAM Identity Center
- 身份提供者和联合身份验证
- 轮换访问密钥
- 安全合作伙伴解决方案：访问和访问控制
- AWS 账户根用户

相关视频：

- 使用 AWS IAM Identity Center（AWS IAM Identity Center 的后继者）大规模管理用户权限
- 在每个层面掌握身份

SEC02-BP03 安全地存储和使用密钥

工作负载需要能够自动向数据库、资源和第三方服务证明其身份。这是使用秘密访问凭证（如 API 访问密钥、密码和 OAuth 令牌）完成的。使用专门构建的服务来存储、管理和轮换这些凭证有助于降低这些凭证泄露的可能性。

期望结果：实施安全管理应用程序凭证的机制，以实现以下目标：

- 确定工作负载需要哪些密钥。
- 尽可能用短期凭证代替长期凭证，从而减少所需的长期凭证的数量。
- 建立安全存储并自动轮换剩余的长期凭证。
- 审计对工作负载中存在的密钥的访问。
- 持续监控，以验证开发期间没有在源代码中嵌入任何密钥。
- 降低凭证被无意中泄露的可能性。

常见反模式：

- 不轮换凭证。
- 将长期凭证存储在源代码或配置文件中。
- 在未加密状态下静态存储凭证。

建立此最佳实践的好处：

- 对存储的凭证进行静态和传输中加密。
- 通过 API 来把关对凭证的访问（可将 API 看作凭证自动售货机）。
- 审计和记录对凭证的访问（包括读和写）。
- 关注点分离：凭证轮换由一个单独的组件执行，该组件可与架构的其余部分隔离开来。
- 密钥自动按需分发给软件组件，并在中心位置进行轮换。
- 可以精细地控制对凭证的访问。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

过去，用于对数据库、第三方 API、令牌和其他密钥进行身份验证的凭证可能已嵌入到源代码或环境文件中。AWS 提供了几种机制来安全地存储这些凭证，自动轮换它们，并审计它们的使用情况。

进行密钥管理的妥善方法是遵循相关指导，以正确地删除、替换和轮换密钥。最安全的凭证是您不必存储、管理或处理的凭证。某些凭证可能不再是正常运行工作负载所必需的，可以安全地删除。

对于仍然是正常运行工作负载所必需的凭证，可能有机会用临时或短期凭证替换长期凭证。例如，相比起对 AWS 秘密访问密钥进行硬编码，不妨考虑使用 IAM 角色，将长期凭证替换为临时凭证。

可能无法删除或替换某些长期密钥。这些密钥可以存储在 AWS Secrets Manager 等服务中，在那里可以集中存储、管理和定期轮换密钥。

对工作负载的源代码和配置文件进行审计，可以发现许多类型的凭证。下表总结了处理常见凭证类型的策略：

Credential type	Description	Suggested strategy
IAM access keys	AWS IAM access and secret keys used to assume IAM roles inside of a workload	Replace: Use IAM 角色 assigned to the compute instances (such as Amazon EC2 or AWS Lambda) instead. For interoperability with third parties that require access to resources in your AWS 账户, ask if they support AWS 跨账户访问. For mobile apps, consider using temporary credentials through Amazon Cognito 身份池（联合身份）. For workloads running outside of AWS, consider IAM Roles Anywhere or AWS Systems Manager 混合激活.
SSH keys	Secure Shell private keys used to log into Linux EC2	Replace: Use AWS Systems Manager or EC2

Credential type	Description	Suggested strategy
	instances, manually or as part of an automated process	Instance Connect to provide programmatic and human access to EC2 instances using IAM roles.
Application and database credentials	Passwords – plain text string	Rotate: Store credentials in AWS Secrets Manager and establish automated rotation if possible.
Amazon RDS and Aurora Admin Database credentials	Passwords – plain text string	Replace: Use the Secrets Manager 与 Amazon RDS 集成 or Amazon Aurora. In addition, some RDS database types can use IAM roles instead of passwords for some use cases (for more detail, see IAM 数据库身份验证).
OAuth tokens	Secret tokens – plain text string	Rotate: Store tokens in AWS Secrets Manager and configure automated rotation.
API tokens and keys	Secret tokens – plain text string	Rotate: Store in AWS Secrets Manager and establish automated rotation if possible.

一种常见的反模式是在源代码、配置文件或移动应用程序中嵌入 IAM 访问密钥。当需要 IAM 访问密钥与 AWS 服务通信时，请使用临时（短期）安全凭证。可以通过 EC2 实例的 IAM 角色、Lambda 功能的执行角色、移动用户访问的 Cognito IAM 角色和 IoT 设备的 IoT Core 策略提供这些短期凭证。与第三方进行交互时，最好将访问权限委托给 IAM 角色，以获得对您账户资源的必要访问权限，而不是配置 IAM 用户并向第三方发送该用户的秘密访问密钥。

在许多情况下，工作负载需要存储与其他服务和资源进行互操作所必需的密钥。AWS Secrets Manager 旨在安全地管理这些凭证，以及 API 令牌、密码和其他凭证的存储、使用和轮换。

AWS Secrets Manager 提供五个关键功能，以确保敏感凭证的安全存储和处理：静态加密、传输中加密、全面审计、精细访问控制和可扩展凭证轮换。AWS 合作伙伴提供的其他密钥管理服务或提供类似功能和保证的本地开发的解决方案也可以接受。

实施步骤

1. 使用自动化工具（如 Amazon CodeGuru）识别包含硬编码凭证的代码路径。
   - 使用 Amazon CodeGuru 扫描您的代码存储库。审查完成后，在 CodeGuru 中筛选 `Type=Secrets` 以查找有问题的代码行。
2. 识别可以删除或替换的凭证。
   a. 识别不再需要的凭证并标明要删除。
   b. 对于嵌入到源代码的 AWS 私有密钥，将其替换为与必要资源相关的 IAM 角色。如果您的部分工作负载在 AWS 之外，但需要 IAM 凭证才能访问 AWS 资源，请考虑 IAM Roles Anywhere 或 AWS Systems Manager 混合激活。
3. 对于其他需要使用轮换策略的第三方、长期密钥，请将 Secrets Manager 集成到代码中，以便在运行时检索第三方密钥。
   a. CodeGuru 控制台可以使用发现的凭证在 Secrets Manager 中自动创建密钥。
   b. 将 Secrets Manager 的密钥检索集成到应用程序代码中。
      - 无服务器 Lambda 功能可以使用与语言无关的 Lambda 扩展。
      - 对于 EC2 实例或容器，AWS 用几种流行的编程语言提供了示例客户端代码，用于从 Secrets Manager 检索密钥。
4. 定期检查代码库并重新扫描，以验证代码中没有添加新的密钥。
   - 考虑使用诸如 git-secrets 之类的工具来防止向源代码存储库提交新的密钥。
5. 监控 Secrets Manager 活动，以发现意外使用、不适当的密钥访问或试图删除密钥的迹象。
6. 减少人类接触凭证的机会。将读取、写入和修改凭证的权限仅授予专用于此目的的 IAM 角色，并仅向一小部分操作用户提供代入该角色的权限。

资源

相关最佳实践：

- SEC02-BP02 使用临时凭证
- SEC02-BP05 定期审计和轮换凭证

相关文档：

- 开始使用 AWS Secrets Manager
- 身份提供者和联合身份验证
- Amazon CodeGuru 推出 Secrets Detector
- AWS Secrets Manager 如何使用 AWS Key Management Service
- Secrets Manager 中的密钥加密和解密
- Secrets Manager 博客条目
- Amazon RDS 宣布与 AWS Secrets Manager 集成

相关视频：

- 有关大规模管理、检索和轮换密钥的最佳实践
- 使用 Amazon CodeGuru Secrets Detector 查找硬编码密钥
- 使用 AWS Secrets Manager 保护混合工作负载的密钥

相关研讨会：

- 在 AWS Secrets Manager 中存储、检索和管理敏感凭证
- AWS Systems Manager 混合激活

SEC02-BP04 依赖集中式身份提供程序

对于员工身份（员工和合同工），请依赖允许您在集中位置管理身份的身份提供程序。这样，您就可以更轻松地跨多个应用程序和系统管理访问权限，因为您在从单一位置创建、分配、管理、撤销和审核访问权限。

期望的结果： 您有一个集中式身份提供程序，可以在其中集中管理员工用户、身份验证策略 [例如要求多重身份验证（MFA）]，以及对系统和应用程序的授权（例如根据用户的群组成员资格或属性分配访问权限）。您的员工用户登录到中央身份提供程序并联合身份验证（单点登录）到内部和外部应用程序，这样用户就无需记住多个凭证。您的身份提供程序已与您的人力资源（HR）系统集成，因此人事变动会自动与身份提供程序同步。例如，如果有人离开您的组织，您可以自动撤消对联合应用程序和系统（包括 AWS）的访问权限。您已在身份提供程序中启用了详细的审核日志记录，并且正在监控这些日志以发现异常用户行为。

常见反模式：

- 您不使用联合身份验证和单点登录。您的员工用户在多个应用程序和系统中创建单独的用户账户和凭证。
- 您尚未实现员工用户身份生命周期的自动化，例如将您的身份提供程序与 HR 系统集成。当用户离开您的组织或变换角色时，您使用手动流程来删除或更新他们在多个应用程序和系统中的记录。

建立此最佳实践的好处： 通过使用集中式身份提供程序，您可以在一个位置管理员工用户身份和策略，可以向用户和群组分配应用程序的访问权限，还可以监控用户登录活动。通过与您的人力资源（HR）系统集成，当用户的角色发生更改时，这些更改会同步到身份提供程序，并自动更新为他们分配的应用程序和权限。当用户离职时，其身份将在身份提供程序中自动被禁用，从而撤消他们对联合应用程序和系统的访问权限。

未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

员工用户访问 AWS 的指南

员工用户（如组织中的员工和合同工）可能需要使用 AWS Management Console或 AWS Command Line Interface（AWS CLI）来访问 AWS，以履行其工作职能。您可以通过从集中式身份提供程序联合到 AWS，在两个层面上向员工用户授予 AWS 访问权：直接联合到每个 AWS 账户，或联合到 AWS 组织中的多个账户。

- 要将员工用户与每个 AWS 账户直接联合，可以使用集中式身份提供程序来联合到该账户中的 AWS Identity and Access Management 。IAM 的灵活性允许您启用单独的 SAML 2.0 或 Open ID Connect（OIDC） 身份提供程序（针对每个 AWS 账户），并使用联合用户属性进行访问控制。您的员工用户将使用网络浏览器，通过提供凭证（如密码和 MFA 令牌码）来登录身份提供程序。身份提供程序向其浏览器发出 SAML 断言，该断言将提交到 AWS Management Console 登录 URL，以允许用户 通过代入 IAM 角色单点登录到 AWS Management Console。用户还可以获取临时 AWS API 凭证以用于 AWS CLI 或 AWS SDK - 从 AWS STS ，方法是 使用身份提供程序的 SAML 断言代入 IAM 角色 。
- 要将员工用户与 AWS 组织中的多个账户联合起来，可以使用 AWS IAM Identity Center 来集中管理员工用户对 AWS 账户和应用程序的访问权限。您为组织启用 Identity Center 并配置身份源。IAM Identity Center 提供一个默认身份源目录，您可以用它来管理用户和组。或者，您也可以选择外部身份源，方法是使用 SAML 2.0 连接外部身份提供程序 并使用 SCIM 自动预置 用户和组，或 连接到您的 Microsoft AD 目录 （使用 AWS Directory Service）。配置身份源后，您可以通过以下方法为用户和组分配 AWS 账户访问权限：在 权限集中定义最低权限策略。您的员工用户可以通过您的中央身份提供程序进行身份验证，以登录 AWS 访问门户 并单点登录到 AWS 账户以及分配给他们的

云应用程序。您的用户可以配置 AWS CLI v2 以使用 Identity Center 进行身份验证，并获取用于运行 AWS CLI 命令的凭证。Identity Center 还允许通过单点登录访问 AWS 应用程序，例如 Amazon SageMaker Studio 和 AWS IoT Sitewise Monitor 门户。

遵循上述指导后，您的员工用户在 AWS 上管理工作负载时，将不再需要使用 IAM users和组来进行通用的操作。相反，您的用户和组将在 AWS 外部进行管理，并且用户可以作为 联合身份访问 AWS 资源。联合身份使用您的集中式身份提供程序定义的组。您应该识别并删除 AWS 账户中不再需要的 IAM 组、IAM users和长期用户凭证（密码和访问密钥）。 您可以 查找未使用的凭证 （使用 IAM 凭证报告）、 删除相应的 IAM users 和 删除 IAM 组。 您可以将 服务控制策略（SCP） 应用于您的组织，它有助于防止创建新的 IAM users和组，并强制通过联合身份访问 AWS。

应用程序用户指南

您可以通过将 Amazon Cognito 用作您的集中式身份提供程序，来管理应用程序（例如移动应用程序）用户的身份。Amazon Cognito 为您的 Web 和移动应用程序启用身份验证、授权和用户管理。Amazon Cognito 提供可扩展到数百万用户的身份存储，支持社交网络和企业身份联合验证，并提供高级安全功能来协助保护您的用户和业务。您可以将自定义 Web 或移动应用程序与 Amazon Cognito 集成，以便在几分钟内为您的应用程序添加用户身份验证和访问控制。Amazon Cognito 以 SAML 和 Open ID Connect（OIDC）等开放式身份标准为基础构建，支持各种合规性法规，并与前端和后端开发资源集成。

实施步骤

员工用户访问 AWS 的步骤

- 通过以下方法之一，使用集中式身份提供程序将您的员工用户联合身份验证到 AWS：
  - 通过与您的身份提供程序联合，使用 IAM Identity Center 来允许单点登录到您的 AWS 组织中的多个 AWS 账户。
  - 使用 IAM 将您的身份提供程序直接连接到每个 AWS 账户，从而实现精细的联合访问。
- 识别并移除被联合身份取代的 IAM users和群组。

适用于您的应用程序用户的步骤

- 将 Amazon Cognito 用作应用程序的集中式身份提供程序。
- 使用 OpenID Connect 和 OAuth 将您的自定义应用程序与 Amazon Cognito 集成。您可以使用 Amplify 库开发自定义应用程序，这些库提供了与各种 AWS 服务（例如用于身份验证的 Amazon Cognito）集成的简单接口。

资源

相关的 Well-Architected 最佳实践：

- SEC02-BP06 利用用户组和属性
- SEC03-BP02 授予最低访问权限
- SEC03-BP06 基于生命周期管理访问权限

相关文档：

- AWS 中的身份联合验证
- IAM 中的安全最佳实践
- AWS Identity and Access Management 最佳实践
- IAM Identity Center 委托管理入门
- 如何针对高级用例在 IAM Identity Center 中使用客户托管策略
- AWS CLI v2：IAM Identity Center 凭证提供程序

相关视频：

- AWS re:Inforce 2022 - AWS Identity and Access Management（IAM）深入探讨
- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Invent 2018: Mastering Identity at Every Layer of the Cake

相关示例：

- Workshop: Using AWS IAM Identity Center to achieve strong identity management
- Workshop: Serverless identity

相关工具：

- AWS 安全能力合作伙伴：身份和访问管理
- saml2aws

SEC02-BP05 定期审计和轮换凭证

定期审计和轮换凭证，以限制凭证可用于访问资源的时间。使用长期凭证会产生许多风险，可通过定期轮换来降低这些风险。

期望结果：实施凭证轮换，以帮助降低长期凭证相关风险。定期审计并纠正不符合凭证轮换策略的情况。

常见反模式：

- 不审计凭证的使用情况。
- 不必要地使用长期凭证。
- 使用长期凭证，不定期轮换。

在未建立这种最佳实践的情况下暴露的风险等级：中等

实施指导

当您无法依赖临时凭证并需要长期凭证时，请审计凭证，以验证实施了定义的控制措施（例如多重身份验证（MFA））并且定期轮换凭证，使凭证具有适当的访问级别。

（最好通过自动化工具）定期验证，以确保实施正确的控制措施。对于人员身份，您应要求用户定期更改他们的密码并弃用访问密钥，以支持临时凭证。从 AWS Identity and Access Management（IAM）用户转向集中式身份时，您可以生成凭证报告来审计您的用户。

我们还建议您在身份提供者中实施并监控 MFA。您可以设置 AWS Config 规则 或使用 AWS Security Hub 安全标准来监控用户是否启用了 MFA。考虑使用 IAM Roles Anywhere 为机器身份提供临时凭证。在无法使用 IAM 角色和临时凭证的情况下，需要经常审计和轮换访问密钥。

实施步骤

- 定期审计凭证：对您的身份提供者和 IAM 中配置的身份进行审计，这有助于验证只有经过授权的身份才能访问您的工作负载。此类身份可能包括但不限于 IAM 用户、AWS IAM Identity Center 用户、Active Directory 用户或不同上游身份提供者中的用户。例如，删除离开组织的人员，并删除不再需要的跨账户角色。制定流程，以定期审计 IAM 实体所访问服务的权限。这有助于您确定需要修改的策略，以删除任何未使用的权限。使用凭证报告和 AWS Identity and Access Management Access Analyzer 来审计 IAM 凭证和权限。您可以使用 Amazon CloudWatch 为 AWS 环境中调用的特定 API 调用设置警报。Amazon GuardDuty 还可以提醒您注意意外活动，出现这种提醒，可表明对 IAM 凭证的访问过于宽松，或出现了意外访问情况。

- 定期轮换凭证：当您无法使用临时凭证时，请定期轮换长期 IAM 访问密钥（最多每 90 天一次）。如果在您不知情的情况下无意中泄露了访问密钥，这将限制凭证用于访问资源的时间。有关轮换 IAM 用户的访问密钥的信息，请参阅轮换访问密钥。
- 审核 IAM 权限：为了提高您的 AWS 账户 的安全性，请定期审核和监控每个 IAM 策略。验证这些策略是否遵循最低权限原则。
- 考虑自动创建和更新 IAM 资源：IAM Identity Center 会自动执行许多 IAM 任务，比如角色和策略管理。或者，AWS CloudFormation 可用于自动部署 IAM 资源（包括角色和策略），以减少人为错误的机会，因为可以验证模板和控制版本。
- 对于机器身份，使用 IAM Roles Anywhere 替换 IAM 用户：IAM Roles Anywhere 将使您能够在传统上无法使用角色的领域（例如本地服务器）使用角色。IAM Roles Anywhere 使用可信的 X.509 证书向 AWS 进行身份验证并接收临时凭证。使用 IAM Roles Anywhere 便无需轮换这些凭证，因为长期凭证不再存储在本地环境中。请注意，您需要监控 X.509 证书，并在该证书即将到期时轮换它。

资源

相关最佳实践：

- SEC02-BP02 使用临时凭证
- SEC02-BP03 安全地存储和使用密钥

相关文档：

- 开始使用 AWS Secrets Manager
- IAM 最佳实践
- 身份提供者和联合身份验证
- 安全合作伙伴解决方案：访问和访问控制
- 临时安全凭证
- 获取 AWS 账户 的凭证报告

相关视频：

- 有关大规模管理、检索和轮换密钥的最佳实践
- 使用 AWS IAM Identity Center 大规模管理用户权限
- 在每个层面掌握身份

相关示例：

- Well-Architected 实验室 - 自动化 IAM 用户清理
- Well-Architected 实验室 - 自动部署 IAM 组和角色

SEC02-BP06 利用用户组和属性

随着您管理的用户数量不断增加，您需要确定如何组织这些用户，以便能够实现规模管理。将具有常见安全要求的用户置于由您的身份提供程序定义的组中，并建立机制以确保用于访问控制的用户属性（例如部门或位置）正确无误且已更新。使用这些组和属性（而不是单个用户）来控制访问权限。这样，您就可以通过使用 权限集一次性更改用户的组成员身份或属性来集中管理访问，而不是在需要更改用户的访问权限时更新多个单独策略。您可以使用 AWS IAM Identity Center（IAM Identity Center）来管理用户组和属性。IAM Identity Center 支持最常用的属性，无论是在创建用户时手动输入的属性还是使用同步引擎自动预置的属性，例如跨域身份管理系统（SCIM，Cross-Domain Identity Management）规范中定义的那些属性。

将具有常见安全要求的用户置于由您的身份提供程序定义的组中，并建立机制以确保用于访问控制的用户属性（例如部门或位置）正确无误且已更新。使用这些组和属性（而不是单个用户）来控制访问。这使您可以通过一次性更改用户的组成员身份或属性来集中管理访问，而不是在用户的访问需要更改时更新多个单独策略。

未建立这种最佳实践的情况下暴露的风险等级： 低

实施指导

- 如果您在使用 AWS IAM Identity Center（IAM Identity Center）配置组：IAM Identity Center 使您能够配置用户组，并为组分配所需的权限级别。
  - AWS Single Sign-On – 管理身份
- 了解基于属性的访问控制（ABAC，Attribute-Based Access Control）：ABAC 是一种基于属性定义权限的授权策略。
  - 什么是适用于 AWS 的 ABAC？
  - 实验室：基于 IAM 标签的 EC2 访问控制

资源

相关文档：

- 开始使用 AWS Secrets Manager
- IAM 最佳实践
- 身份提供程序和联合
- AWS 账户根用户

相关视频：

- 有关大规模管理、检索和轮换密钥的最佳实践
- 使用 AWS IAM Identity Center 大规模管理用户权限
- 在每个层面掌握身份

相关示例：

- 实验室：基于 IAM 标签的 EC2 访问控制

权限管理

管理权限，以控制需要访问 AWS 和您的工作负载的人员和机器身份的访问权限。权限用于控制哪些人可以在什么条件下访问哪些内容。为特定的人员身份和机器身份设置权限，以授权他们/它们访问特定资源上的特定服务操作。此外，为要授予的访问权限指定必须满足的条件。例如，您可以允许开发人员创建新的 Lambda 函数，但只能在特定的区域中创建。当大规模管理您的 AWS 环境时，请遵循以下最佳实践，以确保身份只拥有其需要的访问权限，而没有任何多余的权限。

可通过多种方式向不同类型的资源授予访问权限。一种方式是使用不同的策略类型。

IAM 中基于身份的策略是托管策略与内联策略，并且会附加到 IAM 身份（包括用户、组或角色）。这些策略允许您指定该身份可以执行的操作（其权限）。基于身份的策略可以进一步分类。

托管策略 – 基于身份的独立策略，可附加到 AWS 账户中的多个用户、组和角色。有两种类型的托管策略：

- AWS 托管策略 – 由 AWS 创建和管理的托管策略。
- 客户托管策略 – 您在 AWS 账户中创建和管理的托管策略。与 AWS 托管策略相比，客户托管策略对策略的控制更精确。

托管策略是应用权限的首选方法。不过，您也可以使用直接添加到单个用户、组或角色的内联策略。内联策略在策略和身份之间维护严格的一对一关系。删除身份时将删除内联策略。

在大多数情况下，您应按照最低权限原则创建自己的客户托管策略。

基于资源的策略会附加到资源。例如，S3 存储桶策略是一个基于资源的策略。这些策略向一个主体授予权限，该主体既可以位于资源所在的账户中，也可以位于另一个账户中。要查看支持基于资源的策略的服务列表，请参阅使用 IAM 的 AWS 服务。

权限边界使用托管策略来设置管理员能够设置的最高权限。这样，您就可以为开发人员赋予创建和管理权限的能力，例如创建一个 IAM 角色，但限制他们可以授予的权限，以使他们无法利用他们创建的角色提升自己的权限。

基于属性的访问控制（ABAC） 使您能够基于属性授予权限。在 AWS 中，这些属性称为标签。标签可以附加到 IAM 主体（用户或角色）和 AWS 资源。使用 IAM policy 时，管理员可以创建一个可重复使用的策略，以根据 IAM 主体的属性来应用权限。例如，作为管理员，您可以使用一个 IAM policy，授权您所在组织中的开发人员访问与这些开发人员的项目标签匹配的 AWS 资源。当这一组开发人员为项目添加资源时，会自动根据属性应用权限。这样就无需为每个新资源执行策略更新。

Organizations 服务控制策略（SCP） 为组织或组织单位（OU）的账户成员定义最大权限。SCP 限制基于身份的策略或基于资源的策略授予账户内实体（用户或角色）的权限，但不授予权限。

会话策略代入角色或联合用户。在使用 AWS CLI 或 AWS API 会话策略限制基于角色或用户身份的策略授予会话的权限时，传递会话策略。这些策略限制已创建会话的权限，但不授予权限。有关更多信息，请参阅会话策略。

最佳实践

- SEC03-BP01 定义访问要求
- SEC03-BP02 授予最低访问权限
- SEC03-BP03 建立紧急访问流程
- SEC03-BP04 持续减少权限
- SEC03-BP05 为您的组织定义权限防护机制
- SEC03-BP06 基于生命周期管理访问权限
- SEC03-BP07 分析公共和跨账户访问
- SEC03-BP08 在组织内安全地共享资源
- SEC03-BP09 与第三方安全地共享资源

SEC03-BP01 定义访问要求

管理员、最终用户或其他组件都需要访问您工作负载的每个组件或资源。明确定义哪些人员或事物应当有权访问每个组件，选择用于进行身份验证和授权的适当身份类型和方法。

常见反模式：

- 在应用程序中进行硬编码或存储密码。
- 向每个用户授予自定义权限。
- 使用长期有效的凭证。

未建立这种最佳实践的情况下暴露的风险等级： 高

实施指导

管理员、最终用户或其他组件都需要访问您工作负载的每个组件或资源。明确定义哪些人员或事物应当有权访问每个组件，选择用于进行身份验证和授权的适当身份类型和方法。

应提供对组织内 AWS 账户 的常规访问，方法是使用 联合身份访问 或集中式身份提供者。您还应将身份管理集中处理，确保对于 AWS 将访问集成到员工访问生命周期中已建立了既定做法。例如，当员工转岗到具有不同访问级别的职位时，该员工的小组成员资格也应进行更改以反映新的访问要求。

在定义非人类身份的访问要求时，请确定哪些应用程序和组件需要访问权限以及如何向其授予权限。建议使用通过最低权限访问模型构建的 IAM 角色。AWS 托管策略 提供了预定义的 IAM 策略，这些策略涵盖了大多数常见使用案例。

AWS 服务（例如 AWS Secrets Manager 和 AWS Systems Manager Parameter Store）可以帮助在无法使用 IAM 角色的情况下，安全地将密码与应用程序或工作负载分离。在 Secrets Manager 中，您可以为凭证建立自动轮换。您可以通过 Systems Manager 使用您在创建参数时指定的唯一名称，来引用脚本、命令、SSM 文档、配置和自动化工作流中的参数。

您可以使用 AWS Identity and Access Management Roles Anywhere 获取 IAM 中的临时安全凭证，这种凭证适用于在 AWS 外部运行的工作负载。您的工作负载可以使用 IAM 策略 和 IAM 角色 ，也就是您为访问 AWS 资源在 AWS 应用程序中所用的策略和角色。

如果可能，请优先选择短期临时凭证而不是长期静态凭证。在一些场景中，需要具有编程访问权限和长期凭证的 IAM 用户，此时请使用 访问密钥上次使用的信息 来轮换和删除访问密钥。

资源

相关文档：

- 基于属性的访问控制（ABAC）
- AWS IAM Identity Center
- IAM Roles Anywhere
- 适用于 IAM Identity Center 的 AWS 托管策略
- AWS IAM 策略条件
- IAM 使用案例
- 删除不必要的凭证
- 策略的使用
- 如何根据 AWS 账户、OU 或组织来控制对 AWS 资源的访问
- 使用 AWS Secrets Manager 中的增强搜索来轻松标识、安排和管理密钥

相关视频：

- 在 60 分钟以内成为 IAM 策略高手
- 职责分离、最低权限、委托和 CI/CD
- 简化身份和访问管理以实施创新

SEC03-BP02 授予最低访问权限

最佳实践是向身份授予的访问权限只能是在特定条件下对特定资源执行特定操作所必需的。使用组和身份属性来大规模动态设置权限，而不是为单个用户定义权限。例如，您可以允许一组开发人员访问，以便仅管理其项目的资源。使用这种方法，如果某个开发人员离开项目，则可以自动撤销该开发人员的访问权限，而无需更改底层访问策略。

期望结果：用户应仅具有完成工作所必需的权限。应该仅向用户授予访问生产环境以在有限的时间段内执行特定任务的权限，并且在任务完成后，应撤销访问权限。当不再需要权限时（包括当用户转到其他项目或工作职能时），应撤销权限。管理员权限应该只授予一小部分受信任的管理员。应定期审查权限，以避免权限蔓延。应该为机器或系统账户授予一组完全其任务所需的最低权限。

常见反模式：

- 默认为向用户授予管理员权限。
- 使用根用户进行日常活动。
- 创建过于宽松但没有完全管理员权限的策略。

- 不审查权限以了解是否为它们授予了最低访问权限。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

最低权限原则指出，应仅允许身份执行完成特定任务所需的一组最少操作。这样使可用性、效率和安全性达到平衡。根据此原则运营，有助于限制意外访问，还有助于跟踪谁有权访问哪些资源。默认情况下，IAM 用户和角色没有任何权限。根用户默认具有完全访问权限，所以应该严格控制和监控根用户，并且只用于需要根访问权限的任务。

IAM 策略用于显式地为 IAM 角色或特定资源授予权限。例如，基于身份的策略可以附加到 IAM 组，而 S3 存储桶可由基于资源的策略控制。

创建 IAM 策略时，您可以指定服务操作、资源以及为使 AWS 允许或拒绝访问而必须满足的条件。AWS 支持多种条件以帮助您缩小访问权限范围。例如，通过使用 `PrincipalOrgID` 条件键，如果请求者不属于 AWS Organization，则您可以拒绝操作。

您还可以控制 AWS 服务代表您发出的请求，例如要求 AWS CloudFormation 使用 `CalledVia` 条件键创建 AWS Lambda 函数。您应该对不同的策略类型进行分层，以建立深度防御并限制用户的总体权限。您还可以限制可以授予哪些权限以及在什么条件下授予权限。例如，您可以允许应用程序团队为他们构建的系统创建他们自己的 IAM 策略，但还必须应用权限边界来限制系统可以接收的最大权限。

实施步骤

- 实施最低权限策略：向 IAM 组和角色分配具有最低权限的访问策略，以反映所定义的用户角色或职能。
  - 将策略基于 API 使用情况：确定所需权限的一种方法是查看 AWS CloudTrail 日志。这样就使您可以根据用户在 AWS 内实际执行的操作来创建权限。IAM Access Analyzer 会自动基于活动生成 IAM 策略。您可以在组织或账户级别使用 IAM Access Advisor 来跟踪特定策略的上次访问信息。
- 考虑使用针对工作职能的 AWS 托管策略。 开始创建细粒度权限策略时，很难知道从哪里开始。AWS 拥有针对常见工作角色（例如计费、数据库管理员和数据科学家）的托管策略。这些策略可以帮助缩减用户具备的访问权限，同时确定如何实施最低权限策略。
- 删除不必要的权限：删除不需要的权限，并削减过于宽松的策略。IAM Access Analyzer 策略生成可帮助微调权限策略。
- 确保用户对生产环境仅具有有限的访问权限：用户应该只能访问具有有效使用案例的生产环境。在用户执行需要生产访问权限的特定任务后，应撤销访问权限。限制对生产环境的访问可帮助防止发生影响生产的意外事件，并缩小意外访问的影响范围。

- 考虑使用权限边界：权限边界是一项功能，它使用托管策略设置基于身份的策略可向 IAM 实体授予的最高权限。实体的权限边界仅允许实体执行其基于身份的策略及其权限边界都允许的操作。
- 考虑权限的资源标签：借助使用资源标签且基于属性的访问控制模型，您可以根据资源用途、所有者、环境或其他条件授予访问权限。例如，您可以使用资源标签来区分开发环境和生产环境。您可以使用这些标签将开发人员限制在开发环境中。通过将标记与权限策略结合在一起，您可以实现细粒度的资源访问，而无需为每个工作职能定义复杂的自定义策略。
- 为 AWS Organizations 使用服务控制策略。 服务控制策略集中控制组织中成员账户的最大可用权限。重要的是，您可以使用服务控制策略来限制成员账户中的根用户权限。还要考虑使用 AWS Control Tower，它提供可充实 AWS Organizations 的规范性托管控制。您还可以在 Control Tower 内定义自己的控制。
- 为组织建立用户生命周期策略：用户生命周期策略定义了当用户加入 AWS、更改工作角色或范围，或不再需要访问 AWS 时需要执行的任务。应在用户生命周期的每个步骤中执行权限审查，以确认权限受到适当限制并避免权限蔓延。
- 确立定期的时间表来审查权限并删除任何不需要的权限：您应定期审查用户访问权限，以确认用户不具有过于宽松的访问权限。在审核用户权限时 AWS Config 和 IAM Access Analyzer 可以提供帮助。
- 确立工作角色矩阵：工作角色矩阵形象地展示 AWS 业务覆盖范围内所需的各种角色和访问级别。使用工作角色矩阵，您可以根据用户在组织内的职责来定义和分离权限。使用组而不是将权限直接应用于单个用户或角色。

资源

相关文档：

- 授予最低权限
- IAM 实体的权限边界
- 用于编写最低权限 IAM 策略的方法
- 通过基于访问活动生成 IAM 策略，IAM Access Analyzer 可让您更轻松实施最低权限
- 使用 IAM 权限边界将权限管理委派给开发人员
- 使用上次访问的信息来细化权限
- IAM 策略类型及其使用时间
- 使用 IAM 策略模拟器测试 IAM 策略
- AWS Control Tower 中的防护机制
- 零信任架构：AWS 视角
- 如何使用 CloudFormation StackSets 实施最低权限原则

- 基于属性的访问控制（ABAC）
- 通过查看用户活动缩小策略范围
- 查看角色访问权限
- 使用标记来整理环境和促进责任的履行
- AWS 标记策略
- 标记 AWS 资源

相关视频：

- 新一代权限管理
- 零信任：AWS 视角
- 如何使用权限边界限制用户和角色以防止权限升级？

相关示例：

- 实验室：创建 IAM 权限边界委派角色
- 实验室：基于 IAM 标签的 EC2 访问控制

SEC03-BP03 建立紧急访问流程

创建一个流程，便于在集中式身份提供程序偶尔出现问题时紧急访问您的工作负载。

必须针对可能导致紧急事件的不同故障模式设计流程。例如，在正常情况下，您的员工用户使用集中式身份提供程序联合到云端（SEC02-BP04）来管理他们的工作负载。但是，如果您的集中式身份提供程序出现故障，或者云中联合身份验证的配置被修改，则您的员工用户可能无法联合到云中。紧急访问流程允许授权管理员通过其他方式（例如其他联合形式或直接用户访问）访问云资源，以解决联合配置或工作负载的问题。在恢复正常的联合机制之前，将使用紧急访问流程。

期望的结果：

- 您已经定义并记录了算是紧急情况的故障模式：考虑您的正常情况以及用户管理其工作负载所依赖的系统。考虑这些依赖项中的每一个在哪些情形下会发生故障并导致紧急情况。您可能会发现，可靠性支柱 中的问题和最佳实践有助于识别故障模式和构建更具韧性的系统，从而最大限度地降低发生故障的可能性。
- 您已记录了将故障确认为紧急情况所必须遵循的步骤。例如，您可以要求身份管理员检查主身份提供程序和备用身份提供程序的状态，如果两者均不可用，则宣布身份提供程序故障为紧急事件。

- 您已针对每种紧急情况或故障模式定义了紧急访问流程。应尽可能明确具体，这样可减少用户针对所有类型的紧急情况过度使用通用流程的倾向。紧急访问流程描述了每个流程的使用情形，以及哪些情况下不应使用该流程，并指出了可能适用的替代流程。
- 您的流程有详细的说明和行动手册，便于快速有效地遵循。请记住，对用户来说，紧急事件可能让人很煎熬，他们可能面临极大的时间压力，因此流程设计应尽可能简单。

常见反模式：

- 您没有详细记录并经过充分测试的紧急访问流程。您的用户没有为紧急情况做好准备，在出现紧急事件时遵循临时流程。
- 您的紧急访问流程依赖于与普通访问机制相同的系统（例如集中式身份提供程序）。这意味着，此类系统的故障可能会同时影响您的正常访问和紧急访问机制，并削弱您从故障中恢复的能力。
- 您的紧急访问流程被用于非紧急情况。例如，您的用户经常滥用紧急访问流程，因为他们发现直接进行更改比通过管道提交更改更容易。
- 您的紧急访问流程未生成足够的日志用于审核这些流程，或者没有监控日志以提醒可能存在的流程滥用。

建立此最佳实践的好处：

- 通过拥有记录详实且经过充分测试的紧急访问流程，您可以减少用户响应和解决紧急事件所花费的时间。这可以缩短停机时间，提高您向客户提供的服务的可用性。
- 您可以跟踪每个紧急访问请求，检测未经授权企图对非紧急事件滥用该过程的行为，并发出警报。

未建立这种最佳实践的情况下暴露的风险等级：中

实施指导

本节针对与部署在 AWS 上的工作负载相关的几种故障模式，提供创建紧急访问流程的指导，首先是适用于所有故障模式的通用指导，然后是不同故障模式类型的特定指导。

适用于所有故障模式的通用指南

在针对故障模式设计紧急访问流程时，请考虑以下几点：

- 记录流程的先决条件和假设：何时应使用该流程、何时不应使用该流程。它有助于详细说明故障模式并记录假设，例如其他相关系统的状态。例如，故障模式 2 的流程假设身份提供程序可用，但 AWS 上的配置已修改或已过期。

- 预先创建紧急访问流程所需的资源（SEC10-BP05）。例如，预先创建带有 IAM users和角色的紧急访问 AWS 账户，并在所有工作负载账户中创建跨账户 IAM 角色。这可以验证在发生紧急事件时这些资源是否已准备就绪并且可用。通过预先创建资源，您就不必依赖 AWS 控制平面 API（用于创建和修改 AWS 资源），在紧急情况下，这些 API 可能不可用。此外，通过预先创建 IAM 资源，您也无需考虑 由于最终的一致性问题而可能出现的延迟。
- 将紧急访问流程作为事件管理计划的一部分（SEC10-BP02）。记录如何跟踪紧急事件并将其传达给组织中的其他人，例如同级团队、您的领导层，如果适用，还包括外部的客户和业务合作伙伴。
- 在现有的服务请求工作流程系统（如果有）中定义紧急访问请求流程。通常，此类工作流程系统允许您创建受理表来收集有关请求的信息，在工作流程的每个阶段跟踪请求，并添加自动和手动审批步骤。将每个请求与事件管理系统中所跟踪的相应紧急事件关联起来。通过拥有统一的紧急访问系统，您可以在单个系统中跟踪这些请求，分析使用趋势并改进流程。
- 确保您的紧急访问流程只能由授权用户启动，并且需要用户的同事或管理层（视情况而定）的批准。审批流程在工作时间内和工作时间之外应该能够有效运作。明确在主审批人抽不开身的情况下，如何允许辅助审批人审批请求，并沿管理链条上报，直至获得批准。
- 验证流程是否会针对成功和失败的紧急访问尝试，生成详细的审计日志和事件。监控请求流程和紧急访问机制，以检测滥用或未经授权的访问。将活动与事件管理系统中正在发生的紧急事件关联起来，并在预期时间段之外执行操作时发出警报。例如，您应监控紧急访问 AWS 账户中的活动并发出警报，因为在正常操作过程中绝不应使用该账户。
- 定期测试紧急访问流程，以确保步骤清楚明了，并快速高效地授予正确的访问级别。您的紧急访问流程应作为事件响应模拟（SEC10-BP07）和灾难恢复测试（REL13-BP03）的一部分进行测试。

故障模式 1：用于联合到 AWS 的身份提供程序不可用

如 SEC02-BP04 依赖集中式身份提供程序中所述，我们建议依靠集中式身份提供程序，来联合您的员工用户以授予对 AWS 账户的访问权限。您可以使用 IAM Identity Center 联合到 AWS 组织中的多个 AWS 账户，也可以使用 IAM 将联合到单个 AWS 账户。在这两种情况下，员工用户都要先通过集中式身份提供程序进行身份验证，然后才会被重定向到 AWS 登录端点进行单点登录。

万一集中式身份提供程序不可用，员工用户就无法联合到 AWS 账户或管理其工作负载。在这种紧急情况下，您可以为一小部分管理员提供紧急访问流程，让他们访问 AWS 账户，来执行等不及集中式身份提供程序恢复正常的关键任务。例如，您的身份提供程序在 4 小时内不可用，在此期间，您需要修改生产账户中 Amazon EC2 Auto Scaling 组的上限，以应对客户流量意外激增的情况。您的紧急状况管理员应遵循紧急访问流程，以获得对特定生产 AWS 账户的访问权限并进行必要的更改。

紧急访问流程依赖于预先创建的紧急访问 AWS 账户，该账户仅用于紧急访问，并拥有 AWS 资源（如 IAM 角色和 IAM users）以支持紧急访问流程。在正常运营期间，任何人都不得访问紧急访问账户，而且您必须对滥用该账户的行为进行监控并发出警报（有关更多详情，请参阅前面的“通用指南”部分）。

紧急访问账户具有紧急访问 IAM 角色，有权在需要紧急访问的 AWS 账户中代入跨账户角色。这些 IAM 角色是预先创建的，并配置有信任策略，可信任应急账户的 IAM 角色。

紧急访问过程可以使用以下方法之一：

- 您可以在紧急访问账户中为紧急状况管理员预先创建一组 IAM users ，并使用相关的强密码和 MFA 令牌。这些 IAM users有权代入 IAM 角色，然后在需要紧急访问时，允许跨账户访问 AWS 账户。我们建议尽可能少地创建此类用户，并将每个用户分配给一个紧急状况管理员。在紧急情况下，紧急状况管理员用户使用其密码和 MFA 令牌码登录紧急访问账户，切换到紧急账户中的紧急访问 IAM 角色，最后切换到工作负载账户中的紧急访问 IAM 角色，以执行紧急更改操作。这种方法的优点是，每个 IAM user都分配给一个紧急状况管理员，您可以通过查看 CloudTrail 事件来了解哪个用户已登录。缺点是，您必须维护多个 IAM users及其关联的长寿命密码和 MFA 令牌。
- 您可以使用紧急访问 AWS 账户根用户 来登录紧急访问账户，代入用于紧急访问的 IAM 角色，并代入工作负载账户中的跨账户角色。建议为根用户设置一个强密码和多个 MFA 令牌。我们还建议将密码和 MFA 令牌存储在安全的企业凭证保管库中，该保管库可执行强身份验证和授权。您应确保密码和 MFA 令牌重置因素的安全：将账户的电子邮件地址设置为由云安全管理员监控的电子邮件分发列表，将账户的电话号码设置为同样由安全管理员监控的共享电话号码。这种方法的优点是只需管理一组根用户凭证。缺点是，由于这是共享用户，多个管理员都能以根用户身份登录。您必须审计企业保管库日志事件，以确定是哪位管理员查看了根用户密码。

故障模式 2：AWS 上的身份提供程序配置已修改或已过期

要允许您的员工用户联合到 AWS 账户，您可以使用外部身份提供程序配置 IAM Identity Center，或创建 IAM 身份提供程序（SEC02-BP04）。通常，您需要通过导入身份提供程序提供的 SAML 元数据 XML 文档，来配置这些服务。元数据 XML 文档包含一个 X.509 证书，该证书对应于身份提供程序用来签署其 SAML 断言的私钥。

管理员可能会错误地修改或删除 AWS 端的这些配置。在另一种情形下，导入到 AWS 的 X.509 证书可能会过期，而带有新证书的新元数据 XML 尚未导入到 AWS。这两种情形都可能使您的员工用户无法联合到 AWS，从而出现紧急情况。

在这种紧急情况下，您可以向您的身份管理员提供对 AWS 的访问权限以修复联合问题。例如，身份管理员使用紧急访问流程登录紧急访问 AWS 账户，切换到 Identity Center 管理员账户中的角色，并通过从身份提供程序导入最新的 SAML 元数据 XML 文档来更新外部身份提供程序配置，从而重新启用联合。修复联合后，您的员工用户将继续使用正常操作流程联合到其工作负载账户。

您可以按照前面的故障模式 1 中详述的方法来创建紧急访问流程。您可以向您的身份管理员授予最低访问权限，使其只能访问 Identity Center 管理员账户，并使用该账户对 Identity Center 执行操作。

故障模式 3：Identity Center 中断

如果发生 IAM Identity Center 或 AWS 区域中断这样的小概率事件，我们建议您设置一个可用于临时访问 AWS Management Console的配置。

紧急访问流程使用从身份提供程序到您的紧急账户中的 IAM 的直接联合。有关流程和设计注意事项的详细信息，请参阅 Set up emergency access to the AWS Management Console访问 AWS 资源。

实施步骤

针对所有故障模式的通用步骤

- 创建专门用于紧急访问流程的 AWS 账户。预先创建账户中所需的 IAM 资源，例如 IAM 角色或 IAM users，以及可选的 IAM 身份提供程序。此外，在工作负载 AWS 账户中预先创建跨账户 IAM 角色，并与紧急访问账户中的相应 IAM 角色建立信任关系。您可以将 AWS CloudFormation StackSets 与 AWS Organizations 结合使用，在您组织的成员账户中创建此类资源。

- 创建 AWS Organizations 服务控制策略（SCP），以拒绝删除和修改成员 AWS 账户中的跨账户 IAM 角色。
- 对紧急访问 AWS 账户启用 CloudTrail，并将跟踪事件发送到日志收集 AWS 账户中的中央 S3 存储桶。如果您使用 AWS Control Tower 来设置和管理您的 AWS 多账户环境，则您使用 AWS Control Tower 创建或在 AWS Control Tower 中注册的每个账户默认情况下都已启用 CloudTrail，并发送到专用日志存档 AWS 账户中的 S3 存储桶。
- 通过创建 EventBridge 规则来匹配紧急 IAM 角色所执行的控制台登录和 API 活动，监控紧急访问账户的活动。当事件管理系统中所跟踪的正在发生的紧急事件之外出现活动时，向安全运营中心发送通知。

针对“故障模式 1：用于联合到 AWS 的身份提供程序不可用”和“故障模式 2：AWS 上的身份提供程序配置已修改或已过期”的其他步骤

- 根据您选择的紧急访问机制，预先创建资源：
  - 使用 IAM users： 使用强密码和关联的 MFA 设备预先创建 IAM users。
  - 使用紧急账户的根用户： 为根用户配置一个强密码，并将该密码存储在您的企业凭证库中。将多个物理 MFA 设备与根用户关联，并将设备存放在紧急状况管理员团队成员可以快速访问的位置。

针对“故障模式 3：Identity Center 中断”的其他步骤

- 如 Set up emergency access to the AWS Management Console中所详述的那样，在紧急访问 AWS 账户中，创建 IAM 身份提供程序，以启用从身份提供程序的直接 SAML 联合。
- 在 IdP 中创建没有成员的紧急行动组。
- 在紧急访问账户中创建与紧急行动组相对应的 IAM 角色。

资源

相关的 Well-Architected 最佳实践：

- SEC02-BP04 依赖集中式身份提供程序
- SEC03-BP02 授予最低访问权限
- SEC10-BP02 制定事件管理计划
- SEC10-BP07 执行实际演练

相关文档：

- Set up emergency access to the AWS Management Console
- Enabling SAML 2.0 federated users to access the AWS Management Console
- Break glass access

相关视频：

- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Inforce 2022 - AWS Identity and Access Management（IAM）深入探讨

相关示例：

- AWS Break Glass 角色
- AWS 客户行动手册框架
- AWS 事件响应行动手册样本

SEC03-BP04 持续减少权限

当您的团队确定好所需的访问权限时，删除不需要的权限，并建立审核流程以实现最低权限。持续监控并删除供人类和机器访问的未使用的身份和权限。

期望结果：权限策略应遵循最低权限原则。随着工作职责和角色变得更加明确，需要审查您的权限策略以删除不必要的权限。如果无意中泄露或未经授权访问凭证，这种方法会缩小影响范围。

常见反模式：

- 默认为向用户授予管理员权限。
- 创建过于宽松但没有完全管理员权限的策略。
- 保留不再需要的权限策略。

在未建立这种最佳实践的情况下暴露的风险等级：中等

实施指导

当团队和项目刚刚起步时，可以使用宽松的权限策略来激发创新并提高敏捷性。例如，在开发或测试环境中，开发人员可以获得广泛的访问权限以使用各种 AWS 服务。我们建议您持续评估访问权限，并仅限于访问完成当前作业所必需的服务和服务操作。对于人类和机器身份，均建议进行此项评估。机器身份有时称为系统或服务账户，是让 AWS 访问应用程序或服务器的身份。这种访问权限在生产环境中尤其重要，因为在该环境中，过于宽松的权限会产生广泛的影响，并可能暴露客户数据。

AWS 提供多种方法来帮助识别未使用的用户、角色、权限和凭证。AWS 还可帮助分析 IAM 用户和角色（包括关联的访问密钥）的访问活动，以及对 AWS 资源（如 Amazon S3 存储桶中的对象）的访问。AWS Identity and Access Management Access Analyzer 策略生成可帮助您根据主体与之交互的实际服务和操作来创建限制性权限策略。基于属性的访问控制（ABAC）可帮助简化权限管理，因为您可以使用用户的属性为用户提供权限，而不是将权限策略直接附加到每个用户。

实施步骤

- 使用 AWS Identity and Access Management Access Analyzer：IAM Access Analyzer 可帮助识别您组织和账户中与外部实体共享的资源，例如 Amazon Simple Storage Service（Amazon S3）存储桶或 IAM 角色。
- 使用 IAM Access Analyzer 策略生成：IAM Access Analyzer 策略生成可帮助您基于 IAM 用户或角色的访问活动创建精细的权限策略。
- 为 IAM 用户和角色确定可接受的时间框架和使用策略：使用上次访问时间戳来识别未使用的用户和角色并将它们移除。查看关于服务和操作的上次访问情况的信息，并确定特定用户和角色的权限范围。例如，您可以使用关于上次访问情况的信息，确定您的应用程序角色需要执行的特定 Amazon S3 操作，并只允许该角色访问这些操作。AWS Management Console 中提供了上次获取的信息，您也可以对这些功能进行编程，以便将它们整合到您的基础设施工作流程和自动化工具中。

- 考虑将数据事件录入 AWS CloudTrail：默认情况下，CloudTrail 不会记录数据事件，例如 Amazon S3 对象级活动（如 `GetObject` 和 `DeleteObject`）或 Amazon DynamoDB 表活动（如 `PutItem` 和 `DeleteItem`）。考虑为这些事件启用日志记录，以确定哪些用户和角色需要访问特定的 Amazon S3 对象或 DynamoDB 表项目。

资源

相关文档：

- 授予最低特权
- 删除不必要的凭证
- 什么是 AWS CloudTrail？
- 策略的使用
- 日志记录和监控 DynamoDB
- 为 Amazon S3 存储桶和对象启用 CloudTrail 事件日志记录
- 获取 AWS 账户 的凭证报告

相关视频：

- 在 60 分钟以内成为 IAM 策略高手
- 职责分离、最低权限、委托和 CI/CD
- AWS re:Inforce 2022 - AWS Identity and Access Management（IAM）深入探讨

SEC03-BP05 为您的组织定义权限防护机制

建立通用控件以限制对组织中所有身份的访问。例如，您可以限制对特定 AWS 区域 的访问，或防止操作员删除通用资源，例如用于您的核心安全团队的 IAM 角色。

常见反模式：

- 在组织管理员账户中运行工作负载。
- 在同一账户中运行生产工作负载和非生产工作负载。

未建立这种最佳实践的情况下暴露的风险等级： 中

实施指导

当您在 AWS 中的工作负载增多并管理这些额外的工作负载时，您应使用账户分离这些工作负载，并使用 AWS Organizations 管理这些账户。我们建议您建立常用权限防护机制，以限制您所在组织中的所有身份的访问权限。例如，您可以限制对特定 AWS 区域 的访问，或防止您的团队删除常见资源，例如您的核心安全团队使用的 IAM 角色。

您可以首先实施示例服务控制策略，例如禁止用户禁用密钥服务。SCP 使用 IAM 策略语言，并允许您建立所有 IAM 主体（用户和角色）都要遵循的控制机制。您可以限制对特定服务操作和资源的访问，并根据特定的条件限制访问，以满足您所在组织的访问控制需求。如有必要，您可以为您的防护机制定义异常情况。例如，您可以为账户中除特定管理员角色以外的所有 IAM 实体限制服务操作。

我们建议您避免在管理账户中运行工作负载。应该使用管理账户来管理和部署将影响成员账户的安全防护机制。一些 AWS 服务支持使用委派管理员账户。在可能的情况下，您应使用此委派账户，而不是使用管理账户。您应严格限制对组织管理员账户的访问。

通过使用多账户策略，您可以更灵活地将防护机制应用于工作负载。AWS Security Reference Architecture 提供了有关如何设计账户结构的规范性指南。AWS Control Tower 等 AWS 服务提供了一些功能，可集中管理整个组织内的预防性和检测性控制机制。为组织中的每个账户或 OU 定义明确的用途，并根据该用途限制控制机制。

资源

相关文档：

- AWS Organizations
- 服务控制策略（SCP，Service Control Policy）
- 在多账户环境中充分利用服务控制策略
- AWS Security Reference Architecture（AWS SRA）

相关视频：

- 使用服务控制策略实施预防性防护机制
- 使用 AWS Control Tower 实施大规模管理
- AWS Identity and Access Management 深入探讨

SEC03-BP06 基于生命周期管理访问权限

将访问控制措施与操作员和应用程序生命周期以及您的集中联合身份提供者集成。例如，在用户离开组织或角色发生变化时删除用户的访问权限。

当您使用不同的账户管理工作负载时，您有时需要在这些账户之间共享资源。我们建议您使用 AWS Resource Access Manager (AWS RAM) 来共享资源。使用此服务，您可以轻松、安全地在您的 AWS Organizations 和组织部门内共享 AWS 资源。使用 AWS RAM，当账户移进和移出与之共享资源的组织或组织部门时，会自动授予或撤销对共享资源的访问权限。这样有助于您确保只与您的目标账户共享资源。

未建立此最佳实践暴露的风险等级： 低

实施指导

用户访问生命周期：针对加入的人员、工作职能变更和离开的人员实施用户访问生命周期策略，以确保只有在职用户具有访问权限。

资源

相关文档：

- 基于属性的访问控制 (ABAC)
- 授予最小特权
- IAM Access Analyzer
- 删除不必要的凭证
- 策略的使用

相关视频：

- 在最多 60 分钟的时间内成为 IAM 策略高手
- 职责分离、最低权限、委托和 CI/CD

SEC03-BP07 分析公共和跨账户访问

对于标识出存在公共访问和跨账户访问情况的调查结果，应持续监控。减少公共访问和跨账户访问，使访问仅能触达特定资源。

期望结果：了解您的 AWS 资源中哪些是共享的，以及与谁共享。持续监控和审计您的共享资源，以验证它们仅与授权的主体共享。

常见反模式：

- 不保留共享资源的清单。
- 跨账户访问或公开访问资源时，没有遵循流程。

在未建立这种最佳实践的情况下暴露的风险等级：低

实施指导

如果您的账户在 AWS Organizations 中，您可以向整个组织、特定组织单位或个人账户授予资源访问权限。如果您的账户不是某个组织的成员，您可以与个人账户共享资源。您可以使用基于资源的策略（例如 Amazon Simple Storage Service（Amazon S3）存储桶策略）授予直接跨账户访问权限，也可以允许另一账户中的主体代入您账户中的 IAM 角色来授予该权限。使用资源策略时，请验证访问权限是否仅授予给经过授权的主体。建立一个流程来审批所有需要可公开访问的资源。

AWS Identity and Access Management Access Analyzer 使用可证明的安全性来标识从账户的外部访问某个资源时的所有访问路径。它持续审核资源策略，并报告公开访问和跨账户访问的调查结果，以使您能够轻松分析可能非常宽泛的访问权限。不妨考虑配置 IAM Access Analyzer 与 AWS Organizations，来验证您是否能够查看所有账户。IAM Access Analyzer 也使得您能够先预览调查结果，然后再部署资源权限。这样，您便可以证实更改策略之后，只有您所希望的对象能够授权通过公共和跨账户访问方式触达您的资源。在设计多账户访问权限时，您可以使用信任策略来控制在何种情况下允许代入某个角色。例如，您可以使用 `PrincipalOrgId` 条件键来拒绝从 AWS Organizations 之外代入角色的尝试。

AWS Config 可以报告资源配置错误的情况，并且通过 AWS Config 策略检查，可以检测有何资源配置了公共访问权限。AWS Control Tower 和 AWS Security Hub 等服务简化了跨 AWS Organizations 部署检测性控制和防护机制的流程，可以识别并修复资源公开暴露的情况。例如，AWS Control Tower 具有托管防护机制，可以检测是否有任何 Amazon EBS 快照可由 AWS 账户恢复。

实施步骤

- 考虑为 AWS Organizations 启用 AWS Config：AWS Config 使得您能够将 AWS Organizations 内多个账户的调查结果聚合到一个委派的管理员账户。这将给您提供全局视角，进行跨账户部署 AWS Config 规则 ，以检测可公开访问的资源。
- 配置 AWS Identity and Access Management Access Analyzer IAM Access Analyzer 可帮助您识别组织和账户中与外部实体共享的资源，例如 Amazon S3 存储桶或 IAM 角色。

- 在 AWS Config 中使用自动修复来响应 Amazon S3 存储桶公共访问配置的变更情况：您可以自动重新启用 Amazon S3 存储桶阻止公共访问的设置。
- 实施监控和警报，以确定 Amazon S3 存储桶是否已变得能够公开访问：您必须设置监控和警报，以确定何时禁用 Amazon S3 屏蔽公共访问权限，以及 Amazon S3 存储桶是否已变得能够公开访问。此外，如果您使用 AWS Organizations，则可以创建一个服务控制策略来防止更改 Amazon S3 公共访问策略。AWS Trusted Advisor 检查是否存在具有开放访问权限的 Amazon S3 存储桶。如果向每个人授予“上传/删除”权限，那么任何人都可以向存储桶添加项目或者修改或删除存储桶中的项目，这样会产生潜在的安全问题。Trusted Advisor 可以检查存储桶明确拥有哪些权限，以及是否存在可能能够覆写这些权限的相关存储桶策略。您也可以使用 AWS Config 来监控 Amazon S3 存储桶是否具有公共访问权限。有关更多信息，请参阅如何使用 AWS Config 监控Amazon S3 存储桶允许公共访问的情况并作出响应。检查访问权限时，重要的是要考虑 Amazon S3 存储桶中包含哪些类型的数据。Amazon Macie 有助发现和保护敏感数据，比如 PII、PHI 和凭证（如私有密钥或 AWS 密钥）。

资源

相关文档：

- 使用 AWS Identity and Access Management Access Analyzer
- AWS Control Tower 控制机制库
- AWS 基础安全最佳实践标准
- AWS Config 托管规则
- AWS Trusted Advisor 检查参考
- 用 Amazon EventBridge 监控 AWS Trusted Advisor 检查结果
- 对横跨组织内部所有账户的规则进行管理 AWS Config
- AWS Config 和 AWS Organizations

相关视频：

- 保护多账户环境的最佳实践
- 深入探究 IAM Access Analyzer

SEC03-BP08 在组织内安全地共享资源

随着工作负载数量的增长，您可能需要将这些工作负载中资源的访问权限进行共享，或者跨多个账户多次预置资源。您可能需要进行构造来划分环境，例如划分成开发、测试和生产环境。但是，采取相互分离的构造并不会限制您安全共享权限。通过共享重叠的组件，您可以降低运维开销，并提供一致的体验，而不必猜测在多次创建同一资源时可能遗漏了什么。

期望结果：通过使用安全的方法在组织内共享资源，尽可能地减少意外访问，并帮助实施数据丢失防护计划。与管理单个组件相比，降低了运维开销，减少了多次手动创建同一组件时引起的错误，并提高了工作负载的可扩展性。您可以在多点故障场景中缩短问题解决时间，并在确定何时不再需要某个组件时更有信心。有关分析外部共享资源的规范性指南，请参阅SEC03-BP07 分析公共和跨账户访问。

常见反模式：

- 缺少对意外的外部共享进行持续监控和自动发出警报的流程。
- 缺乏关于应分享什么和不应分享什么的基准。
- 默认采用广泛的开放政策，而不是在需要时明确地分享。
- 手动创建在需要时重叠的基础资源。

在未建立这种最佳实践的情况下暴露的风险等级：中等

实施指导

设计您的访问控制和模式，以安全地管理共享资源的使用，并且仅与可信实体共享。监控共享资源，持续检查共享资源访问权限，并在不适当或意外共享时发出警报。查看分析公共和跨账户访问来帮助您设置监管机制，以减少外部访问，只对需要的资源进行访问，并建立一个持续监控和自动警报的流程。

许多 AWS 服务（如 AWS Security Hub、Amazon GuardDuty 和 AWS Backup）支持 AWS Organizations 内的跨账户共享。这些服务允许将数据共享到中心账户，可从中心账户访问，或从中心账户管理资源和数据。例如，AWS Security Hub 可将调查结果从个人账户转移到中心账户，在那里您可以查看所有调查结果。AWS Backup 可以对资源进行备份并在多个账户之间共享。您可以使用 AWS Resource Access Manager（AWS RAM）来分享其他共用资源，例如 VPC 子网和 Transit Gateway 附件、AWS Network Firewall 或 Amazon SageMaker 管道。

要限制您的账户仅在组织内共享资源，请使用服务控制策略（SCP）阻止访问外部主体。共享资源时，请将基于身份的控制措施和网络控制措施相结合，为您的组织创建数据边界，以帮助防止意外访问。数据边界是一组预防性防护机制，用于帮助验证是否只有您的可信身份才能访问预期网络中的可信资源。这些控制措施会施加适当的限制，确定哪些资源可以共享，并防止共享或暴露不应被外泄的资

源。例如，作为数据边界的一部分，您可以使用 VPC 端点策略和 `AWS:PrincipalOrgId` 条件来确保访问 Amazon S3 存储桶的身份属于您的组织。务必要注意，SCP 并不适用于服务关联角色（LSR）或 AWS 服务主体。

使用 Amazon S3 时，请对您的 Amazon S3 存储桶禁用 ACL，并使用 IAM 策略来定义访问控制。要限制从 Amazon CloudFront 访问 Amazon S3 源，请从来源访问身份（OAI）转为采用来源访问控制（OAC），后者支持其他功能，包括使用 AWS Key Management Service 进行服务器端加密。

在某些情况下，您可能希望允许在组织外部共享资源，或授予第三方访问您资源的权限。有关管理外部共享资源的权限的规范性指南，请参阅权限管理。

实施步骤

1. 使用 AWS Organizations。

   AWS Organizations 是一项账户管理服务，可让您将多个 AWS 账户 整合到您创建并集中管理的组织中。您可以将账户分组为组织单位（OU），并将不同的策略附加到每个 OU，以帮助您满足预算、安全性和合规性需求。您还可以控制 AWS 人工智能（AI）和机器学习（ML）服务收集和存储数据的方式，并使用与 Organizations 集成的 AWS 服务的多账户管理。
2. 将 AWS Organizations 与 AWS 服务集成。

   当您启用 AWS 服务以在组织的成员账户中代表您执行任务时，AWS Organizations 会在每个成员账户中为该服务创建 IAM 服务关联角色。您应使用 AWS Management Console、AWS API 或 AWS CLI 管理可信访问。有关启用可信访问的规范性指南，请参阅将 AWS Organizations 与其他 AWS 服务结合使用和可与 Organizations 一起使用的 AWS 服务。
3. 建立数据边界。

   AWS 边界通常表示为由 AWS Organizations 管理的组织。与本地网络和系统一样，访问 AWS 资源被许多人视为 My AWS 的边界。建立边界的目标，是验证如果身份可信、资源可信并且网络符合预期，则允许访问。

   a. 定义和实施边界。

      对于每个授权条件，请遵循《在 AWS 上构建边界》白皮书的边界实施中描述的步骤。有关保护网络层的规范性指南，请参阅保护网络。

   b. 持续监控并发出警报。

      AWS Identity and Access Management Access Analyzer 可帮助识别您组织和账户中与外部实体共享的资源。您可以将 IAM Access Analyzer 与 AWS Security Hub 集成，以将资源的调查结果从 IAM Access Analyzer 发送并聚合到 Security Hub，从而帮助分析环境的安全状况。要实现集

成，请在每个账户的每个“区域”中同时启用 IAM Access Analyzer 和 Security Hub。您也可以使用 AWS Config 规则 来审计配置，并使用 AWS Chatbot 和 AWS Security Hub 向相关方发出警报。然后，您可以使用 AWS Systems Manager Automation 文档来修复不合规的资源。

c. 有关对外部共享资源进行持续监控并发出警报的规范性指南，请参阅分析公共和跨账户访问。

4. 在 AWS 服务中使用资源共享并进行相应限制。

   许多 AWS 服务允许您与另一账户共享资源，或以另一账户中的资源为目标，比如亚马逊云机器镜像（AMI）和 AWS Resource Access Manager（AWS RAM）。限制 `ModifyImageAttribute` API 以指定可信账户，从而与之共享 AMI。当需要使用 AWS RAM 来将共享限制于您的组织内部时，请指定 `ram:RequestedAllowsExternalPrincipals` 条件，以帮助防止来自不可信身份的访问。有关规范性指南和注意事项，请参阅资源共享和外部目标。

5. 使用 AWS RAM 在账户中或与其他 AWS 账户 安全共享。

   AWS RAM 可帮助您与账户中的角色和用户以及与其他 AWS 账户安全地共享已创建的资源。在多账户环境中，AWS RAM 使您能够一次性创建资源并与其他账户共享。这种方法有助于降低运维开销，同时通过与 Amazon CloudWatch 和 AWS CloudTrail 的集成提供一致性、可见性和可审计性，使用跨账户访问时无法获得这些好处。

   如果您拥有以前使用基于资源的策略共享的资源，则可以使用 `PromoteResourceShareCreatedFromPolicy` API 或等效 API 将资源共享升级为完全 AWS RAM 资源共享。

   在某些情况下，您可能需要采取其他步骤来共享资源。例如，要共享加密快照，您需要共享 AWS KMS 密钥。

资源

相关最佳实践：

- SEC03-BP07 分析公共和跨账户访问
- SEC03-BP09 与第三方安全地共享资源
- SEC05-BP01 创建网络层

相关文档：

- 存储桶拥有者向并非其拥有的对象授予跨账户权限
- 如何将信任策略与 IAM 结合使用

- 在 AWS 上构建数据边界
- 如何在向第三方授予对 AWS 资源的访问权限时使用外部 ID
- 可与 AWS Organizations 一起使用的 AWS 服务
- 在 AWS 上建立数据边界：仅允许可信身份获取公司数据

相关视频：

- 使用 AWS Resource Access Manager 实现精细访问
- 使用 VPC 端点保护您的数据边界
- 在 AWS 上建立数据边界

相关工具：

- 数据边界策略示例

SEC03-BP09 与第三方安全地共享资源

确保云环境安全，不能仅仅局限于保护您的组织。您的组织有一部分数据可能要依赖第三方来管理。管理第三方托管系统的权限，应遵循及时访问的做法，使用最低权限原则和临时凭证。通过与第三方密切合作，您既可以缩小影响范围，又可以降低意外访问的风险。

期望结果：只要凭证有效且处于激活状态，任何人都可以使用与用户关联的长期 AWS Identity and Access Management（IAM）凭证、IAM 访问密钥和私有密钥。使用 IAM 角色和临时凭证可以减少维护长期凭证的工作量，包括这些敏感细节的管理和运维开销，从而帮助您改善总体安全状况。通过在 IAM 信任策略中对外部 ID 使用全局唯一标识符（UUID），并将附加到 IAM 角色的 IAM 策略置于您的控制之下，您可以审计授予第三方的访问权限，并验证该权限不会过于宽松。有关分析外部共享资源的规范性指南，请参阅SEC03-BP07 分析公共和跨账户访问。

常见反模式：

- 采用默认的 IAM 信任策略，不附加任何条件。
- 使用长期 IAM 凭证和访问密钥。
- 重用外部 ID。

在未建立这种最佳实践的情况下暴露的风险等级：中等

实施指导

您可能会希望允许在 AWS Organizations 外部共享资源，或授予第三方访问您账户的权限。例如，第三方提供的监控解决方案可能会需要访问您账户内部的资源。在这些情况下，请创建 IAM 跨账户角色，并仅向该角色提供第三方所需的权限。此外，使用外部 ID 条件定义信任策略。使用外部 ID 时，您或第三方可以为每个客户、第三方或租赁生成唯一 ID。唯一 ID 创建后，不应由除您之外的任何人控制。第三方必须实施具体流程，以一种安全、可审计且可复制的方式将外部 ID 与客户关联起来。

您也可以使用 IAM Roles Anywhere 来管理 AWS 之外使用 AWS API 的应用程序的 IAM 角色。

如果第三方不再需要访问您的环境，则删除该角色。应避免向第三方提供长期凭证。保持对其他支持共享的 AWS 服务的关注。例如，AWS Well-Architected Tool 允许与其他 AWS 账户共享工作负载，AWS Resource Access Manager 可帮助您与其他账户安全共享您拥有的 AWS 资源。

实施步骤

1. 使用跨账户角色提供对外部账户的访问。

   跨账户角色可减少外部账户和第三方为服务客户而存储的敏感信息量。跨账户角色允许您将账户中 AWS 资源的访问权限安全地授予第三方（如 AWS Partner 或组织内的其他账户），同时保持管理和审计该访问权限的能力。

   第三方可能从混合基础设施向您提供服务，或者将数据提取到一个异地位置。IAM Roles Anywhere 可帮助您使第三方工作负载能够安全地与 AWS 工作负载交互，并进一步减少对长期凭证的需求。

   不应使用长期凭证或与用户关联的访问密钥来提供外部账户访问，而应使用跨账户角色来提供跨账户访问。
2. 对第三方使用外部 ID。

   使用外部 ID，您就可以指定谁可以在 IAM 信任策略中代入角色。信任策略可能要求代入角色的用户声明他们所处的条件和追求的目标。它还为账户拥有者提供了一种方法，允许仅在特定情况下代入角色。外部 ID 的主要功能是解决和防止混淆代理问题。

   如果您是 AWS 账户 拥有者，并且您为第三方配置了一个角色（该角色可以访问您的和其他 AWS 账户），或者当您可以代表不同的客户代入角色时，请使用外部 ID。与第三方或 AWS Partner 合作，建立一个包括在 IAM 信任策略中的外部 ID 条件。
3. 使用全局唯一外部 ID。

   实施一个为外部 ID（例如全局唯一标识符（UUID））生成随机唯一值的流程。第三方在不同客户之间重用外部 ID 并不能解决混淆代理问题，因为客户 A 可以通过使用客户 B 的角色 ARN 以及重复的

外部 ID 来查看客户 B 的数据。在多租户环境中，第三方支持多个具有不同 AWS 账户 的客户，此时第三方必须使用不同的唯一 ID 作为每个 AWS 账户 的外部 ID。第三方负责检测重复的外部 ID，并将每个客户安全地映射到各自的外部 ID。第三方应进行测试，以验证他们只能在指定外部 ID 时代入该角色。在需要外部 ID 之前，第三方应避免存储客户角色 ARN 和外部 ID。

外部 ID 不视为密钥，但外部 ID 不能是容易猜测的值，例如电话号码、姓名或账户 ID。将外部 ID 设置为只读字段，这样就无法为了冒充设置而更改外部 ID。

您或第三方可以生成外部 ID。定义一个流程，确定谁负责生成 ID。无论创建外部 ID 的实体是什么，第三方都必须确保客户之间的唯一性和格式一致。

4. 弃用客户提供的长期凭证。

   弃用长期凭证，使用跨账户角色或 IAM Roles Anywhere。如果必须使用长期凭证，请制定相应计划，逐渐转变成基于角色进行访问。有关管理密钥的详细信息，请参阅身份管理。同时与 AWS 账户 团队和第三方合作，建立风险缓解运行手册。有关应对和减轻安全事件潜在影响的规范性指南，请参阅事件响应。

5. 验证设置是否具有规范性指导，或是否实现了自动化。

   为您账户中的跨账户访问创建的策略必须遵循最低权限原则。第三方必须为您提供使用 AWS CloudFormation 模板或等效模板的角色策略文档或自动化设置机制。这减少了手动创建策略时出错的机会，并提供了可审计的跟踪。有关使用 AWS CloudFormation 模板创建跨账户角色的更多信息，请参阅跨账户角色。

   第三方应提供一个自动化的、可审计的设置机制。但是，通过使用角色策略文档（此文档大致列出了所需的访问权限），角色设置的自动化应该由您来完成。使用 AWS CloudFormation 模板或等效模板，您应将偏差检测纳入审计流程以监控变更。

6. 对变更做出解释。

   您的账户结构、您对第三方的需求或他们提供的服务可能会发生变更。您应预料到可能会发生变动和失败，并进行相应的规划：请安排合适的人员，建立适当的流程并采用正确的技术进行应对。应定期审计您提供的访问级别，并实施检测方法，以便在发生意外变更时向您发出警报。监控并审计角色的使用情况，以及外部 ID 的数据存储状态。若发生意外变更或存在不当访问模式，您应准备暂时或永久撤销第三方访问权限。此外，还要衡量撤销操作造成的影响，包括执行该操作所需的时间、涉及的人员、成本以及对其他资源的影响。

   有关检测方法的规范性指南，请参阅检测最佳实践。

资源

相关最佳实践：

- SEC02-BP02 使用临时凭证
- SEC03-BP05 为您的组织定义权限防护机制
- SEC03-BP06 基于生命周期管理访问权限
- SEC03-BP07 分析公共和跨账户访问
- SEC04 检测

相关文档：

- 存储桶拥有者向并非其拥有的对象授予跨账户权限
- 如何将信任策略与 IAM 结合使用
- 使用 IAM 角色委派跨 AWS 账户 的访问权限
- 如何使用 IAM 访问其他 AWS 账户 中的资源？
- IAM 中的安全最佳实践
- 跨账户策略评估逻辑
- 如何在向第三方授予对 AWS 资源的访问权限时使用外部 ID
- 使用自定义资源从外部账户中创建的 AWS CloudFormation 资源中收集信息
- 安全地使用外部 ID 访问他人拥有的 AWS 账户
- 使用 IAM Roles Anywhere 将 IAM 角色扩展到 IAM 外部工作负载

相关视频：

- 如何允许单独 AWS 账户 中的用户或角色访问我的 AWS 账户？
- AWS re:Invent 2018：在 60 分钟以内成为 IAM 策略高手
- AWS 知识中心实况：IAM 最佳实践和设计决策

相关示例：

- Well-Architected 实验室 - Lambda 跨账户 IAM 角色代入（第 300 级）
- 配置对 Amazon DynamoDB 的跨账户访问
- AWS STS 网络查询工具

检测

检测分为两个部分：一个部分是检测意外或不需要的配置更改，另一个部分是检测意外行为。在应用程序交付生命周期中，可以在多个位置执行第一个部分。利用基础设施即代码（例如，CloudFormation 模板），您可以通过在 CI/CD 管道或源代码控制中实施检查，在部署工作负载前检查不需要的配置。之后，当您将工作负载部署到非生产环境和生产环境中时，您可以使用原生 AWS、开源或 AWS 合作伙伴工具检查配置。通过执行这些检查，可以查明不符合安全原则或最佳实践的配置，或者已测试配置与已部署配置之间的变动。对于正在运行的应用程序，您可以检查是否意外地更改了配置，包括在已知部署或自动扩展事件的外部进行更改。

对于检测的第二个部分，您可以使用工具或通过在特定类型的 API 调用增加时发出提醒来检测意外行为。利用 Amazon GuardDuty，您可以在 AWS 账户中发生意外活动和可能未经授权或恶意的活动时收到提醒。您还应明确监控不希望在工作负载中使用的变异 API 调用，以及改变安全状况的 API 调用。

使用检测功能，您可以识别潜在安全配置错误、威胁或意外行为。检测是安全生命周期的重要组成部分，可用于支持质量流程、法律或合规义务，还可以用于威胁识别和响应工作。检测机制分为多种不同的类型。例如，可以分析来自您工作负载的日志，以找到正在被利用的漏洞。您应定期查看与您的工作负载相关的检测机制，以确保符合内部和外部的策略和要求。自动化警报和通知应基于所定义的条件，以使您的团队或工具能够执行调查。这些机制都是重要的响应手段，可以帮助您的组织识别和了解异常活动的范围。

在 AWS 中，可以使用很多方法来解决检测性机制问题。以下部分介绍了如何使用这些方法：

最佳实践

- SEC04-BP01 配置服务和应用程序日志记录
- SEC04-BP02 集中分析日志、结果和指标
- SEC04-BP03 自动响应事件
- SEC04-BP04 实施可操作的安全事件

SEC04-BP01 配置服务和应用程序日志记录

保留服务和应用程序的安全事件日志。这是审计、调查和运营使用案例的基本安全原则，也是由监管、风险与合规性（GRC）标准、政策和程序驱动的共同安全要求。

期望结果：当需要履行内部流程或义务（如安全事件响应）时，组织应能够及时、可靠且一致地从 AWS 服务和应用程序中检索安全事件日志。考虑将日志集中起来，以取得更好的运营成果。

常见反模式：

- 日志被永久存储或过早删除。
- 每个人都可以访问日志。
- 完全依赖手动流程进行日志治理和使用。
- 存储每一种类型的日志，以备不时之需。
- 仅在必要时检查日志完整性。

建立此最佳实践的好处：为安全事件实施根本原因分析（RCA）机制，并为您的监管、风险与合规性义务提供证据来源。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

在安全调查或基于要求的其他使用案例期间，您需要能够查看相关日志，以记录并了解事件的来龙去脉和时间线。警报生成也需要日志，以指示发生了某些感兴趣的操作。选择、启用、存储和设置查询、检索机制以及警报至关重要。

实施步骤

- 选择并启用日志源。 进行安全调查之前，您需要捕获相关日志，以便以回溯方式重建 AWS 账户 中的活动。选择并启用与工作负载相关的日志源。

  日志源的选择标准应基于业务所需的使用案例。使用 AWS CloudTrail 或 AWS Organizations 跟踪为每个 AWS 账户 建立跟踪，并为其配置 Amazon S3 存储桶。

  AWS CloudTrail 是一项日志记录服务，可跟踪针对 AWS 账户 捕获 AWS 服务活动所进行的 API 调用。它默认情况下启用，管理事件保留 90 天，可以使用 AWS Management Console、AWS CLI 或 AWS SDK 通过 CloudTrail 事件历史记录检索这些事件。为了更长久地保留和了解数据事件，请创建 CloudTrail 跟踪并将其与 Amazon S3 存储桶关联，也可以选择与 Amazon CloudWatch 日志组关联。或者，您可以创建 CloudTrail Lake，这可保留 CloudTrail 日志长达七年之久，并提供基于 SQL 的查询工具

  AWS 建议使用 VPC 的客户分别使用 VPC 流日志和 Amazon Route 53 解析器查询日志启用网络流量和 DNS 日志，并将其流式传输到 Amazon S3 存储桶或 CloudWatch 日志组。您可以为 VPC、子网或网络接口创建 VPC 流日志。对于 VPC 流日志，您可以选择使用流日志的方式和位置，以降低成本。

AWS CloudTrail 日志、VPC 流日志和 Route 53 解析器查询日志是支持 AWS 中安全调查的基本日志记录源。您还可以使用亚马逊安全数据湖以 Apache Parquet 格式和开放网络安全架构框架（OCSF）收集、标准化和存储这些日志数据，以便于查询。安全数据湖还支持其他 AWS 日志和来自第三方的日志。

AWS 服务可以生成基本日志源未捕获到的日志，如 Elastic Load Balancing 日志、AWS WAF 日志、AWS Config 记录器日志、Amazon GuardDuty 调查结果、Amazon Elastic Kubernetes Service（Amazon EKS）审计日志，以及 Amazon EC2 实例操作系统和应用程序日志。有关日志记录和监控选项的完整列表，请参阅 AWS 安全事件响应指南的附录 A：云功能定义 – 日志记录和事件。

- 研究每项 AWS 服务和应用程序的日志记录功能：每项 AWS 服务和应用程序都为您提供了日志存储选项，每个选项都有自己的保留和生命周期功能。两种很常见的日志存储服务是 Amazon Simple Storage Service（Amazon S3）和 Amazon CloudWatch。如果保留期较长，建议使用 Amazon S3，因为它具有成本效益和灵活的生命周期功能。如果主要日志记录选项是 Amazon CloudWatch Logs，作为一种选择，您应该考虑将不太经常访问的日志存档到 Amazon S3。
- 选择日志存储：日志存储的选择通常与您使用的查询工具、保留能力、熟悉程度和成本有关。日志存储的主要选项是 Amazon S3 存储桶或 CloudWatch 日志组。

  Amazon S3 存储桶提供持久且经济高效的存储，并具有可选的生命周期策略。可以使用 Amazon Athena 等服务查询存储在 Amazon S3 存储桶中的日志。

  CloudWatch 日志组通过 CloudWatch Logs Insights 提供持久存储和内置查询工具。
- 确定适当的日志保留时长：使用 Amazon S3 存储桶或 CloudWatch 日志组存储日志时，必须为每个日志源建立足够的生命周期，以优化存储和检索成本。客户通常可以查询三个月到一年的日志，日志保留期长达七年。可用性和保留时长的选择应与您的安全要求以及法律法规和业务授权的综合因素相一致。
- 使用适当的保留时长和生命周期策略为每个 AWS 服务和应用程序启用日志记录：对于组织内的每个 AWS 服务或应用程序，请查找特定的日志记录配置指南：
  - 配置 AWS CloudTrail 跟踪
  - 配置 VPC 流日志
  - 配置 Amazon GuardDuty 调查结果导出
  - 配置 AWS Config 记录
  - 配置 AWS WAF Web ACL 流量
  - 配置 AWS Network Firewall 网络流量日志
  - 配置 Elastic Load Balancing 访问日志

  - 配置 Amazon Route 53 解析器查询日志
  - 配置 Amazon RDS 日志
  - 配置 Amazon EKS 控制面板日志
  - 为 Amazon EC2 实例和本地服务器配置 Amazon CloudWatch 代理
- 选择和实施日志查询机制：对于日志查询，可以使用 CloudWatch Logs Insights 对存储在 CloudWatch 日志组中的数据进行查询，使用 Amazon Athena 和 Amazon OpenSearch Service 对存储在 Amazon S3 中的数据进行查询。您还可以使用第三方查询工具，如安全信息和事件管理（SIEM）服务。

  选择日志查询工具的过程中，应考虑安全运营的人员、流程和技术方面。选择一款能够满足运营、业务和安全要求并可长期使用和维护的工具。请记住，当要扫描的日志数量保持在工具的限制范围内时，日志查询工具的工作状态最佳。由于成本或技术限制，拥有多款查询工具的情况并不罕见。

  例如，您可能使用第三方安全信息和事件管理（SIEM）工具对过去 90 天的数据执行查询，但由于 SIEM 的日志提取成本较高，使用 Athena 来执行 90 天以上的查询。无论采用何种实施方式，都要验证您的方法能够尽可能地减少充分提高运营效率所需的工具数量，尤其在安全事件调查期间。
- 使用日志发出警报：AWS 通过多项安全服务提供警报功能：
  - AWS Config 监控和记录您的 AWS 资源配置，并允许您对照所需的配置自动执行评估和修复。
  - Amazon GuardDuty 是一项威胁检测服务，可持续监控恶意活动和未经授权的行为，以保护您的 AWS 账户 和工作负载。GuardDuty 可从 AWS CloudTrail 管理和数据事件、DNS 日志、VPC 流日志和 Amazon EKS 审计日志等来源提取、聚合和分析信息。GuardDuty 可直接从 CloudTrail、VPC 流日志、DNS 查询日志和 Amazon EKS 提取独立的数据流。您无需管理 Amazon S3 存储桶策略，也无需修改日志的收集和存储方式。仍建议保留这些日志，以便您自己进行调查和遵守法规。
  - AWS Security Hub 集中聚合、组织和优先处理来自多个 AWS 服务和可选第三方产品的安全警报或调查结果，以使您全面了解安全警报和合规性状态。

  您也可以使用自定义警报生成引擎来处理这些服务未涵盖的安全警报或与您的环境相关的特定警报。有关构建这些警报和检测的信息，请参阅 AWS 安全事件响应指南中的“检测”。

资源

相关最佳实践：

- SEC04-BP02 集中分析日志、结果和指标
- SEC07-BP04 定义数据生命周期管理

- SEC10-BP06 预部署工具

相关文档：

- AWS 安全事件响应指南
- 亚马逊安全数据湖入门
- 入门：Amazon CloudWatch Logs
- 安全合作伙伴解决方案：日志记录和监控

相关视频：

- AWS re:Invent 2022 - 介绍亚马逊安全数据湖

相关示例：

- 专为 AWS 提供的 Assisted Log Enabler
- AWS Security Hub 调查结果历史导出

相关工具：

- Snowflake 增强网络安全

SEC04-BP02 集中分析日志、结果和指标

安全运营团队依靠收集日志和使用搜索工具来发现需要关注的潜在事件，这些事件可能代表未经授权的活动或无意的更改。但是，仅仅分析收集的数据和手动处理信息不足以应对从复杂架构流出的大量信息。单凭分析和报告无法及时分配合适的资源来处理事件。

建立成熟的安全运维团队的最佳实践是，将安全事件和调查结果的流程深度集成到通知和工作流系统中，例如票证系统、错误或问题系统或者其他安全信息和事件管理（SIEM，Security Information and Event Management）系统。这样，工作流可以摆脱电子邮件和静态报告，让您能够路由、上报和管理事件或调查结果。许多组织也在逐步将安全警报集成到他们的聊天或协作以及开发人员工作效率平台中。对于正在踏上自动化之旅的组织，在规划首要自动化任务时，一个由 API 驱动的低延迟票证系统能够提供极高的灵活性。

这种最佳实践不仅适用于从描述用户活动或网络事件的日志消息生成的安全事件，还适用于在基础设施本身检测到的更改生成的安全事件。当面对一些更改，而且这些更改的不受欢迎程度足够微妙，以致于目前无法使用 AWS Identity and Access Management（IAM）和 AWS Organizations 配置的组合来防止这些更改发生时，为了保持和验证安全架构，必须能够检测更改、确定更改是否适当，然后将这些信息路由到正确的修复工作流程。

Amazon GuardDuty 和 AWS Security Hub 为日志记录提供了聚合、重复数据删除和分析机制，您也可以通过其他 AWS 服务提供这些机制。GuardDuty 可从 AWS CloudTrail 管理和数据事件、VPC DNS 日志以及 VPC 流日志等来源提取、聚合和分析信息。Security Hub 能够提取、聚合和分析来自 GuardDuty、AWS Config、Amazon Inspector、Amazon Macie、AWS Firewall Manager 以及 AWS Marketplace 中提供的大量第三方安全产品的输出，如果您相应构建了自己的代码，还将包括这些代码。GuardDuty 和 Security Hub 都有一个管理员-成员模型，此模型可以跨多个账户聚合调查结果和见解，拥有本地 SIEM 的客户通常将 Security Hub 用作 AWS 端日志和警报预处理器和聚合器，随后即可通过基于 AWS Lambda 的处理器和转发服务器提取 Amazon EventBridge。

未建立这种最佳实践的情况下暴露的风险等级： 高

实施指导

- 评估日志处理功能：评估用于处理日志的选项
  - 使用 Amazon OpenSearch Service 来记录和监控（几乎）所有内容
  - 寻找专门提供日志记录和监控解决方案的合作伙伴
- 作为分析 CloudTrail 日志的开始，请测试 Amazon Athena。
  - 配置 Athena 分析 CloudTrail 日志
- 在 AWS 中实施集中式日志记录：请参阅以下 AWS 示例解决方案来集中处理多个来源的日志记录。
  - 集中日志记录解决方案
- 通过合作伙伴集中处理日志记录：APN 合作伙伴拥有可以帮助您集中分析日志的解决方案。
  - 日志记录和监控

资源

相关文档：

- AWS Answers：集中式日志记录
- AWS Security Hub
- Amazon CloudWatch

- Amazon EventBridge
- 开始使用：Amazon CloudWatch Logs
- 安全合作伙伴解决方案：日志记录和监控

相关视频：

- 集中监控资源配置和合规性
- 修正 Amazon GuardDuty 和 AWS Security Hub 调查结果
- 云中的威胁管理：Amazon GuardDuty 和 AWS Security Hub

SEC04-BP03 自动响应事件

使用自动化流程调查和修复事件可减少人工处理工作量和人为错误，从而扩展调查功能。定期审核将帮助您优化自动化工具，并实现持续迭代。

在 AWS 中，可以使用 Amazon EventBridge，调查感兴趣的事件以及自动化工作流程可能发生的意外变化的相关信息。此服务提供可扩展的规则引擎，可代理原生 AWS 事件格式（例如 AWS CloudTrail 事件）以及您可以从应用程序中生成的自定义事件。Amazon GuardDuty 还允许您将这些事件路由到构建意外事件响应系统（AWS Step Functions）的工作流程系统中，或者路由到中央安全账户或存储桶中以执行进一步分析。

检测更改并将此信息路由到正确的工作流的操作也可以使用 AWS Config 规则 和 合规包完成。AWS Config 会检测对范围内服务的更改（虽然延迟会比 EventBridge 更高），并生成可使用 AWS Config 规则 进行解析的事件，以便进行回滚、强制实施合规性策略以及将信息转发到相关系统（如变更管理平台和运营票证系统）。除了编写您自己的 Lambda 函数以响应 AWS Config 事件，您还可以充分利用 AWS Config 规则 开发工具包以及 一组开源 AWS Config 规则。合规包是 AWS Config 规则 和修复操作的集合，您可将其作为以 YAML 模板格式创作的单个实体进行部署。一个 示例合规包模板， 面向 Well-Architected 安全性支柱提供。

未建立这种最佳实践的情况下暴露的风险等级： 中

实施指导

- 使用 GuardDuty 实施自动化警报：GuardDuty 是一种威胁检测服务，可持续监控恶意活动和未经授权的行为，从而保护您的 AWS 账户和工作负载。启用 GuardDuty 并配置自动化警报。
- 自动执行调查流程：制定自动化流程来调查事件并向管理员报告信息，以便节省时间。
  - 实验室：Amazon GuardDuty 动手实践

资源

相关文档：

- AWS Answers：集中式日志记录
- AWS Security Hub
- Amazon CloudWatch
- Amazon EventBridge
- 开始使用：Amazon CloudWatch Logs
- 安全合作伙伴解决方案：日志记录和监控
- 设置 Amazon GuardDuty

相关视频：

- 集中监控资源配置和合规性
- 修正 Amazon GuardDuty 和 AWS Security Hub 调查结果
- 云中的威胁管理：Amazon GuardDuty 和 AWS Security Hub

相关示例：

- 实验室：自动部署检测性控制

SEC04-BP04 实施可操作的安全事件

创建发送给团队并将由团队处理的警报。确保警报包含团队采取措施所需的相关信息。对于您的每个检测性机制，您还应调查一个以 运行手册 或者 行动手册形式存在的流程。例如，当您启用 Amazon GuardDuty时，它会生成不同的 调查结果。您的每个调查结果类型都应具有一个运行手册条目，例如，如果发现了 特洛伊木马程序， 您的运行手册的简单说明可以指示某个人员进行调查和修复。

未建立这种最佳实践的情况下暴露的风险等级： 低

实施指导

- 发现可用于 AWS 服务的指标：发现可通过 Amazon CloudWatch 用于您正在使用的服务的指标。
  - AWS 服务文档

  - 使用 Amazon CloudWatch 指标
- 配置 Amazon CloudWatch 告警。
  - 使用 Amazon CloudWatch 告警

资源

相关文档：

- Amazon CloudWatch
- Amazon EventBridge
- 安全合作伙伴解决方案：日志记录和监控

相关视频：

- 集中监控资源配置和合规性
- 修正 Amazon GuardDuty 和 AWS Security Hub 调查结果
- 云中的威胁管理：Amazon GuardDuty 和 AWS Security Hub

基础设施保护

基础设施保护包括满足最佳实践和组织、法律及监管义务所必需的控制方法（例如深度防御）。使用这些方法对于在云中持续成功运营至关重要。

基础设施保护是信息安全计划的关键组成部分之一。它可以确保您的工作负载中的系统和服务受到保护，防止意外和未经授权的访问以及潜在的漏洞对其造成危害。例如，您可以定义信任边界（例如网络边界和账户边界）、系统安全配置和维护（例如强化、最小化和修补）、操作系统身份验证和授权（例如用户、密钥和访问级别）以及其他适当的策略执行点（例如 Web 应用程序防火墙和/或 API 网关）。

区域、可用区、AWS Local Zones 和 AWS Outposts

确保您熟悉区域、可用区、AWS Local Zones和 AWS Outposts，它们是 AWS 安全全球基础设施的组成部分。

在 AWS 中，区域的概念是指世界范围内的一个用来聚集数据中心的物理位置。我们将每个逻辑数据中心组称作可用区（AZ，Availability Zone）。每个 AWS 区域由一个地理区域内的多个隔离且物理上独立的 AZ 组成。如果您有数据驻留要求，则可以选择靠近所需位置的 AWS 区域。在数据所在的物理区域中，您保留对数据的完全控制权和所有权，这有助于满足您的区域合规性和数据驻留要求。每个 AZ 都有独立的电力、冷却和物理安全性。如果跨 AZ 对应用程序进行分区，则可以更好地隔离并保护您免受断电、雷击、龙卷风、地震等问题的影响。各个 AZ 之间在物理上具有相当的距离（许多公里），不过这个距离都在 100 公里（60 英里）以内。一个 AWS 区域中的所有 AZ 都使用完全冗余的专用城域光纤，通过高带宽、低延迟的网络互联，从而在 AZ 之间实现高吞吐量、低延迟的联网。AZ 之间的所有流量都已加密。专注于高可用性的 AWS 客户可以将其应用程序设计为在多个 AZ 中运行，以实现更大的容错能力。AWS 区域可满足最高级别的安全性、合规性和数据保护要求。

AWS Local Zones 将计算、存储、数据库和其他精选 AWS 服务放置在更靠近最终用户的位置。利用 AWS Local Zones，您可以轻松运行要求为最终用户提供数毫秒延迟的高要求应用程序，例如媒体和娱乐内容创建、实时游戏、储层模拟、电子设计自动化和机器学习。每个 AWS Local Zone 位置都是 AWS 区域的扩展，您可以在其中使用 AWS 服务（例如，Amazon EC2、Amazon VPC、Amazon EBS、Amazon File Storage 和 Elastic Load Balancing），在靠近最终用户的地理位置运行对延迟敏感的应用程序。AWS Local Zones 在本地工作负载与在 AWS 区域中运行的工作负载之间提供高带宽的安全连接，使您能够通过相同的 API 和工具集无缝连接到所有区域内服务。

AWS Outposts 可将 AWS 原生服务、基础设施和运营模式引入到几乎任何数据中心、主机托管空间或本地设施。您可以跨本地设施和 AWS Cloud，使用相同的 AWS API、工具和基础设施来交付真正一致

的混合体验。AWS Outposts 专为互联环境而设计，可用于支持因低延迟或本地数据处理需求而必须保留在本地的工作负载。

在 AWS 中，有许多基础设施保护方法。以下部分介绍了如何使用这些方法。

主题

保护网络

不论是您的员工还是客户，用户可以在任何地方。传统模式信任有权访问您网络的任何人和任何对象，您需要摆脱这种模式。当您遵循在所有层应用安全性的原则时，您可以使用 零信任 方案。零信任安全性是一种模式，在这个模式中，应用程序组件或微服务被视为是彼此分离的，并且任何组件或微服务均不相互信任。

妥善规划和管理您的网络设计，这是为您工作负载中的资源提供分离和边界的基础。由于您工作负载中的很多资源都运行在 VPC 中并继承安全属性，因此必须使用由自动化作为后盾的检查和保护机制来支持设计。同样，对于在 VPC 之外运行的工作负载，当使用纯粹边缘服务和/或无服务器环境时，这些最佳实践适用于更加简化的方法。有关 Web 应用程序后端方面的建议，请参阅 AWS Well-Architected 无服务器应用程序剖析 ，以获得有关无服务器安全性的特定指导。

最佳实践

SEC05-BP01 创建网络层

将具有共同敏感度要求的组件分成若干层，以尽量缩小未经授权访问的潜在影响范围。例如，应将虚拟私有云（VPC）中无需进行互联网访问的数据库集群，放在无法向/从互联网路由的子网中。流量应仅从相邻的下一个最不敏感的资源流出。应考虑设置一个位于负载均衡器后面的 Web 应用程序。不应直接从负载均衡器访问数据库。只有业务逻辑或 Web 服务器才能直接访问数据库。

期望结果：创建分层网络。分层网络有助于对类似的网络组件进行逻辑分组。它们还缩小了未经授权网络访问的潜在影响范围。适当分层的网络使未经授权的用户更难转向 AWS 环境中的其他资源。除了保护内部网络路径之外，还应保护网络边缘，如 Web 应用程序和 API 端点。

常见反模式：

- 在单个 VPC 或子网中创建所有资源。
- 使用过于宽松的安全组。
- 未能使用子网。
- 允许直接访问数据库等数据存储。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

具有相同可访问性要求的组件（例如 Amazon Elastic Compute Cloud（Amazon EC2）实例、Amazon Relational Database Service（Amazon RDS）数据库集群和 AWS Lambda 函数）可细分为由子网构成的层。不妨考虑在 VPC 内或 Amazon API Gateway 后部署无服务器工作负载，如 Lambda 函数。应将无需进行互联网访问的 AWS Fargate (Fargate) 任务放在无法向/从互联网路由的子网中。此分层方法可减轻单层错误配置的影响，这种错误可能导致能够发生意外访问。对于 AWS Lambda，您可以在 VPC 中运行您的函数，以充分利用基于 VPC 的控制。

对于可能包括数千个 VPC、AWS 账户 和本地网络的网络连接，您应使用 AWS Transit Gateway。Transit Gateway 充当一个枢纽，以控制如何在类似于辐条的所有互联网络之间路由流量。Amazon Virtual Private Cloud（Amazon VPC）和 Transit Gateway 之间的流量仍在 AWS 专用网络上，这减少了对未经授权用户的外部暴露和潜在的安全问题。Transit Gateway 区域间对等也会对区域间流量加密，而且不会出现任何单点故障或带宽瓶颈。

实施步骤

- 根据配置，使用 Reachability Analyzer 分析源和目标之间的路径：Reachability Analyzer 使得您能够自动验证与 VPC 所连资源的连接性。请注意，此分析是通过检查配置完成的（在进行分析时不发送网络数据包）。
- 使用 Amazon VPC 网络访问分析器识别资源意外受到网络访问的情况：Amazon VPC 网络访问分析器使您能够指定网络访问要求，并识别潜在的网络访问路径。
- 考虑资源是否需要在公有子网中：不要将资源放在您的 VPC 的公有子网中，除非它们绝对必须要接收来自公共来源的入站网络流量。

- 在 VPC 中创建子网：为每个网络层创建子网（在包含多个可用区的组中），以增强微分段。还要验证您已将正确的路由表与子网关联，以控制路由和互联网连接。
- 使用 AWS Firewall Manager 管理 VPC 安全组：AWS Firewall Manager 有助于减轻使用多个安全组的管理负担。
- 使用 AWS WAF 防范常见的 Web 漏洞：AWS WAF 可通过检查流量中是否存在常见的 Web 漏洞（如 SQL 注入）来帮助增强边缘安全性。它还使您能够限制来自特定国家/地区或地理位置的 IP 地址的流量。
- 使用 Amazon CloudFront 作为内容分发网络（CDN）：Amazon CloudFront 可通过将数据存储在更靠近用户的位置来帮助加快 Web 应用程序的速度。它还可以实施 HTTPS，限制对地理区域的访问，并确保网络流量只能在通过 CloudFront 路由时访问资源，从而提高边缘安全性。
- 创建应用程序编程接口（API）时使用 Amazon API Gateway：Amazon API Gateway 可帮助发布、监控和保护 REST、HTTPS 和 WebSocket API。

资源

相关文档：

- AWS Firewall Manager
- Amazon Inspector
- Amazon VPC 安全性
- Reachability Analyzer
- Amazon VPC 网络访问分析器

相关视频：

- 用于各种 VPC 的 AWS Transit Gateway 参考架构
- 使用 Amazon CloudFront、AWS WAF 和 AWS Shield 提供应用程序加速和保护
- AWS re:Inforce 2022 - 验证 AWS 上的网络访问控制措施的有效性
- AWS re:Inforce 2022 - 使用 AWS WAF 针对机器人进行高级防护

相关示例：

- Well-Architected 实验室 - 自动部署 VPC
- 研讨会：Amazon VPC 网络访问分析器

SEC05-BP02 控制所有层的流量

当构建您的网络拓扑时，您应检查每个组件的连接要求。例如，某个组件是否需要互联网可访问性（入站和出站）、连接到 VPC 的能力、边缘服务和外部数据中心。

使用 VPC，您可以使用您设置的私有 IPv4 地址范围或者 AWS 选择的 IPv6 地址范围来定义跨 AWS 区域的网络拓扑。对于入站和出站流量，您应采用深度防御方法应用多种控制，包括使用安全组（状态检测防火墙）、网络 ACL、子网和路由表。在 VPC 中，您可以在可用区中创建子网。每个子网都可以拥有一个关联的路由表，此表定义了用于管理流量在子网内所采用路径的路由规则。您可以将要连接到互联网或 NAT 网关的路由连接到 VPC 或使其经过另一个 VPC，以定义互联网可路由子网。

当在 VPC 内启动某个实例、Amazon Relational Database Service（Amazon RDS）数据库或其他服务时，它的每个网络接口都有自己的安全组。此防火墙位于操作系统层之外，可用于定义允许入站和出站流量的规则。您还可以定义安全组之间的关系。例如，通过参考对相关的实例应用的安全组，数据库层安全组中的实例仅接受来自应用程序层内实例的流量。除非您在使用非 TCP 协议，否则不必在以下情况下允许互联网直接访问 Amazon Elastic Compute Cloud（Amazon EC2）实例（甚至使用安全组禁止使用的端口）：没有负载均衡器或 CloudFront。这样有助于防止通过操作系统或应用程序问题进行意外访问。您还可以为子网附加网络 ACL，它将用作无状态防火墙。您应配置网络 ACL 以缩小各层之间允许的流量范围，但请注意，您需要定义入站和出站规则。

一些 AWS 服务要求组件访问互联网进行 API 调用，其目标是 AWS API 端点 所在的位置。另外一些 AWS 服务使用 VPC 端点 ，这些端点位于您的 Amazon VPC 中。很多 AWS 服务（包括 Amazon S3 和 Amazon DynamoDB）都支持 VPC 端点，并且已在 AWS PrivateLink中广泛使用此技术。我们建议您使用此方法来访问 AWS 服务、第三方服务以及安全地托管在其他 VPC 中您自己的服务。AWS PrivateLink 上的所有网络流量保持在 AWS 骨干网中，永远不会通过互联网。连接只能由服务的使用方启动，不能由服务的提供方启动。为外部服务访问使用 AWS PrivateLink 让您可以创建没有互联网访问的气隙 VPC，帮助您保护 VPC 免受外部威胁因素的影响。第三方服务可以使用 AWS PrivateLink 允许其客户通过私有 IP 地址，从其 VPC 连接到服务。对于需要出站连接到互联网的 VPC 资产，可以让它们通过 AWS 托管的 NAT 网关、仅出站的互联网网关或者您创建并管理的 Web 代理进行仅出站（单向）连接。

未建立这种最佳实践的情况下暴露的风险等级： 高

实施指导

- 控制 VPC 中的网络流量：实施 VPC 最佳实践来控制流量。
  - Amazon VPC 安全性
  - VPC 端点
  - Amazon VPC 安全组

  - 网络 ACL
- 控制边缘站点的流量：实施边缘服务（例如 Amazon CloudFront），以提供一层额外的保护和其他功能。
  - Amazon CloudFront 使用案例
  - AWS Global Accelerator
  - AWS Web Application Firewall（AWS WAF）
  - Amazon Route 53
  - Amazon VPC 传入路由
- 控制私有网络流量：实施保护工作负载专有流量的服务。
  - Amazon VPC 对等连接
  - Amazon VPC 端点服务（AWS PrivateLink）
  - Amazon VPC Transit Gateway
  - AWS Direct Connect
  - AWS Site-to-Site VPN
  - AWS Client VPN
  - Amazon S3 接入点

资源

相关文档：

- AWS Firewall Manager
- Amazon Inspector
- 开始使用 AWS WAF

相关视频：

- 用于各种 VPC 的 AWS Transit Gateway 参考架构
- 使用 Amazon CloudFront、AWS WAF 和 AWS Shield 提供应用程序加速和保护

相关示例：

- 实验室：自动部署 VPC

SEC05-BP03 自动执行网络防护

自动运行保护机制，以提供基于威胁情报和异常检测的自我防御网络。例如可应对最新的威胁并减轻它们的影响的那些入侵检测和预防工具。您可以通过实施 Web 应用程序防火墙来实现自动化的网络保护，例如使用 AWS WAF Security Automations 解决方案（https://github.com/awslabs/aws-waf-security-automations）来自动拦截来自已知威胁媒介相关 IP 地址的请求。

未建立这种最佳实践的情况下暴露的风险等级： 中

实施指导

- 自动执行基于 Web 流量的保护：AWS 提供了使用 AWS CloudFormation 自动部署一组 AWS WAF 规则的解决方案，旨在筛选常见的基于 Web 的攻击。用户可以从预配置的保护功能中进行选择，这些功能定义 AWS WAF Web 访问控制列表（Web ACL）中包含的规则。
  - AWS WAF 安全自动化
- 考虑使用 AWS Partner 解决方案：AWS 合作伙伴提供数百种业界领先的产品，这些产品与您的本地环境中的现有控制措施等效、相同或与之集成。这些产品对现有 AWS 服务起到补充作用，使您能够在云和本地部署环境中部署全面的安全架构，进而实现更无缝的体验。
  - 基础设施安全性

资源

相关文档：

- AWS Firewall Manager
- Amazon Inspector
- Amazon VPC 安全性
- 开始使用 AWS WAF

相关视频：

- 用于各种 VPC 的 AWS Transit Gateway 参考架构
- 使用 Amazon CloudFront、AWS WAF 和 AWS Shield 提供应用程序加速和保护

相关示例：

- 实验室：自动部署 VPC

SEC05-BP04 实施检查和保护

检查和筛选每层的流量。您可以使用 VPC Network Access Analyzer检测 VPC 配置中可能存在的意外访问。您可以指定网络访问需求，然后确定不能满足这些要求的潜在网络路径。对于通过基于 HTTP 的协议处理的组件，Web 应用程序防火墙可帮助防止常见的攻击。 AWS WAF 是一个 Web 应用程序防火墙，可监控和拦截与转发到 Amazon API Gateway API、Amazon CloudFront 或 Application Load Balancer 的可配置规则匹配的 HTTP(s) 请求。要开始使用 AWS WAF，您可以将 AWS 托管式规则 与您自己的规则结合使用，也可以使用现有的 合作伙伴集成。

要管理 AWS WAF、AWS Shield Advanced 保护以及跨 AWS Organizations 的 Amazon VPC 安全组，您可以使用 AWS Firewall Manager。它允许您跨账户和应用程序集中配置和管理防火墙规则，从而更轻松地扩展常见规则的实施。通过使用 AWS Shield Advanced或 能够自动拦截向 您的 Web 应用程序发送非必要请求的解决方案，它还使您能够快速响应攻击。Firewall Manager 也可以与 AWS Network Firewall结合使用。AWS Network Firewall 是一种托管服务，使用规则引擎为您提供对有状态和无状态网络流量的精细控制。它支持 与 Suricata 兼容的 开源入侵防御系统（IPS，Intrusion Prevention System）规范，以便使用规则来保护您的工作负载。

未建立这种最佳实践的情况下暴露的风险等级： 低

实施指导

- 配置 Amazon GuardDuty：GuardDuty 是一种威胁检测服务，可持续监控恶意活动和未经授权的行为，从而保护您的 AWS 账户 和工作负载。启用 GuardDuty 并配置自动化警报。
  - Amazon GuardDuty
  - 实验室：自动部署检测性控制
- 配置虚拟私有云（VPC）流日志：VPC 流日志功能使您能够进一步捕获有关传入和传出 VPC 中网络接口的 IP 流量信息。流日志数据可以发布到 Amazon CloudWatch Logs 和 Amazon Simple Storage Service（Amazon S3）。创建流日志后，您可以在选定目标中检索和查看其数据。
- 考虑使用 VPC 流量径向：流量镜像是一项 Amazon VPC 功能，您可以用它从 Amazon Elastic Compute Cloud（Amazon EC2）实例的弹性网络接口复制网络流量，然后将其发送到带外安全和监控设备，以进行内容检查、威胁监控和故障排除。
  - VPC 流量镜像

资源

相关文档：

- AWS Firewall Manager
- Amazon Inspector
- Amazon VPC 安全性
- 开始使用 AWS WAF

相关视频：

- 用于各种 VPC 的 AWS Transit Gateway 参考架构
- 使用 Amazon CloudFront、AWS WAF 和 AWS Shield 提供应用程序加速和保护

相关示例：

- 实验室：自动部署 VPC

保护计算

计算资源包括 EC2 实例、容器、AWS Lambda 函数、数据库服务、IoT 设备等。这些计算资源类型中的每种资源均需使用不同的方法来予以保护。不过，在您考虑常见策略时，它们确实具备许多共性：深度防御、漏洞管理、减小攻击面、配置和操作自动化以及远距离执行操作。在此部分中，您将找到有关保护关键服务的计算资源的一般指南。对于使用的每项 AWS 服务，请您务必查看服务文档中的具体安全建议。

最佳实践

- SEC06-BP01 执行漏洞管理
- SEC06-BP02 缩小攻击面
- SEC06-BP03 实施托管服务
- SEC06-BP04 自动保护计算
- SEC06-BP05 帮助人员远程执行操作
- SEC06-BP06 验证软件完整性

SEC06-BP01 执行漏洞管理

频繁扫描和修补您的代码、依赖项和基础设施中的漏洞，以帮助防御新的威胁。

期望结果：制定并维护漏洞管理计划。定期扫描和修补资源，例如 Amazon EC2 实例、Amazon Elastic Container Service（Amazon ECS）容器和 Amazon Elastic Kubernetes Service（Amazon EKS）工作负载。为 AWS 托管的资源（如 Amazon Relational Database Service（Amazon RDS）数据库）配置维护时段。使用静态代码扫描检查应用程序源代码的常见问题。如果贵组织具备必要的技能或可以聘请外部人员协助，则不妨考虑执行 Web 应用程序渗透测试。

常见反模式：

- 未制定漏洞管理计划。
- 在不考虑严重性或风险规避的情况下执行系统修补。
- 使用已超过供应商提供的生命周期结束（EOL）日期的软件。
- 在分析安全问题之前，将代码部署到生产环境中。

建立此最佳实践的好处：

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

漏洞管理计划包括安全评估、识别问题、确定优先级，以及执行修补操作（问题解决流程的一部分）。自动化是持续扫描工作负载（以查找问题和意外网络暴露）并执行补救措施的关键。自动创建和更新资源可以节省时间，并降低配置错误产生进一步问题的风险。完善的漏洞管理计划还应考虑在软件生命周期的开发和部署阶段进行漏洞测试。在开发和部署期间实施漏洞管理有助于减少漏洞进入生产环境的机会。

实施漏洞管理计划需要很好地理解 AWS 责任共担模式，以及它与特定工作负载的关系。在责任共担模式下，AWS 负责保护 AWS Cloud 的基础设施。此基础设施由运行 AWS Cloud 服务的硬件、软件、网络和设施组成。您负责云中的安全（例如，Amazon EC2 实例的实际数据、安全配置和管理任务），并验证 Amazon S3 对象已正确分类和配置。漏洞管理方法也可能因您使用的服务而异。例如，AWS 管理我们的托管关系数据库服务 Amazon RDS 的修补，但您将负责修补自托管数据库。

AWS 提供一系列服务来帮助您制定漏洞管理计划。Amazon Inspector 持续扫描 AWS 工作负载是否存在软件问题和意外网络访问。AWS Systems Manager 补丁管理器可帮助跨 Amazon EC2 实例管理修补。AWS Security Hub 是一项云安全状况管理服务，有助于自动执行 AWS 安全检查并将安全警报集中起来，您可以通过该服务查看 Amazon Inspector 和 Systems Manager。

Amazon CodeGuru 可以使用静态代码分析，帮助识别 Java 和 Python 应用程序中的潜在问题。

实施步骤

- 配置 Amazon Inspector：Amazon Inspector 会自动检测新发布的 Amazon EC2 实例、Lambda 函数和推送给 Amazon ECR 的符合条件的容器映像，并立即扫描它们以查找软件问题、潜在缺陷和意外网络暴露。
- 扫描源代码：扫描库和依赖项，以确定是否有问题和缺陷。Amazon CodeGuru 会进行扫描，并提供建议以修复 Java 和 Python 应用程序的常见安全问题。OWASP Foundation 发布了一系列源代码分析工具（也称为 SAST 工具）。
- 实施一种机制来扫描和修补现有环境，以及作为 CI/CD 管道构建流程的一部分进行扫描：实施一种机制来扫描和修补依赖项和操作系统中的问题，以帮助防范新的威胁。定期运行这种机制。软件漏洞管理对于了解需要应用补丁或解决软件问题的位置至关重要。在持续集成/持续交付（CI/CD）管道中尽早嵌入漏洞评估，对潜在的安全问题进行优先修复。您的方法可能会因您使用的 AWS 服务而异。要检查 Amazon EC2 实例中运行的软件中的潜在问题，请在管道中添加 Amazon Inspector，以便在检测到问题或潜在缺陷时发出警报并停止构建过程。Amazon Inspector 会持续监控资源。您还可以使用开源产品（如 OWASP Dependency-Check、Snyk、OpenVAS、程序包管理器和 AWS Partner 工具）进行漏洞管理。
- 使用 AWS Systems Manager：您负责自己 AWS 资源的补丁管理，包括 Amazon Elastic Compute Cloud（Amazon EC2）实例、亚马逊云机器镜像（AMI）以及其他计算资源。AWS Systems Manager 补丁管理器使用安全相关的更新和其他类型的更新自动执行修补托管实例的流程。补丁管理器可用于在 Amazon EC2 实例上为操作系统和应用程序（包括 Microsoft 应用程序、Windows Service Pack 和基于 Linux 实例的次要版本升级）应用补丁。除了 Amazon EC2 之外，补丁管理器还可用于对本地服务器进行修补。

  有关支持的操作系统的列表，请参阅 Systems Manager 用户指南中的支持的操作系统。您可以扫描实例以单独查看缺失补丁的报告，也可以扫描并自动安装所有缺失的补丁。
- 使用 AWS Security Hub：Security Hub 可提供 AWS 中安全状况的全面视图。它跨多项 AWS 服务收集安全性数据，并以标准化格式提供这些调查结果，使您能够对 AWS 服务中的安全性调查结果进行优先级排序。
- 使用 AWS CloudFormation：AWS CloudFormation 是一项基础设施即代码（IaC）服务，可通过跨多个账户和环境实现资源部署自动化和资源架构标准化来帮助管理漏洞。

资源

相关文档：

- AWS Systems Manager
- AWS Lambda 安全性概述
- Amazon CodeGuru

- 使用新的 Amazon Inspector 改进了云工作负载的自动化漏洞管理
- 使用 Amazon Inspector 和 AWS Systems Manager 自动执行 AWS 中的漏洞管理和修复 – 第 1 部分

相关视频：

- 保护无服务器和容器服务
- 有关 Amazon EC2 实例元数据服务的安全最佳实践

SEC06-BP02 缩小攻击面

通过强化操作系统，并尽量减少所使用的组件、库和外部可用的服务，缩小暴露在意外访问下的危险。首先减少未使用的组件，无论它们是操作系统程序包、应用程序（适用于基于 Amazon Elastic Compute Cloud（Amazon EC2）的工作负载）还是您代码中的外部软件模块（适用于所有工作负载）。您可以找到许多面向常见的操作系统和服务器软件的强化和安全配置指南。例如，您可以从 互联网安全中心 开始并进行迭代。

在 Amazon EC2 中，您可以创建自己的亚马逊云机器镜像（AMI）并进行修补和强化，以帮助您满足企业的特定安全要求。您应用到 AMI 上的补丁和其他安全控制措施在其创建时生效，它们并非动态的，除非您在启动之后进行了修改，例如，使用 AWS Systems Manager 进行修改。

您可以使用 EC2 Image Builder 简化构建安全 AMI 的过程。EC2 Image Builder 可大幅减少创建和维护黄金镜像所需的工作，无需编写和维护自动化过程。在有软件更新可用时，Image Builder 自动生成新的镜像，无需用户手动迭代镜像工作版本。通过 EC2 Image Builder，您可以使用 AWS 提供的测试和自己的测试，在将镜像部署到生产环境中之前轻松地验证镜像的功能和安全性。您还可以应用 AWS 提供的安全设置来进一步保护自己的镜像，满足内部安全标准。例如，您可以使用 AWS 提供的模板，生成符合安全技术实施指南（STIG，Security Technical Implementation Guide）标准的镜像。

使用第三方静态代码分析工具，您可以确定常见的安全问题，例如未检查的函数输入边界，以及适用的通用漏披露（CVE，Common Vulnerabilities and Exposures）。您可以对所支持的语言使用 Amazon CodeGuru 。您还可以使用第三方依赖关系检查工具，确定代码链接的库是否是最新版本、它们是否不含 CVE，并确保您拥有符合您软件政策要求的许可条件。

使用 Amazon Inspector，您可以针对 CVE，对您的实例执行配置评估、根据安全基准执行评估以及实现缺陷通知自动化。Amazon Inspector 在生产实例或构建管道中运行，它会在发现结果时通知开发人员和工程师。您可以通过编程方式访问调查结果，并将您的团队引导至待办事项和错误跟踪系统。 EC2 Image Builder 可通过自动化修补、AWS 提供的安全策略实施和其他自定义来维护服务器映像 (AMI)。当使用容器时，在您的构建管道中对您的映像存储库定期实施 ECR 映像扫描 ，以便在您的容器中查找 CVE。

尽管 Amazon Inspector 和其他工具能够有效地确定配置和存在的任何 CVE，但也需要使用其他方法在应用程序级别测试您的工作负载。 模糊 是一种众所周知的查错方法，可自动将格式不正确的数据注入到您应用程序的输入字段和其他区域来查错。

未建立此最佳实践暴露的风险等级： 高

实施指导

- 强化操作系统：配置操作系统以符合最佳实践。
  - 保护 Amazon Linux
  - 保护 Microsoft Windows Server
- 强化容器化资源：配置容器化资源以符合安全最佳实践。
- 实施 AWS Lambda 最佳实践。
  - AWS Lambda 最佳实践

资源

相关文档：

- AWS Systems Manager
- 使用 Amazon EC2 Systems Manager 替换堡垒主机
- AWS Lambda 安全性概述

相关视频：

- 在 Amazon EKS 上运行高安全性工作负载
- 保护无服务器和容器服务
- 有关 Amazon EC2 实例元数据服务的安全最佳实践

相关示例：

- 实验室：自动部署 Web 应用程序防火墙

SEC06-BP03 实施托管服务

实施用于托管资源的服务，例如 Amazon Relational Database Service（Amazon RDS）、AWS Lambda 和 Amazon Elastic Container Service（Amazon ECS），以便在责任共担模式中减少安全维护任务。例如，Amazon RDS 可帮助您设置、操作和扩展关系数据库，并自动执行管理任务，例如硬件预置、数据库设置、修补和备份。这意味着您将有更多的空闲时间，因此可以专注于通过 AWS Well-Architected Framework 中所述的其他方法来保护您的应用程序。使用 Lambda，无需使用预置或托管服务器即可运行代码，因此您只需在代码级别专注于连接、调用和安全性，而不是基础设施或操作系统级别。

未建立此最佳实践暴露的风险等级： 中

实施指导

- 探索可用的服务：探索、测试和实施管理资源的服务，例如 Amazon RDS、AWS Lambda 和 Amazon ECS。

资源

相关文档：

- AWS 网站
- AWS Systems Manager
- 使用 Amazon EC2 Systems Manager 替换堡垒主机
- AWS Lambda 安全性概述

相关视频：

- 在 Amazon EKS 上运行高安全性工作负载
- 保护无服务器和容器服务
- 有关 Amazon EC2 实例元数据服务的安全最佳实践

相关示例：

- 实验室：AWS Certificate Manager 请求公有证书

SEC06-BP04 自动保护计算

自动执行计算保护机制，包括管理漏洞、缩小攻击面和管理资源。此自动化将帮助您投入时间以保护工作负载的其他方面，并降低人为犯错的风险。

未建立这种最佳实践的情况下暴露的风险等级： 中

实施指导

- 自动管理配置：使用配置管理服务或工具自动实施安全配置并对其进行验证。
  - AWS Systems Manager
  - AWS CloudFormation
  - 实验室：自动部署 VPC
  - 实验室：自动部署 EC2 Web 应用程序

- 自动修补 Amazon Elastic Compute Cloud（Amazon EC2）实例：AWS Systems Manager Patch Manager 使用安全相关的更新和其他类型的更新来自动执行修补托管实例的流程。您可以使用 Patch Manager 为操作系统和应用程序应用修补程序。
  - AWS Systems Manager 补丁管理器
  - 使用 AWS Systems Manager Automation 集中完成多账户和多区域的修补

- 实施入侵检测和预防：实施入侵检测和预防工具，以监控并停止实例上的恶意活动。
- 考虑使用 AWS Partner 解决方案：AWS 合作伙伴提供数百种业界领先的产品，这些产品与您的本地环境中的现有控制措施等效、相同或与之集成。这些产品对现有 AWS 服务起到补充作用，使您能够在云和本地部署环境中部署全面的安全架构，进而实现更无缝的体验。
  - 基础设施安全性

资源

相关文档：

- AWS CloudFormation
- AWS Systems Manager
- AWS Systems Manager 补丁管理器
- 使用 AWS Systems Manager Automation 集中完成多账户和多区域的修补

- 基础设施安全性
- 使用 Amazon EC2 Systems Manager 替换堡垒主机
- AWS Lambda 安全性概述

相关视频：

- 在 Amazon EKS 上运行高安全性工作负载
- 保护无服务器和容器服务
- 有关 Amazon EC2 实例元数据服务的安全最佳实践

相关示例：

- 实验室：自动部署 Web 应用程序防火墙
- 实验室：自动部署 EC2 Web 应用程序

SEC06-BP05 帮助人员远程执行操作

移除交互式访问功能可降低人为错误的风险以及手动配置或管理的可能性。例如，通过更改管理工作流，使用基础设施即代码部署 Amazon Elastic Compute Cloud（Amazon EC2）实例，然后使用 AWS Systems Manager 等工具管理 Amazon EC2 实例，而不是允许直接访问或通过堡垒主机进行访问。AWS Systems Manager 可以使用 自动化 工作流、文档 （行动手册）和 Run Command等功能自动执行多种维护和部署任务。AWS CloudFormation 堆栈从管道进行构建，并能够自动执行您的基础设施部署和管理任务，而无需直接使用 AWS Management Console或 API。

未建立此最佳实践暴露的风险等级： 低

实施指导

- 替换控制台访问：用 AWS Systems Manager Run Command 替换实例的控制台访问（SSH 或 RDP），以自动管理任务。

- AWS Systems Manager Run Command

资源

相关文档：

- AWS Systems Manager
- AWS Systems Manager Run Command
- 使用 Amazon EC2 Systems Manager 替换堡垒主机
- AWS Lambda 安全性概述

相关视频：

- 在 Amazon EKS 上运行高安全性工作负载
- 保护无服务器和容器服务
- 有关 Amazon EC2 实例元数据服务的安全最佳实践

相关示例：

- 实验室：自动部署 Web 应用程序防火墙

SEC06-BP06 验证软件完整性

实施一些机制（例如代码签名），以确保工作负载中使用的软件、代码和库来自可信的来源且未被篡改。例如，您应验证二进制文件和脚本的代码签名证书以确认作者，并确保证书自作者创建以来未被篡改。AWS Signer 通过集中管理代码签名生命周期，包括签名证书以及公有和私有密钥，帮助确保代码的可信度和完整性。您可以了解如何对 AWS Lambda使用代码签名的高级模式和最佳实践。此外，通过将您下载的软件的校验和与提供商提供的校验和进行对比，可帮助确保它未被篡改。

未建立这种最佳实践的情况下暴露的风险等级： 低

实施指导

- 调查机制：代码签名是一种可用来验证软件完整性的机制。
  - NIST：代码签名的安全注意事项

资源

相关文档：

- AWS Signer
- 新增 – 代码签名，用于 AWS Lambda 的可信度和完整性控制措施

数据保护

在为任何工作负载设计架构之前，您应确定可能影响安全性的基本实践。例如，数据分级提供了一种基于敏感程度对数据进行分类的方法，加密通过让未经授权的用户无法获知数据的真正内容来保护数据。这些方法非常重要，因为它们有助于实现诸如履行监管义务或避免处理不当等目标。

在 AWS 中，实施数据保护时可以使用很多不同的方法。以下部分介绍了如何使用这些方法。

主题

数据分级

数据分类提供了一种基于关键性和敏感度对组织数据进行分类的方法，以帮助您确定适当的保护和保留控制措施。

最佳实践

SEC07-BP01 识别工作负载内的数据

了解工作负载所处理数据的类型和分类、关联的业务流程、数据存储位置以及数据所有者至关重要。您还应了解工作负载的适用法律和合规性要求，以及需要执行的数据控制措施。识别数据是数据分类过程的第一步。

建立此最佳实践的好处：

通过数据分类，工作负载所有者可以识别存储敏感数据的位置，并确定访问和共享这些数据的方式。

数据分类旨在回答以下问题：

- 您拥有什么类型的数据？

  可能包括以下数据：

  - 知识产权（IP），例如商业秘密、专利或合同协议。
  - 受保护健康信息（PHI），例如包含与个人相关的病史信息的医疗记录。
  - 个人身份信息（PII），例如姓名、地址、出生日期和国民身份证或登记号码。
  - 信用卡数据，例如主账号（PAN）、持卡人姓名、到期日期和服务代码编号。
  - 敏感数据存储在哪里？
  - 谁可以访问、修改和删除数据？
  - 了解用户权限对于防范潜在的数据误操作至关重要。
- 谁可以执行创建、读取、更新和删除（CRUD）操作？
  - 通过了解谁可以管理数据权限，对潜在的权限升级进行说明。
- 如果数据被无意中披露、更改或删除，可能会产生什么业务影响？
  - 了解数据被修改、删除或无意中披露的风险后果。

通过了解这些问题的答案，您可以采取以下行动：

- 缩小敏感数据的范围（如敏感数据位置的数量），并限制敏感数据的访问权限，仅限经批准的用户进行访问。
- 了解不同的数据类型，以便实施适当的数据保护机制和技术，如加密、数据丢失防护以及身份和权限管理。
- 通过为数据提供正确的控制目标来优化成本。
- 自信地回答监管机构和审计人员提出的关于数据类型和数量，以及不同敏感度的数据如何相互隔离的问题。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

数据分类是确定数据敏感度的行为。该行为可能涉及标记，以使数据易于搜索和跟踪。数据分类还可减少数据的重复，这有助于降低存储和备份成本，同时加快搜索过程。

使用 Amazon Macie 等服务大规模将敏感数据的发现和分类自动化。其他服务（如 Amazon EventBridge 和 AWS Config）可用于自动修复数据安全问题，如未加密的 Amazon Simple Storage

Service（Amazon S3）存储桶和 Amazon EC2 EBS 卷或未标记的数据资源。有关 AWS 服务集成的完整列表，请参阅 EventBridge 文档。

通过使用 Amazon Comprehend（一种自然语言处理（NLP）服务，该服务使用机器学习（ML）在非结构化文本中查找人物、地点、情感和主题等洞察和关系），可以在非结构化数据（如客户电子邮件、支持工单、产品评论和社交媒体）中检测 PII。有关可协助进行数据识别的 AWS 服务的列表，请参阅使用 AWS 服务检测 PHI 和 PII 数据的常见技术。

另一种支持数据分类和保护的方法是 AWS 资源标记。通过标记，可以为 AWS 资源分配元数据，您可以使用这些元数据来管理、识别、组织、搜索和筛选资源。

在某些情况下，您可能会选择标记整个资源（例如 S3 存储桶），尤其当特定工作负载或服务预计将存储已知数据分类的进程或传输时。

在适当情况下，您可以标记 S3 存储桶而不是单个对象，以便于管理和安全维护。

实施步骤

检测 Amazon S3 内的敏感数据：

1. 开始之前，请确保您拥有访问 Amazon Macie 控制台和执行 API 操作的适当权限。有关其他详细信息，请参阅开始使用 Amazon Macie。
2. 当敏感数据驻留在 Amazon S3 中时，使用 Amazon Macie 执行自动化数据发现。
   - 使用开始使用 Amazon Macie 指南为敏感数据发现结果配置存储库，并为敏感数据创建发现作业。
   - 如何使用 Amazon Macie 预览 S3 存储桶中的敏感数据。

     默认情况下，Macie 通过使用我们建议用于自动化敏感数据发现的一组托管数据标识符来分析对象。您可以定制分析，方法是将 Macie 配置为在为您的账户或组织执行自动化敏感数据发现时，使用特定的托管数据标识符、自定义数据标识符和允许列表。您可以通过排除特定的存储桶（例如，通常存储 AWS 日志记录数据的 S3 存储桶）来调整分析范围。
3. 要配置和使用自动化敏感数据发现，请参阅使用 Amazon Macie 执行自动化敏感数据发现。
4. 您也可以考虑针对 Amazon Macie 的自动化数据发现。

检测 Amazon RDS 内的敏感数据：

有关 Amazon Relational Database Service（Amazon RDS）数据库中数据发现的更多信息，请参阅使用 Macie 为 Amazon RDS 数据库启用数据分类。

检测 DynamoDB 内的敏感数据：

- 使用 Macie 检测 DynamoDB 内的敏感数据介绍了如何使用 Amazon Macie 通过将数据导出到 Amazon S3 进行扫描来检测 Amazon DynamoDB 表中的敏感数据。

AWS 合作伙伴解决方案：

- 考虑使用我们广泛的 AWS Partner Network。AWS 合作伙伴拥有广泛的工具和合规性框架，可直接与 AWS 服务集成。合作伙伴可以为您提供量身定制的治理和合规性解决方案，以帮助您满足组织需求。
- 有关数据分类中的定制解决方案，请参阅监管和合规性要求时代的数据治理。

通过使用 AWS Organizations 创建和部署策略，您可以自动实施贵组织所采用的标记标准。您可以通过标签策略来指定规则，规则中定义有效的密钥名称以及对每个密钥有效的值。您可以选择仅监控，这使您有机会评估和清理现有标签。标签符合所选标准后，您可以在标签策略中启用强制执行，以防止创建不合规的标签。有关更多详细信息，请参阅使用 AWS Organizations 中的服务控制策略保护用于授权的资源标签，以及关于防止标签被授权主体以外的人员修改的示例策略。

- 要开始在 AWS Organizations 中使用标签策略，强烈建议您先遵循开始使用标签策略中的工作流程，然后再使用更高级的标签策略。在扩展到整个组织单位（OU）或组织之前，了解将简单的标签策略附加到单个账户的效果，可以在强制遵守某项标签策略之前看到该标签策略的效果。开始使用标签策略提供了指向更高级策略相关任务的说明的链接。
- 不妨考虑评估一下数据分类白皮书中列出的支持数据分类的其他 AWS 服务和功能。

资源

相关文档：

- 开始使用 Amazon Macie
- 使用 Amazon Macie 执行自动化数据发现
- 开始使用标签策略
- 检测 PII 实体

相关博客：

- 如何使用 Amazon Macie 预览 S3 存储桶中的敏感数据。
- 使用 Amazon Macie 执行自动化敏感数据发现。

- 使用 AWS 服务检测 PHI 和 PII 数据的常见技术
- 使用 Amazon Comprehend 检测和编辑 PII
- 使用 AWS Organizations 中的服务控制策略保护用于授权的资源标签
- 使用 Macie 为 Amazon RDS 数据库启用数据分类
- 使用 Macie 检测 DynamoDB 中的敏感数据
-

相关视频：

- 使用 Amazon Macie 的事件驱动型数据安全性
- 用于数据保护和治理的 Amazon Macie
- 利用允许列表对敏感数据调查结果进行微调

SEC07-BP02 定义数据保护控制措施

根据数据分类级别保护数据。例如，使用相关建议保护分类为公共的数据，同时使用其他控制措施保护敏感数据。

通过使用资源标签、根据敏感度（可能还包括限制性条款、飞地或感兴趣的社区）划分 AWS 账户、IAM 策略、AWS Organizations SCP、AWS Key Management Service（AWS KMS）和 AWS CloudHSM，您可以定义并实施您的数据分类和加密保护策略。例如，如果您的项目具有包含极关键数据的 S3 存储桶或者处理机密数据的 Amazon Elastic Compute Cloud（Amazon EC2）实例，则可以使用 `Project=ABC` 标签对其进行标记。只有您的直属团队知道项目代码的含义，它提供了一种使用基于属性的访问控制措施的方法。您可以通过关键策略和授权定义对 AWS KMS 加密密钥的访问级别，以确保只有适当的服务可以通过安全机制访问敏感内容。如果您正在根据标签做出授权决定，您应确保在 AWS Organizations 中使用标签策略适当定义对于标签的权限。

未建立此最佳实践暴露的风险等级： 高

实施指导

- 定义您的数据识别和分类架构：对数据执行标识和分类，用于评估您要存储的数据的潜在影响和类型，并确定谁可以访问数据。
  - AWS 文档
- 发现可用的 AWS 控制措施：对于您正在使用或计划使用的 AWS 服务，发现安全控制措施。许多服务在其文档中都会提供一个安全部分。

  - AWS 文档
- 确定 AWS 合规性资源：确定 AWS 为您提供帮助的资源。
  - https://aws.amazon.com/compliance/

资源

相关文档：

- AWS 文档
- 数据分类白皮书
- 开始使用 Amazon Macie
- 缺少文本

相关视频：

- 新 Amazon Macie 简介

SEC07-BP03 自动识别和分类

自动识别和分类数据可帮助您实施正确的控制措施。在这方面实现自动化而不是允许人员直接访问，可以降低人为犯错和漏洞的风险。您应使用 Amazon Macie等工具执行评估，这些工具使用机器学习来自动发现、分类和保护 Amazon Macie 中的敏感数据。AWS 可以识别个人身份信息（PII，Personally Identifiable Information）或知识产权之类的敏感数据，并为您提供控制面板和警报，让您了解此类数据的访问或移动情况。

未建立此最佳实践暴露的风险等级： 中

实施指导

- 使用 Amazon Simple Storage Service（Amazon S3）清单：Amazon S3 清单是可以用来审核和报告对象的复制和加密状态的工具之一。
  - Amazon S3 清单
- 考虑使用 Amazon Macie：Amazon Macie 使用机器学习来自动发现存储在 Amazon S3 中的数据，并对其进行分类。
  - Amazon Macie

资源

相关文档：

- Amazon Macie
- Amazon S3 清单
- 数据分类白皮书
- 开始使用 Amazon Macie

相关视频：

- 新 Amazon Macie 简介

SEC07-BP04 定义数据生命周期管理

您定义的生命周期策略应基于敏感度级别以及法律和组织要求。应考虑您的数据保留期限、数据销毁流程、数据访问管理、数据转换和数据共享等方面。当选择数据分类方法时，请平衡可用性与访问权限。您还应考虑多种访问级别及其细微差别，以便针对每个级别实施安全且有效的方法。始终采用深度防御方法并减少人工访问数据次数以及数据转换、删除或复制机制。例如，要求用户对应用程序执行严格身份验证，并为应用程序而不是用户授予执行远程操作的必要访问权限。此外，确保用户来自可信网络路径并要求其获取解密密钥。使用控制面板和自动报告等工具为用户提供数据信息，而不是让他们直接访问数据。

未建立这种最佳实践的情况下暴露的风险等级： 低

实施指导

- 识别数据类型：确定您正在工作负载中存储或处理的数据类型。这些数据可以是文本、图像、二进制数据库等。

资源

相关文档：

- 数据分类白皮书
- 开始使用 Amazon Macie

相关视频：

- 新 Amazon Macie 简介

保护静态数据

静态数据 代表您在工作负载期间的任意时间段内保留在非易失性存储器中的任何数据。其中包括数据块存储、对象存储、数据库、存档、IoT 设备和用来保留数据的任何其他存储介质。在实施了加密和适当的访问控制时，保护静态数据可以降低未经授权访问的风险。

加密和令牌化是两个重要但不同的数据保护方案。

令牌化 是一个支持您定义令牌以表示其他敏感信息的过程（例如代表客户信用卡号的令牌）。令牌自身必须没有任何意义，而且不能是从它令牌化的数据衍生而来 – 因此，无法将加密摘要用作令牌。通过认真规划令牌化方法，您可以为内容提供额外保护，并确保满足合规性要求。例如，如果您使用令牌而不是信用卡号，就可以缩小信用卡处理系统的合规性范围。

加密 可以将内容转换为这样一种形式：如果用户没有将这些内容解密为纯文本所需的密钥，就无法读取。令牌化和加密都可用于酌情保护信息。此外，可以使用掩码这种技术编辑数据的某个部分，以使剩余的数据不被视为敏感数据。例如，PCI-DSS 允许在合规性范围边界之外保留卡号的最后四位数字，以供索引使用。

审计加密密钥的使用： 确保您了解并审计加密密钥的使用，以确保对密钥正确实施访问控制措施。例如，使用 AWS KMS 密钥的任何 AWS 服务都会在 AWS CloudTrail 中记录每次密钥使用。随后，您可以使用 Amazon CloudWatch 等工具查询 AWS CloudTrail，以确保您的密钥的所有使用都有效。

最佳实践

- SEC08-BP01 实施安全密钥管理
- SEC08-BP02 强制实施静态加密
- SEC08-BP03 自动执行静态数据保护
- SEC08-BP04 强制实施访问控制
- SEC08-BP05 使用机制限制对数据的访问

SEC08-BP01 实施安全密钥管理

安全密钥管理包括密钥材料的存储、轮换、访问控制和监控，这些都是保护工作负载的静态数据安全所必需的。

期望的结果： 一种可扩展、可重复且自动化的密钥管理机制。该机制应能够强制对密钥材料实施最低权限访问，在密钥可用性、机密性和完整性之间取得适当的平衡。应监控密钥的使用，并通过自动化流程轮换密钥材料。密钥材料永远不应能够通过人员的身份来访问。

常见反模式：

- 由人类访问未加密的密钥材料。
- 创建自定义加密算法。
- 访问密钥材料的权限过于宽泛。

建立此最佳实践的好处： 通过为您的工作负载建立安全的密钥管理机制，您可以帮助保护您的内容免遭未经授权的访问。此外，您可能需要遵守对数据进行加密的监管要求。有效的密钥管理解决方案可以提供符合这些法规的技术机制，进而保护密钥材料。

未建立这种最佳实践的情况下暴露的风险等级： 高

实施指导

许多监管要求和最佳实践都将静态数据加密作为一项基本的安全控制措施。为了遵守这种控制措施，您的工作负载需要一种机制，来安全地存储和管理用于加密静态数据的密钥材料。

AWS 的 AWS Key Management Service（AWS KMS）可为 AWS KMS 密钥提供持久、安全和冗余的存储空间。 许多 AWS 服务都与 AWS KMS 集成 来支持对您的数据进行加密。AWS KMS 使用经 FIPS 140-2 Level 3 验证的硬件安全模块来保护您的密钥。不存在以纯文本格式导出 AWS KMS 密钥的机制。

使用多账户策略部署工作负载时， 最佳实践 是将 AWS KMS 密钥与使用密钥的工作负载保存在同一个账户中。在这种分布式模型中，由应用程序团队负责管理 AWS KMS 密钥。在其他用例中，组织可以选择将 AWS KMS 密钥存储到集中式账户中。这种集中式结构需要额外的策略，以实现工作负载账户访问集中式账户中存储的密钥所需的跨账户访问，但可能更适用于多个 AWS 账户 账户共享单个密钥的用例。

无论密钥材料存放在哪里，都应通过使用 密钥策略 和 IAM 策略来严格控制对密钥的访问。密钥策略是控制对 AWS KMS 密钥的访问的主要方式。此外，AWS KMS 密钥授权可以提供对 AWS 服务的访问权限，从而代表您加密和解密数据。花点时间回顾 AWS KMS 密钥访问控制的最佳实践访问 AWS 资源。

最佳实践是监控加密密钥的使用情况，以检测异常的访问模式。使用 AWS 托管密钥和 AWS KMS 中存储的客户自主管理型密钥执行的操作可以记录在 AWS CloudTrail 中，并应定期进行审查。应特别注意监控密钥销毁事件。为了减少意外或恶意破坏密钥材料的情况，密钥销毁事件不会立即删除密钥材

料。尝试删除 AWS KMS 中的密钥时会经历一个 等待期，默认为 30 天，这让管理员有时间查看这些操作并在必要时回滚请求。

大多数 AWS 服务使用 AWS KMS 的方式对您来说都是透明的，您只需要决定是使用 AWS 托管密钥还是客户自主管理型密钥。如果您的工作负载需要直接使用 AWS KMS 来加密或解密数据，则最佳实践是使用 信封加密 来保护您的数据。此 AWS 加密开发工具包 可为您的应用程序提供客户端加密原语，来实施信封加密并与 AWS KMS 集成。

实施步骤

1. 为密钥确定合适的 密钥管理选项 （AWS 托管或客户自主管理）。
   - 为便于使用，AWS 为大多数服务提供 AWS 自有和 AWS 托管密钥，这样，无需管理密钥材料或密钥策略，即可提供静态加密功能。
   - 使用客户自主管理型密钥时，请考虑使用默认密钥存储，以便在敏捷性、安全性、数据主权和可用性之间取得最佳平衡。其他用例可能需要使用附带 AWS CloudHSM 的自定义密钥存储或使用外部密钥存储访问 AWS 资源。
2. 查看您用于工作负载的服务列表，以了解 AWS KMS 如何与该服务集成。例如，EC2 实例可以使用加密的 EBS 卷，验证从这些卷创建的 Amazon EBS 快照是否也使用客户自主管理型密钥进行加密，并减少未加密快照数据的意外泄露。
   - AWS 服务如何使用 AWS KMS
   - 有关 AWS 服务提供的加密选项的详细信息，请参阅该服务的用户指南或开发人员指南中的“静态加密”主题。
3. 实施 AWS KMS：AWS KMS 使您可以轻松创建和管理密钥，并控制各种 AWS 服务和应用程序中的加密使用情况。
   - 入门：AWS Key Management Service（AWS KMS）
   - 请查看 AWS KMS 密钥访问控制的最佳实践访问 AWS 资源。
4. 考虑 AWS Encryption SDK：当您的应用程序需要加密客户端数据时，使用包含 AWS KMS 集成的 AWS Encryption SDK。
   - AWS Encryption SDK
5. 启用 IAM Access Analyzer 以自动审查是否存在过于宽泛的 AWS KMS 密钥策略并相应地发出通知。
6. 启用 Security Hub 以便在密钥策略配置错误、计划删除密钥或存在未启用自动轮换的密钥时，接收通知。
7. 确定适合您的 AWS KMS 密钥的日志记录级别。由于对 AWS KMS 的调用（包括只读事件）会被记录下来，因此与 AWS KMS 关联的 CloudTrail 日志可能会变得非常庞大。

- 一些组织倾向于将 AWS KMS 日志活动分成单独的跟踪。有关更多详细信息，请参阅 使用 CloudTrail 记录 AWS KMS API 调用 部分（位于 AWS KMS 开发者指南中）。

资源

相关文档：

- AWS Key Management Service
- AWS 加密服务和工具
- 利用加密保护 Amazon S3 数据
- 信封加密
- 数字主权承诺
- 揭开 AWS KMS 密钥操作的神秘面纱、自带密钥、自定义密钥库和密文可移植性
- AWS Key Management Service 加密详情

相关视频：

- AWS 中的加密原理
- 在 AWS 上保护您的数据块存储
- AWS data protection: Using locks, keys, signatures, and certificates

相关示例：

- 使用 AWS KMS 实施高级访问控制机制

SEC08-BP02 强制实施静态加密

您应该强制对静态数据使用加密。加密后，即使遭到未经授权访问或意外泄露，也能保持敏感数据的机密性。

期望结果：私有数据在静态时应默认加密。加密有助于保持数据的机密性，并提供额外一层保护，防止有意或无意的数据泄露或外流。如果不首先对数据进行解密，则无法读取或访问加密的数据。任何未加密便存储的数据都应进行清点和控制。

常见反模式：

- 不使用默认加密配置。
- 提供对解密密钥过于宽松的访问权限。
- 不监控加密和解密密钥的使用。
- 未加密便存储数据。
- 对所有数据使用相同的加密密钥，而不考虑数据用途、类型和分类。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

将加密密钥映射到工作负载中的数据分类。当对数据使用单个或非常少量的加密密钥时，这种方法有助于防止过于宽松的访问（请参阅SEC07-BP01 识别工作负载内的数据）。

AWS Key Management Service（AWS KMS）与许多 AWS 服务集成，使加密静态数据更加轻松。例如，在 Amazon Simple Storage Service（Amazon S3）中，您可以对存储桶设置默认加密，以自动加密新对象。使用 AWS KMS 时，请考虑需要对数据进行多严格的限制。默认和服务控制的 AWS KMS 密钥由 AWS 代表您进行管理和使用。对于需要对底层加密密钥进行精细访问的敏感数据，请考虑使用客户自主管理型密钥（CMK）。您可以完全控制 CMK，包括通过使用密钥策略进行轮换和访问管理。

此外，Amazon Elastic Compute Cloud（Amazon EC2）和 Amazon S3 支持通过设置默认加密来强制加密。您可以使用 AWS Config 规则 自动检查您已使用了加密，例如针对 Amazon Elastic Block Store（Amazon EBS）卷、Amazon Relational Database Service（Amazon RDS）实例和 Amazon S3 存储桶。

AWS 还提供客户端加密选项，使您能够在将数据上传到云之前对数据进行加密。AWS Encryption SDK 提供了一种使用信封加密对数据进行加密的方法。您提供包装密钥，AWS Encryption SDK 为它加密的每个数据对象生成一个唯一数据密钥。如果需要托管的单租户硬件安全模块（HSM），请考虑 AWS CloudHSM。AWS CloudHSM 使您可在通过 FIPS 140-2 Level 3 验证的 HSM 上生成、导入和管理加密密钥。AWS CloudHSM 的一些使用案例包括保护用于签发证书颁发机构（CA）的私有密钥，以及为 Oracle 数据库启用透明数据加密（TDE）。AWS CloudHSM 客户端开发工具包提供的软件使您可在将数据上传到 AWS 之前，使用存储在 AWS CloudHSM 中的密钥对客户端数据进行加密。Amazon DynamoDB Encryption Client 还使您可在将项目上传到 DynamoDB 表之前，对项目进行加密和签名。

实施步骤

- 对 Amazon S3 强制实施静态加密：实施 Amazon S3 存储桶默认加密。

为新的 Amazon EBS 卷配置默认加密：指定所有新创建的 Amazon EBS 卷要以加密形式创建，并选择使用 AWS 提供的默认密钥或您创建的密钥。

配置加密亚马逊云机器镜像（AMI）：通过复制启用加密功能的现有 AMI，可自动加密根卷和快照。

配置 Amazon RDS 加密：通过使用加密选项，配置对您的 Amazon RDS 数据库集群和静态快照的加密。

使用策略创建和配置 AWS KMS 密钥，以限制对每个数据分类的相应主体的访问：例如，创建一个 AWS KMS 密钥用于加密生产数据，创建一个不同的密钥用于加密开发或测试数据。您还可以提供对其他 AWS 账户的密钥访问权限。不妨考虑分开设立开发环境和生产环境的账户。如果您的生产环境需要解密开发账户中的构件，您可以编辑用于加密开发构件的 CMK 策略，使生产账户有能力解密这些构件。然后，生产环境可以摄取解密后的数据以用于生产。

在其他 AWS 服务中配置加密：对于您使用的其他 AWS 服务，请查看该服务的安全文档，以确定该服务的加密选项。

资源

相关文档：

- AWS 加密工具
- AWS 文档
- AWS Encryption SDK
- AWS KMS 加密详情白皮书
- AWS Key Management Service
- AWS 加密服务和工具
- Amazon EBS 加密
- Amazon EBS 卷的默认加密
- 加密 Amazon RDS 资源
- 如何为 Amazon S3 存储桶启用默认加密？
- 利用加密保护 Amazon S3 数据

相关视频：

- AWS 中的加密原理
- 在 AWS 上保护您的数据块存储

SEC08-BP03 自动执行静态数据保护

利用自动化工具持续验证和实施静态数据控制措施，例如，确保只存在经过加密的存储资源。您可以自动确认所有 EBS 卷都已经过加密，方法是 使用 AWS Config 规则。AWS Security Hub 还可以按照安全标准执行自动化检查，以验证多种不同的控制措施。此外，您的 AWS Config 规则可以自动 修复不合规的资源。

未建立此最佳实践暴露的风险等级： 中

实施指导

静态数据 代表您在工作负载期间的任意时间段内保留在非易失性存储器中的任何数据。其中包括数据块存储、对象存储、数据库、存档、IoT 设备和用来保留数据的任何其他存储介质。在实施了加密和适当的访问控制时，保护静态数据可以降低未经授权访问的风险。

强制实施静态加密：您应确保只以加密的方式存储数据。AWS KMS 与很多 AWS 服务无缝集成，使您能够更轻松地加密所有静态数据。例如，在 Amazon Simple Storage Service（Amazon S3）中，您可以对存储桶设置 默认加密 ，以自动加密所有的新对象。此外，Amazon EC2 和 Amazon S3 支持通过设置默认加密来强制加密。您可以使用 AWS 托管 Config 规则 自动检查您已使用了加密，例如针对 EBS 卷、Amazon Relational Database Service（Amazon RDS）实例和 Amazon S3 存储桶。

资源

相关文档：

- AWS 加密工具
- AWS 加密开发工具包

相关视频：

- AWS 中的加密原理
- 在 AWS 上保护您的数据块存储

SEC08-BP04 强制实施访问控制

为了帮助保护静态数据，请使用隔离和版本控制等机制强制实施访问控制，并应用最低权限原则。防止允许公众访问您的数据。

期望结果：验证只有获得授权的用户才能按照“需要知晓”的原则访问数据。通过定期备份和版本控制来保护您的数据，防止数据被有意或无意地修改或删除。将关键数据与其他数据隔离，以保护其机密性和数据完整性。

常见反模式：

- 将具有不同敏感度要求或分类的数据存储在一起。
- 解密密钥的权限过于宽松。
- 数据分类不当。
- 不保留重要数据的详细备份。
- 提供对生产数据的持久访问。
- 未审计数据访问，也未定期检查权限

在未建立这种最佳实践的情况下暴露的风险等级：低

实施指导

多项控制措施可以帮助保护静态数据，包括访问（使用最低权限）、隔离和版本控制。您应使用检测性机制（例如 AWS CloudTrail）和服务级别日志（例如 Amazon Simple Storage Service（Amazon S3）访问日志），审计对您的数据进行的访问。您应清点可公开访问的数据，并制定一份计划，以便随着时间的推移减少公开可用的数据量。

Amazon S3 Glacier 文件库锁定和 Amazon S3 对象锁定可为 Amazon S3 中的对象提供强制访问控制 – 利用合规性选项锁定文件库策略之后，在锁定过期之前，即使根用户也无法对其进行更改。

实施步骤

- 强制实施访问控制：强制实施最低权限访问控制，包括对加密密钥的访问。
- 根据不同分类级别隔离数据：针对数据分类级别使用不同的 AWS 账户，并使用 AWS Organizations 管理这些账户。
- 查看 AWS Key Management Service（AWS KMS）策略：查看 AWS KMS 策略中授予的访问级别。

- 查看 Amazon S3 存储桶和对象权限：定期查看 S3 存储桶策略中授予的访问级别。最佳做法是避免使用可公开读取或写入的存储桶。考虑使用 AWS Config 检测可公开访问的存储桶，并使用 Amazon CloudFront 提供 Amazon S3 中的内容。验证正确配置了不允许公开访问的存储桶，以防止公开访问。默认情况下，所有 S3 存储桶都是私有的，只有被明确授予访问权限的用户才可以访问。
- 启用 AWS IAM Access Analyzer：IAM Access Analyzer 可对 Amazon S3 存储桶进行分析，并在 S3 策略授予外部实体访问权限时生成结果。
- 在适当情况下启用 Amazon S3 版本控制和对象锁定。
- 使用 Amazon S3 清单：Amazon S3 清单可用来审计和报告 S3 对象的复制和加密状态。
- 查看 Amazon EBS 和 AMI 共享权限：共享权限将使得镜像和卷能够与您的工作负载外部的 AWS 账户共享。
- 查看 AWS Resource Access Manager：定期共享，以确定是否应继续共享资源。 您可以通过 Resource Access Manager 在您的 Amazon VPC 内共享资源，如 AWS Network Firewall 策略、Amazon Route 53 解析器规则和子网。定期审计共享资源，停止共享不再需要共享的资源。

资源

相关最佳实践：

- SEC03-BP01 定义访问要求
- SEC03-BP02 授予最低访问权限

相关文档：

- AWS KMS 加密详情白皮书
- 管理对 Amazon S3 资源的访问权限简介
- 管理对 AWS KMS 资源的访问权限概览
- AWS Config 规则
- Amazon S3 + Amazon CloudFront：云中的天作之合
- 使用版本控制
- 使用 Amazon S3 对象锁定来锁定对象
- 共享 Amazon EBS 快照
- 已共享的 AMI
- 在 Amazon S3 上托管单页应用程序

相关视频：

- 在 AWS 上保护您的数据块存储

SEC08-BP05 使用机制限制对数据的访问

禁止所有用户直接访问正常运行环境中的敏感数据和系统。例如，利用变更管理工作流程，借助工具管理 Amazon Elastic Compute Cloud（Amazon EC2）实例，而不是允许直接访问或通过堡垒主机进行访问。这可以使用 AWS Systems Manager Automation来实现，此功能将使用 包含您的任务执行步骤的 自动化文档。这些文档可以存储在源代码控制中、在运行之前接受对等审核，并接受全面测试以便最大程度降低风险（与 shell 访问相比）。企业用户可以使用一个仪表板而不是通过直接访问数据存储库来执行查询。当未使用 CI/CD 管道时，确定需要利用哪些控制措施和流程来充分提供通常禁用的 Break Glass 访问机制。

未建立这种最佳实践的情况下暴露的风险等级： 低

实施指导

- 实施可限制对数据的访问的机制：这些机制包括使用控制面板（例如 Amazon QuickSight），以向用户显示数据，而不是直接查询。
  - Amazon QuickSight
- 自动管理配置：远程执行操作，使用配置管理服务或工具自动实施安全配置并对其进行验证。避免使用堡垒主机或直接访问 EC2 实例。
  - AWS Systems Manager
  - AWS CloudFormation
  - AWS 上适用于 AWS CloudFormation 模板的 CI/CD 管道

资源

相关文档：

- AWS KMS 加密详情白皮书

相关视频：

- AWS 中的加密原理
- 在 AWS 上保护您的数据块存储

保护动态数据

传输中的数据 是指从一个系统发送到另一个系统的任何数据。这包括您工作负载中的资源之间的通信以及其他服务与您的最终用户之间的通信。通过为传输中数据提供适当级别的保护，您就可以保护工作负载数据的机密性和完整性。

保护 VPC 之间或本地位置传输的数据： 您可以使用 AWS PrivateLink ，在 Amazon Virtual Private Cloud（Amazon VPC）之间创建安全的专用网络连接，或创建本地与托管在 AWS 中的服务之间的连接。您可以访问 AWS 服务、第三方服务和其他 AWS 账户中的服务，就像它们在您的专用网络上一样。利用 AWS PrivateLink，您可以跨具有重叠 IP CIDR 的账户访问服务，而无需互联网网关或 NAT。您也无需配置防火墙规则、路径定义或路由表。流量保留在 Amazon 主干网上，并且不会穿过互联网，从而保护您的数据。您可以遵守行业特定的合规性法规，例如 HIPAA 和欧盟/美国隐私盾。AWS PrivateLink 与第三方解决方案无缝协作，创建了简化的全球网络，可让您加速迁移到云并利用可用的 AWS 服务。

最佳实践

- SEC09-BP01 实施安全密钥和证书管理
- SEC09-BP02 在传输中执行加密
- SEC09-BP03 自动检测意外数据访问
- SEC09-BP04 对网络通信进行身份验证

SEC09-BP01 实施安全密钥和证书管理

传输层安全性协议（TLS）证书用于保障网络通信的安全，确立网站、资源和工作负载在互联网上以及专用网络上的身份。

期望的结果： 一个安全的证书管理系统，可以在公钥基础设施（PKI，Public Key Infrastructure）中预置、部署、存储和续订证书。安全密钥和证书管理机制可防止证书私钥材料泄露，并定期自动续订证书。它还与其他服务集成，为工作负载内的计算机资源提供安全的网络通信和标识。密钥材料永远不应能够通过人员的身份来访问。

常见反模式：

- 在证书部署或续订流程中执行人工步骤。
- 在设计私有证书颁发机构（CA，Certificate Authority）时，对 CA 层次结构的关注不够。
- 对公共资源使用自签名证书。

建立此最佳实践的好处：

- 通过自动化的部署和续订流程简化证书管理
- 鼓励使用 TLS 证书对传输中数据进行加密
- 提高了证书颁发机构执行的证书操作的安全性和可审计性
- 在 CA 层次结构的不同层级上组织管理职责

未建立这种最佳实践的情况下暴露的风险等级： 高

实施指导

现代化工作负载广泛使用通过 PKI 协议（如 TLS）进行加密的网络通信。PKI 证书管理可能很复杂，但是，通过自动化的证书预置、部署和续订机制，可以减少与证书管理相关的麻烦。

AWS 提供了两种服务用于管理通用 PKI 证书： AWS Certificate Manager 和 AWS Private Certificate Authority（AWS Private CA）。ACM 是客户用于预置、管理和部署证书的主要服务，适用于面向公众的工作负载以及私有 AWS 工作负载。ACM 使用 AWS Private CA 颁发证书，并 集成 许多其他 AWS 托管服务，用于向工作负载提供安全的 TLS 证书。

利用 AWS Private CA，您可以建立自己的根证书颁发机构或从属证书颁发机构，并通过 API 颁发 TLS 证书。在 TLS 连接的客户端一侧控制和管理信任链的场景中，您可以使用这些类型的证书。除了 TLS 使用场景外，还可以使用 AWS Private CA 通过 自定义模板向 Kubernetes 容器组（pod）、Matter 设备产品认证、代码签名和其他使用场景颁发证书。您还可以使用 IAM Roles Anywhere ，向已经为其颁发了 X.509 证书（使用您的私有 CA 签名）的本地工作负载，提供临时 IAM 凭证。

除了 ACM 和 AWS Private CA 之外， AWS IoT Core 针对为物联网设备预置、管理和部署 PKI 证书提供专业化支持。AWS IoT Core 提供专门的机制，用于大规模 将物联网设备载入 到您的公钥基础设施中。

建立私有 CA 层次结构的注意事项

当您需要建立私有 CA 时，请务必重视预先正确设计 CA 层次结构。在创建私有 CA 层次结构时，最佳实践是将 CA 层次结构的每个级别部署到单独的 AWS 账户中。这个有意而为的步骤可减少 CA 层次结构中每个级别的暴露范围，使得发现 CloudTrail 日志数据中的异常变得更加简单，并可在某个账户遭到未经授权的访问时，缩小访问或影响的范围。根 CA 应位于自己的独立账户中，并且只能用于发布一个或多个中间 CA 证书。

然后，在不同于根 CA 账户的账户中创建一个或多个中间 CA，为最终用户、设备或其他工作负载发布证书。最后，从您的根 CA 向中间 CA 颁发证书，后者随之向您的最终用户或设备颁发证书。有关规划

CA 部署和设计 CA 层次结构（包括弹性规划、跨区域复制、在组织中共享 CA 等）的更多信息，请参阅 规划 AWS Private CA 部署。

实施步骤

1. 确定您的使用场景所需的相关 AWS 服务：
   - 许多使用场景都可以利用现有的 AWS 公钥基础设施并使用 AWS Certificate Manager。ACM 可用于为 Web 服务器、负载均衡器或公共可信证书的其他用途部署 TLS 证书。
   - 在您需要建立自己的私有证书颁发机构层次结构或需要使用可导出证书时，请考虑 AWS Private CA 。然后，可以使用 ACM 颁发 多种类型的终端实体证书 （使用 AWS Private CA）。
   - 对于必须为嵌入式物联网（IoT）设备大规模预置证书的使用场景，请考虑使用 AWS IoT Core。
2. 尽可能实施自动证书续订：
   - 将 ACM 托管续订 用于 ACM 颁发的证书以及集成的 AWS 托管服务。
3. 建立日志记录和审计跟踪：
   - 启用 CloudTrail 日志 ，以便跟踪对具有证书颁发机构的账户的访问。请考虑在 CloudTrail 中配置日志文件完整性验证，用于验证日志数据的真实性。
   - 定期生成和审查 审计报告 ，列出您的私有 CA 已颁发或撤销的证书。这些报告可以导出到 S3 存储桶。
   - 部署私有 CA 时，您还需要创建一个 S3 存储桶，用于存储证书撤销列表（CRL，Certificate Revocation List）。有关根据工作负载要求配置此 S3 存储桶的指南，请参阅 计划证书撤销列表 (CRL)。

资源

相关最佳实践：

- SEC02-BP02 使用临时凭证
- SEC08-BP01 实施安全密钥管理
- SEC09-BP04 对网络通信进行身份验证

相关文档：

- How to host and manage an entire private certificate infrastructure in AWS
- How to secure an enterprise scale ACM Private CA hierarchy for automotive and manufacturing
- Private CA best practices

- How to use AWS RAM to share your ACM Private CA cross-account

相关视频：

- Activating AWS Certificate Manager Private CA（研讨会）

相关示例：

- Private CA workshop
- 物联网设备管理研讨会（包括设备预置）

相关工具：

- 使用 AWS Private CA 的 Kubernetes 证书管理器插件

SEC09-BP02 在传输中执行加密

根据贵组织的政策、监管义务和标准定义，实施加密要求，以帮助满足组织、法律和合规性要求。如要在虚拟私有云（VPC）外部传输敏感数据，务必仅使用具有加密功能的协议。即使数据在不可信的网络中传输，加密也有助于保持数据的机密性。

期望结果：所有数据在传输过程中都应使用安全的 TLS 协议和密码套件进行加密。必须对资源和互联网之间的网络流量进行加密，以减少对数据的未经授权访问。对于仅在您内部 AWS 环境中的网络流量，应尽可能使用 TLS 进行加密。默认情况下，AWS 内部网络进行了加密，除非未经授权的一方能够访问正在生成流量的任何资源（如 Amazon EC2 实例和 Amazon ECS 容器），否则无法假冒或嗅探 VPC 内的网络流量。不妨考虑使用 IPsec 虚拟专用网络（VPN）保护网络到网络流量。

常见反模式：

- 使用已弃用的 SSL、TLS 和密码套件组件版本（例如，SSL v3.0、1024 位 RSA 密钥和 RC4 密码）。
- 允许未加密的（HTTP）流量进出面向公众的资源。
- 未在 X.509 证书到期前监控和替换证书。
- 对 TLS 使用自签名 X.509 证书。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

AWS 服务提供使用 TLS 的 HTTPS 端点进行通信，从而可以在与 AWS API 通信时提供传输中加密。可以使用安全组在 VPC 中审计和拦截不安全的协议，例如 HTTP。也可以在 Amazon CloudFront 中或 Application Load Balancer 上，将 HTTP 请求自动重定向到 HTTPS。您可以完全控制计算资源，以便在整个服务中实施加密。您也可以利用 VPN 连接从外部网络或 AWS Direct Connect 连接到您的 VPC 中，以便于对流量进行加密。请验证您的客户端至少使用 TLS 1.2 调用 AWS API，因为 AWS 将在 2023 年 6 月弃用 TLS 1.0 和 1.1。如果您有特殊要求，可以使用 AWS Marketplace 中提供的第三方解决方案。

实施步骤

- 实施传输中加密：您定义的加密要求应基于最新的标准和最佳实践，且仅允许使用安全协议。例如，配置一个安全组，仅允许通过 HTTPS 协议访问应用程序负载均衡器或 Amazon EC2 实例。
- 在边缘服务中配置安全协议：使用 Amazon CloudFront 配置 HTTPS，并使用适合您的安全状况和使用案例的安全配置文件。
- 将 VPN 用于外部连接：考虑使用 IPsec VPN 来保护点对点或网络对网络连接，以帮助实现数据隐私和完整性。
- 在负载均衡器中配置安全协议：选择一个安全策略，该策略提供受客户端支持且将要连接到侦听器的强大密码套件。为 Application Load Balancer 创建 HTTPS 侦听器。
- 在 Amazon Redshift 中配置安全协议：将集群配置为要求安全套接字层（SSL）或传输层安全性协议（TLS）连接。
- 配置安全协议：查看 AWS 服务文档，以确定传输中加密功能。
- 上传到 Amazon S3 存储桶时配置安全访问：使用 Amazon S3 存储桶策略控制措施执行对数据的安全访问。
- 不妨考虑使用 AWS Certificate Manager：ACM 允许您预置、管理和部署用于 AWS 服务的公有 TLS 证书。
- 不妨考虑使用 AWS Private Certificate Authority 满足私有 PKI 需求：AWS Private CA 允许您创建私有证书颁发机构（CA）层次结构，以签发可用于创建加密 TLS 通道的终端实体 X.509 证书。

资源

相关文档：

- AWS 文档
- 将 HTTPS 与 CloudFront 搭配使用

- 使用 AWS Virtual Private Network 将 VPC 连接到远程网络
- 为 Application Load Balancer 创建 HTTPS 侦听器
- 教程：在 Amazon Linux 2 上配置 SSL/TLS
- 使用 SSL/TLS 加密与数据库实例的连接
- 针对连接配置安全选项

SEC09-BP03 自动检测意外数据访问

使用 Amazon GuardDuty 等工具自动检测可疑活动或尝试将数据移动到定义的边界之外。例如，GuardDuty 可以通过以下方法，检测异常的 Amazon Simple Storage Service（Amazon S3）读取活动： Exfiltration:S3/AnomalousBehavior finding。除了 GuardDuty 以外，还可以将Amazon VPC 流日志（用于捕获网络流量信息）与 Amazon EventBridge 配合使用，以触发对已成功和被拒绝的异常连接的检测。Amazon S3 Access Analyzer 可以帮助评估您的 Amazon S3 存储桶中的哪些数据可供哪些人访问。

未建立此最佳实践暴露的风险等级： 中

实施指导

- 自动检测意外数据访问：使用工具或检测机制自动检测试图将数据移出定义边界的行为；例如，检测正在将数据复制到无法识别的主机的数据库系统。
  - VPC 流日志
- 考虑 Amazon Macie：Amazon Macie 是一项完全托管式数据安全和数据隐私服务，该服务使用机器学习和模式匹配发现和保护 AWS 中的敏感数据。
  - Amazon Macie

资源

相关文档：

- VPC 流日志
- Amazon Macie

SEC09-BP04 对网络通信进行身份验证

使用传输层安全性 (TLS) 或 IPsec 等支持身份验证的协议来验证通信的身份。

将您的工作负载设计为在服务和应用程序之间通信或与用户通信时，使用经过身份验证的安全网络协议。使用支持身份验证和授权的网络协议，可以加强对网络流量的控制，并减少未授权访问的影响。

期望结果：工作负载具有明确定义的数据面板和控制面板，它们控制流量在服务之间的流动。在技术上可行的情况下，流量将使用经过身份验证和加密的网络协议。

常见反面模式：

- 工作负载中存在未加密或未经身份验证的流量。
- 在多个用户或实体之间重用身份验证凭证。
- 仅依赖网络控制作为访问控制机制。
- 创建自定义身份验证机制，而不是依赖行业通用身份验证机制。
- VPC 中的服务组件或其他资源之间有过于宽松的流量流动。

建立此最佳实践的好处：

- 限制未经授权访问工作负载某一部分的影响范围。
- 提供更高级别的保障，即操作只能由经过身份验证的实体执行。
- 通过明确定义和强制执行预期的数据传输接口，改善服务的解耦。
- 通过请求归因和明确定义的通信界面，增强监控、日志记录和事件响应。
- 通过将网络控制与身份验证和授权控制相结合，为您的工作负载提供深度防御。

在未建立这种最佳实践的情况下暴露的风险等级：低

实施指导

您的工作负载的网络流量模式可分为两类：

- 东西向流量 代表构成工作负载的服务之间的流量。
- 南北向流量 代表您的工作负载和使用器之间的流量。

对南北向流量进行加密是常见做法，而使用经过身份验证的协议保护东西向流量则不太常见。现代安全实践建议，仅靠网络设计并不足以在两个实体之间建立可信关系。当两项服务可能位于公共网络边界内时，最佳做法仍然是对这些服务之间的通信进行加密、身份验证和授权。

例如，无论请求来自哪个网络，AWS 服务 API 都使用 AWS 签名版本 4（SigV4）签名协议对调用方进行身份验证。这种身份验证可确保 AWS API 可以验证请求操作的身份，然后可以将该身份与策略结合起来，作出授权决策，以确定是否应该允许该操作。

Amazon VPC Lattice 和 Amazon API Gateway 等服务允许您使用相同的 SigV4 签名协议，为自己的工作负载中的东西向流量添加身份验证和授权。如果您的 AWS 环境之外的资源要与服务进行通信，而服务需要基于 Sigv4 的身份验证和授权，则您可以对非 AWS 资源使用 AWS Identity and Access Management（IAM）Roles Anywhere 来获取临时 AWS 凭证。这些凭证可用于对使用 SigV4 的服务请求进行签名，以授权访问权限。

验证东西向流量的另一种常见机制是 TLS 双向身份验证（mTLS）。许多物联网（IoT）、企业对企业应用程序和微服务都使用 mTLS，通过使用客户端和服务器端 X.509 证书来验证 TLS 通信两端的身份。这些证书可以由 AWS Private Certificate Authority（AWS Private CA）颁发。您可以使用 Amazon API Gateway 和 AWS App Mesh 等服务，针对工作负载间或工作负载内的通信提供 mTLS 身份验证。虽然 mTLS 为 TLS 通信的两端提供身份验证信息，但它不提供授权机制。

最后，OAuth 2.0 和 OpenID Connect（OIDC）这两种协议通常用于控制用户对服务的访问，但如今在服务间流量中也变得越来越流行。API Gateway 提供了 JSON Web 令牌（JWT）授权方，允许工作负载使用 OIDC 或 OAuth 2.0 身份提供商颁发的 JWT 来限制对 API 路由的访问。可依据 OAuth2 范围来作出基本授权决策，但授权检查仍需要在应用层实现，仅靠 OAuth2 范围无法支持更复杂的授权需求。

实施步骤

- 定义并记录您的工作负载网络流：实施深度防御策略的第一步是定义工作负载的流量。
  - 创建数据流示意图，明确定义构成工作负载的不同服务之间如何传输数据。此示意图是强制这些流量通过经身份验证的网络渠道传输的第一步。
  - 在开发和测试阶段对您的工作负载进行检测，以验证数据流示意图是否准确反映了工作负载在运行时的行为。
  - 在执行威胁建模练习时，数据流示意图可能也很有用，如 SEC01-BP07 使用威胁模型识别威胁并确定缓解措施的优先级中所述。
- 建立网络控制：考虑使用 AWS 功能建立起与数据流相符的网络控制。虽然网络边界不应该是唯一的安全控制措施，但它们在深度防御策略中提供了一个安全层来保护您的工作负载。
  - 使用安全组来建立、定义和限制资源之间的数据流。
  - 考虑使用 AWS PrivateLink 与 AWS 以及支持 AWS PrivateLink 的第三方服务通信。通过 AWS PrivateLink 接口端点发送的数据保留在 AWS 网络主干内，而不通过公共互联网传输。
- 在工作负载中实施跨服务的身份验证和授权：选择最适合在工作负载中提供经过身份验证的加密流量的一组 AWS 服务。

  - 考虑使用 Amazon VPC Lattice 来保护服务间的通信。VPC Lattice 可以结合使用 Sigv4 身份验证与身份验证策略来控制服务间的访问。
  - 对于使用 mTLS 进行的服务间通信，请考虑使用 API Gateway 或 App Mesh。AWS Private CA 可用于建立能够颁发与 mTLS 结合使用的证书的私有 CA 层次结构。
  - 与使用 OAuth 2.0 或 OIDC 的服务集成时，可以考虑使用 JWT 授权方的 API Gateway。
  - 对于工作负载和物联网设备之间的通信，可以考虑使用 AWS IoT Core，它提供多种网络流量加密和身份验证选项。
- 监控未经授权的访问：持续监控非预期的通信渠道、试图访问受保护资源的未授权主体以及其他不当访问模式。
  - 如果使用 VPC Lattice 来管理对服务的访问，请考虑启用和监控 VPC Lattice 访问日志。这些访问日志包括有关请求实体的信息、源和目的地 VPC 等网络信息以及请求元数据。
  - 考虑启用 VPC 流日志，以捕获网络流量的元数据并定期检查是否存在异常。
  - 有关规划、模拟和响应安全事件的更多指导，请参阅 AWS 安全事件响应指南 和 AWS Well-Architected Framework 安全性支柱的“事件响应”部分。

资源

相关最佳实践：

- SEC03-BP07 分析公共和跨账户访问
- SEC02-BP02 使用临时凭证
- SEC01-BP07 使用威胁模型识别威胁并确定缓解措施的优先级

相关文档：

- 评估用于保护 Amazon API Gateway API 的访问控制方法
- 为 REST API 配置双向 TLS 身份验证
- 如何使用 JWT 授权方保护 API Gateway HTTP 端点
- 使用 AWS IoT Core 凭证提供程序授权直接调用 AWS 服务
- AWS 安全事件响应指南

相关视频：

- AWS re: invent 2022：VPC Lattice 简介

- AWS re: invent 2020：AWS 上 HTTP API 的无服务器 API 身份验证

相关示例：

- Amazon VPC Lattice 研讨会
- 零信任第 1 集 — 幻影服务边界研讨会

事件响应

即使采用成熟的预防和检测性控制措施，您的组织也应实施机制来响应安全事件并缓解安全事件可能带来的影响。您的准备工作会极大地影响团队在事件发生期间采取有效行动、对问题进行隔离、遏制和取证并将运行状态恢复到已知良好状态的能力。在安全事件发生之前确保相关工具和访问权限部署到位，然后通过实际试用定期进行事件响应演练，这样有助于确保您有能力恢复并最大限度避免业务中断。

主题

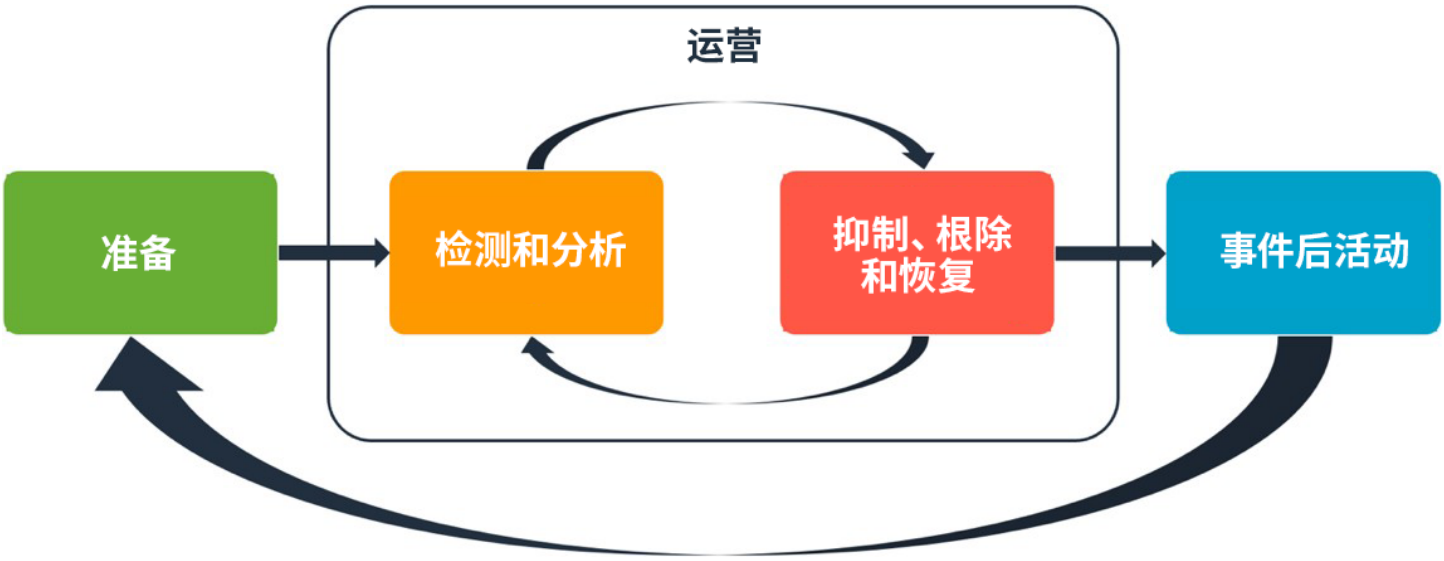
AWS 事件响应的各个方面

组织内的所有 AWS 用户都应对安全事件响应流程有基本的了解，并且安全人员应了解如何响应安全问题。教育、培训和经验对于成功的云事件响应计划至关重要，最好在处理可能发生的安全事件之前提前实施。云端成功的事件响应计划的基础是 准备、 运营和 事件后活动访问 AWS 资源。

要了解其中的各个方面，请考虑以下描述：

- 准备：让您的事件响应团队做好准备，以便在 AWS 中检测和响应事件，方法是启用检测控件，并确保对必要的工具和云服务拥有适当的访问权限。此外，还应通过人工和自动化的方式准备必要的行动手册，以确保可靠且一致的响应。
- 运营：按照 NIST 的事件响应阶段对安全事件和潜在事件采取行动：检测、分析、控制、根除和恢复。
- 事件后活动：对安全事件和模拟的结果进行迭代，以提高响应的有效性，从响应和调查中获得更多价值，并进一步降低风险。您必须从事件中吸取教训，并对改进活动有很大的主导权。

下图显示了这些方面的流程，与前面提到的 NIST 事件响应生命周期一致，但操作包括检测、分析、控制、根除和恢复。



AWS 事件响应的各个方面

云响应的设计目标

尽管事件响应的一般流程和机制（例如《NIST SP 800-61 计算机安全事件处理指南》中定义的那些流程和机制）依然有效，但我们鼓励您评估这些与云环境中的安全事件响应相关的特定设计目标：

- 建立响应目标： 与利益相关方、法律顾问和组织领导合作，以确定事件响应目标。一些共同的目标包括控制和缓解问题、恢复受影响的资源、保留数据为便取证、恢复到已知安全的操作，以及最终从事件中吸取教训。
- 利用云进行响应： 在云端（即事件和数据的发生地）实施响应模式。
- 了解您拥有和需要的证据： 通过复制日志、资源、快照和其他证据并将其存储在一个集中的响应专用云账户中，来保存这些内容。使用标签、元数据和保留策略实施机制。您需要了解自己使用了哪些服务，然后确定调查这些服务的要求。为了便于您了解自己的环境，您还可以使用标记。
- 使用重新部署机制： 如果安全异常可归因于某个配置错误，那么可能只需使用适当的配置重新部署资源以删除差异，即可完成修复。如果发现可能存在漏洞，请核实您重新部署时是否包括成功且经过验证的根本原因缓解措施。
- 尽可能自动化： 当问题出现或事件重复发生时，建立机制，以程序化方式对常见事件进行分类和响应。对于自动化程度不足的独特、复杂或敏感事件，使用人工响应。
- 选择可扩展的解决方案： 尽量让您的组织采用的方法的可扩展性与云计算能力相匹配。实施可在您的环境中扩展的检测和响应机制，以有效地缩短检测与响应之间的时间差。
- 了解并改进您的流程： 主动找出流程、工具或人员差距，并实施计划来修复这些差距。模拟是找到差距和改进流程的妥善方法。

这些设计目标会提醒您审查您的架构实施情况，以确定是否同时具备事件响应和威胁检测能力。在规划云端实施时，应考虑如何应对事件，最好使用具备司法有效性的响应方法。在某些情况下，这意味着您可能需要专门为这些响应任务设置多个组织、账户和工具。这些工具和职能应通过部署管道提供给事件响应者。它们不应该是静态的，因为这会导致更大的风险。

准备

要想及时、有效地应对事件，为事件做好准备至关重要。准备工作涉及三个领域：

- 人员：要让员工做好应对安全事件的准备，需要确定事件响应的利益相关方，并对他们进行事件响应和云技术方面的培训。
- 流程：为安全事件做好流程准备，包括记录架构、制定全面的事件响应计划，以及创建行动手册，以便对安全事件做出一致响应。
- 技术：为安全事件做好技术准备，包括设置访问权限、汇总和监控必要的日志、实施有效的警报机制，以及培养响应和调查能力。

这些领域中的每一个对于有效的事件响应都同等重要。没有这三项，任何事件响应计划都不完整或无效。您需要在人员、流程和技术方面做好准备，并将其紧密集成，以便为应对事件做好准备。

最佳实践

- SEC10-BP01 确定关键人员和外部资源
- SEC10-BP02 制定事件管理计划
- SEC10-BP03 准备取证能力
- SEC10-BP04 制定和测试安全事件响应行动手册
- SEC10-BP05 预置访问权限
- SEC10-BP06 预部署工具
- SEC10-BP07 运行模拟

SEC10-BP01 确定关键人员和外部资源

确定能够帮助您的组织响应事件的内部和外部人员、资源、以及法律义务。

当您与其他团队（例如法律顾问、领导、业务利益相关者、AWS Support 服务等）一起在云中定义您的事件响应方法时，您必须确定关键人员、利益相关者和相关联系人。为了降低依赖性并缩短响应时

间，请确保为您的团队、专家安全团队和响应者开展培训，以使他们了解您使用的服务并有机会练习动手实践。

我们鼓励您寻找外部 AWS 安全合作伙伴，他们应当能够为您带来外部专业知识和不同的视角，以增强您的响应能力。您的可靠安全合作伙伴可以帮助您发现您可能并不熟悉的潜在风险或威胁。

未建立这种最佳实践的情况下暴露的风险等级： 高

实施指导

- 确定组织中的关键人员： 制定联系人列表，列出组织内需要参与事件响应和事件恢复的人员。
- 确定外部合作伙伴： 根据需要联系能够帮助您响应事件并从事件中恢复的外部合作伙伴。

资源

相关文档：

- AWS 事件响应指南

相关视频：

- 准备和响应 AWS 环境中的安全事件

相关示例：

SEC10-BP02 制定事件管理计划

为事件响应制定的第一个文档是事件响应计划。事件响应计划旨在为您的事件响应计划和战略奠定基础。

建立此最佳实践的好处： 要想成功实现可扩展的事件响应计划，制定全面且明确定义的事件响应流程是关键。在发生安全事件时，明确的步骤和工作流有助于您及时做出响应。您可能已经有事故响应流程。无论您当前的状态如何，定期更新、迭代和测试事件响应流程都很重要。

未建立这种最佳实践的情况下暴露的风险等级： 高

实施指导

对于响应、缓解安全事件的潜在影响并从中恢复来说，事件管理计划是至关重要的。事件管理计划是一个结构化的过程，用于及时地确定、补救和响应安全事件。

云的许多操作角色和要求都与本地环境中的相同。在创建事件管理计划时，应考虑最符合业务成果和合规性要求的响应和恢复策略，这一点非常重要。例如，如果您在 AWS 中运行在美国符合 FedRAMP 标准的工作负载，则遵守《NIST SP 800-61 计算机安全处理指南》会很有帮助。同样，在使用欧洲个人身份信息（PII）数据运行工作负载时，请考虑具体的场景，例如如何根据以下条例的强制要求，保护和应对与数据驻留相关的问题：欧盟通用数据保护条例（GDPR）。

在为 AWS 中的工作负载制定事件管理计划时，请首先使用 AWS 责任共担模式，以便构建针对事件响应的深度防御方法。在此模式中，AWS 负责管理云本身的安全，云内部的安全则由您负责。这意味着您将保留控制权，并对选择实施的安全控制机制负责。此 AWS 安全事件响应指南 详细介绍了构建以云为中心的事件管理计划的关键概念和基本指南。

必须不断地迭代有效的事件管理计划，使其与您的云运营目标保持一致。在创建和改进事件管理计划时，请考虑使用下面详述的实施计划。

实施步骤

定义角色和职责

处理安全事件需要在整个组织中落实纪律要求和行动意愿。在您的组织结构中，发生事件时，负责、追责、咨询或者告知信息等各个环节会涉及到不同的人员，例如人力资源（HR）、高管团队和法律部门的代表。请考虑这些角色和职责，以及是否必须有第三方参与。请注意，许多地区的当地法律都规定了，哪些事情能做，哪些事情不能做。尽管为安全响应计划建立一个负责、问责、咨询和知情（RACI）的图表可能显得过于繁文缛节，但这样做有利于快速直接地进行沟通，并清楚地概述在事件不同阶段负责的领导层。

在事件发生期间，让受影响应用程序和资源的负责人和开发人员参与进来非常关键，因为这些人员是主题专家（SME），可以提供信息和背景情况来协助衡量影响。您应该先练习并与开发人员和应用程序负责人建立关系，然后才能依靠他们的专业知识进行事件响应。应用程序负责人或 SME，如云管理员或工程师，可能需要在不熟悉环境、面临复杂情况或响应人员没有访问权限的情况下采取行动。

最后，值得信赖的合作伙伴可以参与到调查或响应中，因为他们可以提供额外的专业知识和宝贵的审查工作。当您自己的团队缺乏具备这些技能的人员时，您可能需要聘请外部人员寻求帮助。

了解 AWS 响应团队和支持

- AWS Support
  - AWS Support 提供了一系列计划，可供您利用各种工具和专业知识，为您的 AWS 解决方案的成功和正常运行提供支持。如果您需要技术支持及更多资源来规划、部署和优化 AWS 环境，则可以选择最符合您 AWS 使用场景的支持计划。

  - 考虑将 支持中心 （在 AWS Management Console中，需要登录）作为中心联系点，为影响您 AWS 资源的问题获取支持。对 AWS Support 的访问由 AWS Identity and Access Management 控制。有关获取对 AWS Support 功能的访问的更多信息，请参阅 开始使用 AWS Support。
- AWS 客户事件响应团队（CIRT）
  - AWS 客户事件响应团队（CIRT）是一支专业的 AWS 全球团队，全天候向客户提供支持，协助客户解决根据 AWS 责任共担模式应由客户一方负责的安全事件。
  - 当 AWS CIRT 为您提供支持时，他们会为 AWS 上出现的安全事件提供分类和恢复方面的协助。他们可以使用 AWS 服务日志来协助分析根本原因，并为您提供恢复建议。他们还可以提供安全建议和最佳实践，从而让您以后能够避免出现安全事件。
  - AWS 客户要与 AWS CIRT 交流，可以开立 AWS Support 案例。
- DDoS 响应支持
  - AWS 提供 AWS Shield，它提供了托管的分布式拒绝服务（DDoS）攻击保护服务，可保护在 AWS 上运行的 Web 应用程序。Shield 提供不间断检测和自动化内嵌缓解措施，可以最大限度地减少应用程序停机时间和延迟，因此无需与 AWS Support 交流即可从 DDoS 保护中受益。Shield 分为两个级别：AWS Shield Standard 和 AWS Shield Advanced。要了解这两个级别之间的区别，请参阅 Shield 功能文档。
- AWS Managed Services（AMS）
  - AWS Managed Services（AMS） 可持续管理您的 AWS 基础设施，让您可以专注于应用程序。AMS 实施最佳实践来维护您的基础设施，让您您能够降低运营开销和风险。AMS 可以自动执行常见活动（例如更改请求、监控、补丁管理、安全性和备份服务），并可以提供全生命周期服务来预置、运行和支持您的基础设施。
  - AMS 负责部署一套安全检测控制措施，并全天候提供对警报的第一线响应。启动警报后，AMS 遵循一组标准的自动和手动行动手册，验证是否有一致的响应。这些行动手册在功能部署期间与 AMS 客户共享，这样客户就能够开发并与 AMS 协调响应措施。

制定事件响应计划

事件响应计划旨在为您的事件响应计划和战略奠定基础。事件响应计划应包含在正式文档中。事件响应计划通常包括以下部分：

- 事件响应团队概述： 概述事件响应团队的目标和职能。
- 角色和职责： 列出事件响应利益相关者，并详细说明他们在发生事件时的角色。
- 沟通计划： 详细的联系信息，以及在事件发生期间如何进行沟通。

- 后备沟通方法： 此时的最佳实践是采用带外通信，作为事件沟通的后备。AWS Wickr 就是一个提供安全的带外通信渠道的应用程序示例。
- 事件响应阶段和应采取的行动： 列举事件响应的各个阶段（例如，检测、分析、消除、遏制和恢复），包括在这些阶段中要采取的高级别操作。
- 事件严重性和优先级定义： 详细说明如何对事件的严重性进行分类，如何确定事件的优先级，然后详细说明严重性定义对上报程序有何影响。

尽管这些内容部分在各种规模和行业的公司中很常见，但每个组织的事件响应计划都是独一无二的。您需要制定最适合贵组织的事件响应计划。

资源

相关最佳实践：

- SEC04（“您如何检测和调查安全事件？”）

相关文档：

- AWS 安全事件响应指南
- NIST：计算机安全事件处理指南

SEC10-BP03 准备取证能力

在发生安全事件之前，可以考虑构建取证能力来支持安全事件调查工作。

未建立这种最佳实践的情况下暴露的风险等级： 中

传统本地取证的概念适用于 AWS。有关开始在 AWS Cloud中构建取证功能的关键信息，请参阅 AWS Cloud中的取证调查环境策略。

设置好取证的环境和 AWS 账户结构后，确定在以下四个阶段有效执行可靠取证方法所需的技术：

- 收集： 收集相关的 AWS 日志，例如 AWS CloudTrail、AWS Config、VPC 流日志和主机级日志。收集受影响的 AWS 资源的快照、备份和内存转储（如果有）。
- 检查： 通过提取和评测相关信息来检查收集到的数据。
- 分析： 分析收集到的数据，以便了解事件并从中得出结论。
- 报告： 提供分析阶段得出的信息。

实施步骤

准备取证环境

AWS Organizations 有助于您随着 AWS 资源的增长和扩展，集中管理和监管 AWS 环境。AWS 组织会整合您的 AWS 账户，这样您就能够将这些账户作为一个单元进行管理。可以使用组织单位（OU）将账户分组到一起，以便作为一个单元进行管理。

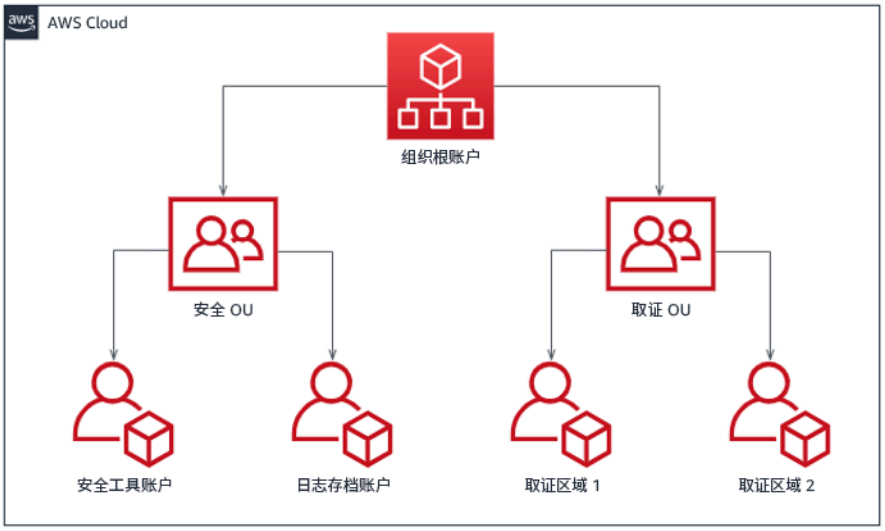
对于事件响应，拥有一个支持事件响应功能的 AWS 账户结构（包括 安全 OU 和 取证 OU）会很有帮助。在安全 OU 中，您应该拥有以下账户：

- 日志存档： 将日志聚合到权限有限的日志存档 AWS 账户中。
- 安全工具： 将安全服务集中在安全工具 AWS 账户中。此账户以安全服务的委托管理员身份运行。

在取证 OU 中，您可以选择实施单一取证账户，也可以为您运营的每个区域实施账户，具体取决于哪种账户最适合您的业务和运营模式。如果为每个区域创建一个取证账户，就可以阻止在该区域之外创建 AWS 资源，降低资源被复制到非预期区域的风险。例如，如果您只在US East (N. Virginia) Region（`us-east-1`）和 US West (Oregon)（`us-west-2`）运营，那么您将在取证 OU 中拥有两个账户：一个用于 `us-east-1` ，另一个用于 `us-west-2`。

可以为多个区域创建取证 AWS 账户。在将 AWS 资源复制到该账户时应小心谨慎，确保符合数据主权要求。由于预置新账户需要时间，因此必须在事件发生前创建和分析取证账户，以便响应人员能够做好准备，有效地使用这些账户进行响应。

下图显示了一个账户结构示例，其中包括一个取证 OU，涵盖了根据每个区域创建的取证账户：



用于响应事件而根据区域创建的账户结构

捕获备份和快照

为关键系统和数据库建立备份对于从安全事件中恢复和取证至关重要。有了备份，您就能够将系统恢复到以前的安全状态。在 AWS 上，您可以创建各种资源的快照。快照为您提供这些资源的时间点备份。有许多 AWS 服务能够在备份和恢复方面为您提供支持。有关这些服务以及备份和恢复方法的详细信息，请参阅 备份和恢复规范性指南 以及 使用备份从安全事件中恢复。

特别是遇到勒索软件等情况时，妥善保护备份至关重要。有关保护备份的指导，请参阅 AWS 中保护备份的 10 大安全最佳实践。除了确保备份安全外，您还应当定期测试备份和还原流程，从而确保现有的技术和流程按预期运行。

自动取证

在安全事件期间，您的事件响应团队必须能够快速收集和分析证据，同时保持事件相关时间段的准确性（例如捕获与特定事件或资源相关的日志，或收集 Amazon EC2 实例的内存转储）。对于事件响应团队来说，手动收集相关证据既具有挑战性又很耗时，尤其是在存在大量实例和账户的情况下。此外，手动收集容易出现人为错误。出于这些原因，您应该尽可能开发和实现取证自动化功能。

AWS 提供了大量用于取证的自动化资源，这些资源在下面的“资源”部分中列出。这些资源是我们开发并由客户实施的取证模式示例。虽然这些资源可能是有用的参考架构，但可以考虑根据您的环境、要求、工具和取证流程对资源进行修改，或者创建新的取证自动化模式。

资源

相关文档：

- AWS 安全事件响应指南 – 构建取证能力
- AWS 安全事件响应指南 – 取证资源
- AWS Cloud中的取证调查环境策略
- 如何在 AWS 中自动实施取证磁盘收集
- AWS Prescriptive Guidance – 自动化事件响应和取证

相关视频：

- 自动化事件响应和取证

相关示例：

- 自动事件响应和取证框架
- Amazon EC2 的自动取证编排工具

SEC10-BP04 制定和测试安全事件响应行动手册

准备事件响应流程的关键环节是制定行动手册。事件响应行动手册提供了一系列规范性指导和步骤，供发生安全事件时遵循。清晰的结构和步骤可简化响应，减少发生人为错误的可能性。

未建立这种最佳实践的情况下暴露的风险等级： 中

实施指导

应针对以下事件场景创建行动手册：

- 预期事件：应针对预期的事件创建行动手册。这包括拒绝服务（DoS）、勒索软件和凭证泄露等威胁。
- 已知的安全调查发现或警报：应针对已知的安全调查发现和警报（如 GuardDuty 调查发现）创建行动手册。您可能会收到一个 GuardDuty 调查发现，然后想："现在怎么办？" 为防止错误处理或忽略 GuardDuty 调查发现，应针对每个可能的 GuardDuty 调查发现创建行动手册。有关补救细节和指导的信息可在 GuardDuty 文档中找到。需要注意的是，默认情况下并不会启用 GuardDuty，而且需要付费。有关 GuardDuty 的详细信息，请参阅 附录 A：云功能定义 – 可见性和警报。

行动手册应包含安全分析师需要完成的技术步骤，以便充分调查和应对潜在的安全事件。

实施步骤

行动手册中应包括的项目有：

- 行动手册概述：本行动手册针对哪些风险或事件场景？ 本行动手册的目标是什么？
- 先决条件：此事件场景需要哪些日志、检测机制和自动化工具？ 预期的通知是什么？
- 沟通和上报信息：谁参与其中，他们的联系信息是什么？ 每个利益相关者的责任是什么？
- 响应步骤：在事件响应的各个阶段，应采取哪些战术性措施？ 分析师应该进行哪些查询？ 应该运行什么代码才能达到预期的结果？
  - 检测：如何检测事件？
  - 分析：如何确定影响范围？
  - 控制：如何隔离事件来限制其影响范围？

  - 消除：如何从环境中消除威胁？
  - 恢复：受影响的系统或资源将如何恢复生产？
- 预期结果：运行查询和代码后，行动手册的预期结果是什么？

资源

相关的 Well-Architected 最佳实践：

- SEC10-BP02 – 制定事件管理计划

相关文档：

- 事件响应行动手册框架
- 制定自己的事件响应行动手册
- 事件响应行动手册样本
- 使用 Jupyter 行动手册和 CloudTrail Lake 构建 AWS 事件响应运行手册

SEC10-BP05 预置访问权限

确保事件响应者将正确的访问权限预置到 AWS 中，以缩短调查到恢复所需的时间。

常见反模式：

- 使用根账户进行事件响应。
- 变更现有用户账户。
- 在提供实时权限提升时直接操作 IAM 权限。

未建立这种最佳实践的情况下暴露的风险等级： 中

实施指导

AWS 建议尽可能减少或消除对长期有效凭证的依赖，转而使用临时凭证和 实时 权限提升机制。长期有效的凭证容易带来安全风险，并且会增加运营开销。对于大多数管理任务以及事件响应任务，建议您对管理访问实施 身份联合验证 以及 临时上报。在此模型中，用户请求提升到更高级别的权限（例如事

件响应角色），如果用户符合提升条件，则会向审批者发送请求。如果请求获得批准，用户将收到一组临时的 AWS 凭证 ，可用于完成用户任务。在这些凭证过期后，用户必须提交新的提升请求。

在大多数事件响应场景中，建议使用临时权限提升。执行此操作的正确方法是使用 AWS Security Token Service 和 会话策略 来限定访问范围。

在一些场景中，联合身份不可用，例如：

- 与被盗用的身份提供者（IdP）相关的中断。
- 导致联合访问管理系统损坏的错误配置或人为错误。
- 恶意活动，例如分布式拒绝服务（DDoS，Distributed Denial of Service）事件或导致系统不可用的活动。

在上述情况下，应配置紧急 Break Glass 访问，以允许调查事件并及时给予补救。我们建议您使用 具有适当权限的 IAM 用户， 来执行任务和访问 AWS 资源。请仅将根凭证用于 需要根用户访问权限的任务。要确认事件响应者对 AWS 和其他相关系统是否具有正确的访问权限级别，建议预置专用的用户账户。用户账户需要特许的访问权限，并且必须受到严格的控制和监视。在构建账户时，必须使用执行必要任务所需的最少权限，并且访问级别应基于作为事件管理计划的一部分创建的行动手册。

最好使用专门构建的专用用户和角色。通过添加 IAM 策略来临时提升用户或角色的访问权限，既会导致无法清楚地了解用户在事件期间拥有哪些访问权限，又会带来无法撤销提升的权限的风险。

请务必删除尽可能多的依赖项，以确保能在尽可能多的故障场景中获得访问权限。为了支持此操作，可创建一个行动手册，验证是否在专用的安全账户中创建事件响应用户作为 AWS Identity and Access Management 用户，而不是通过任何现有的联合身份验证或单点登录（SSO）解决方案管理他们。每个响应者都必须有自己的指定账户。账户配置必须实施 强密码策略 和多重身份验证（MFA）。如果事件响应行动手册仅需要对 AWS Management Console的访问权限，则用户不应配置访问密钥，并且应明确禁止用户创建访问密钥。可以使用 IAM 策略或服务控制策略（SCP，Service Control Policy）进行此配置，如 AWS 安全最佳实践（适用于 AWS Organizations SCP）中所述。用户仅能够在其他账户中代入事件响应角色，而不应具有其他任何权限。

在事件处理期间，可能需要向其他内部或外部人员授予访问权限，以支持调查、补救或恢复活动。在这种情况下，可以使用前面提到的行动手册机制，并且必须创建一个流程，确保在事件结束后立即撤消其他任何访问权限。

要确保能正确地监控和审计对事件响应角色的使用，至关重要的一点是，为此目的创建的 IAM 用户账户不会在人员之间共享，并且不会使用 AWS 账户 根用户，除非 特定任务要求这样做。如果需要根用户（例如，对特定账户的 IAM 访问权限不可用），请使用单独的流程和可用的行动手册来验证根用户密码和 MFA 令牌的可用性。

要为事件响应角色配置 IAM 策略，请考虑使用 IAM Access Analyzer 来生成基于 AWS CloudTrail 日志的策略。为此，请在非生产账户中向事件响应角色授予管理员访问权限，并运行行动手册。完成后，会创建一个策略，仅允许已执行的操作。之后，可以跨所有账户将此策略应用于所有事件响应角色。您可能希望为每个行动手册创建一个单独的 IAM 策略，以便更轻松地进行管理和审计。示例行动手册可能包括针对勒索软件、数据泄露、丢失生产访问权限和其他场景的响应计划。

使用事件响应用户账户可在 其他 AWS 账户 中代入专用的事件响应 IAM 角色。必须将这些角色配置为仅可由安全账户中的用户代入，并且信任关系必须要求调用主体已使用 MFA 进行身份验证。角色必须使用严格界定的 IAM 策略来控制访问。确保这些角色的所有 AssumeRole 请求都记录在 CloudTrail 中并发出提醒，并确保记录使用这些角色执行的任何操作。

强烈建议清楚地命名 IAM 用户账户和 IAM 角色，以便在 CloudTrail 日志中轻松找到他们。例如，将 IAM 账户命名为 *<USER_ID>*-BREAK-GLASS# 并将 IAM 角色命名为 BREAK-GLASS-ROLE。

CloudTrail 用于记录 AWS 账户中的 API 活动，并且应该用于 配置关于使用事件响应角色的提醒。。请参阅博文，了解有关配置使用根密钥时的提醒。可以修改说明以配置 Amazon CloudWatch 指标筛选条件，从而筛选 AssumeRole 事件（与事件响应 IAM 角色相关）：

```
{ $.eventName = "AssumeRole" && $.requestParameters.roleArn =
 "<INCIDENT_RESPONSE_ROLE_ARN>" && $.userIdentity.invokedBy NOT EXISTS && $.eventType !
= "AwsServiceEvent" }
```

由于事件响应角色可能具有高级别的访问权限，因此，请务必将这些提醒转至广泛的群体，并及时采取适当的行动。

在事件处理期间，响应者可能需要访问不受 IAM 直接保护的系统。它们可能包括 Amazon Elastic Compute Cloud 实例、Amazon Relational Database Service 数据库或软件即服务（SaaS）平台。强烈建议不要使用 SSH 或 RDP 等本机协议，而是使用AWS Systems Manager Session Manager 对 Amazon EC2 实例进行所有管理访问。可以使用安全且经过审计的 IAM 控制此访问。此外，还可以使用 AWS Systems Manager Run Command 文档自动实施行动手册的部分内容，这可以减少用户出错的机会并缩短恢复时间。对于访问数据库和第三方工具，我们建议将访问凭证存储在 AWS Secrets Manager 中，并向事件响应者角色授予访问权限。

最后，事件响应 IAM 用户账户的管理应该添加到您的 合并人员、移动人员和离开人员流程中，并定期进行检查和测试，以确认只允许预期访问。

资源

相关文档：

- 管理对 AWS 环境的临时提升的访问权限
- AWS 安全事件响应指南
- AWS Elastic Disaster Recovery
- AWS Systems Manager Incident Manager
- 为 IAM 用户设置账户密码策略
- 在 AWS 中使用多重身份验证（MFA）
- 使用 MFA 配置跨账户存取
- 使用 IAM Access Analyzer 生成 IAM 策略
- 多账户环境中的 AWS Organizations 服务控制策略的最佳实践
- 如何在使用 AWS 账户的根访问密钥时接收通知
- 使用 IAM 托管策略创建精细会话权限

相关视频：

- 在 AWS 中自动化事件响应和取证
- 运行手册、事件报告和事件响应 DIY 指南
- 准备和响应 AWS 环境中的安全事件

相关示例：

- 实验室：AWS 账户设置和根用户
- 实验：使用 AWS 控制台和 CLI 的事件响应

SEC10-BP06 预部署工具

确保安全人员预部署了适当的工具，来缩短从调查到恢复的时间。

未建立这种最佳实践的情况下暴露的风险等级： 中

实施指导

要自动执行安全响应和操作功能，您可以使用 AWS 提供的一整套 API 和工具。您可以完全自动执行身份管理、网络安全、数据保护和监控功能，并使用您已采用的常见软件开发方法交付这些功能。当构建

安全自动化时，您的系统可以监控、审核和启动响应，您不必安排人员监控您的安全位置并对事件做出人为响应。

如果您的事件响应团队继续以同样的方式响应警报，警报可能会让他们应接不暇。久而久之，团队对警报的敏感性可能会下降，并可能在处理正常情况时犯错或者错过异常警报。利用一些功能自动处理重复和正常的警报，并将敏感、特殊的事件交由人员来处理，这样有助于避免疲于应对警报。集成异常检测系统（例如 Amazon GuardDuty、AWS CloudTrail Insights 和 Amazon CloudWatch Anomaly Detection）可以减轻常见阈值警报的负担。

您可以通过编程方式自动执行此流程中的步骤，从而改进手动流程。为事件定义修复模式之后，您可以将此模式分解为可执行的逻辑，并编写代码以执行此逻辑。然后，响应人员可以运行该代码来修复问题。久而久之，您就可以自动化越来越多的步骤，并最终自动处理各类常见事件。

在安全调查期间，您需要能够查看相关日志，以便记录并了解事件的来龙去脉和时间线。生成警报时也需要日志，因为日志可以指示某些相关操作已经发生。选择、启用、存储、设置查询和检索机制以及设置警报至关重要。此外，提供工具来搜索日志数据的有效方法是 Amazon Detective。

AWS 提供 200 多种云服务和数千种功能。我们建议您检查可支持和简化事件响应策略的服务。

除日志记录外，还应当制定并实施 标记策略。标记有助于提供有关 AWS 资源用途的背景信息。标记也可用于实现自动化。

实施步骤

选择并设置用于分析和报警的日志

请参阅以下关于配置事件响应日志记录的文档：

- 安全事件响应的日志记录策略
- SEC04-BP01 配置服务和应用程序日志记录

启用安全服务来支持检测和响应

AWS 提供了本机检测、预防和响应功能，而其他服务可用于构建自定义安全解决方案。有关与安全事件响应最相关的服务列表，请参阅 云功能定义。

制定和实施标记策略

要获取围绕 AWS 资源的业务场景和相关内部利益相关者的背景信息，可能很困难。要做到这一点，可以采用标签的形式，标签为 AWS 资源分配元数据，并由用户定义的键和值组成。您可以创建标签，按照用途、所有者、环境、处理的数据类型以及您选择的其他标准对资源进行分类。

采用一致的标记策略可以加快响应速度，并通过快速识别和辨别 AWS 资源的背景信息，最大限度地减少在组织背景方面所花费的时间。标签还可以充当启动自动响应的机制。有关要标记的内容的详细信息，请参阅 标记 AWS 资源。首先，您需要定义要在组织内实施的标签。之后，实施并强制执行这一标记策略。有关实施和强制执行的详细信息，请参阅 使用 AWS 标签策略和服务控制策略（SCP）实施 AWS 资源标记策略。

资源

相关的 Well-Architected 最佳实践：

- SEC04-BP01 配置服务和应用程序日志记录
- SEC04-BP02 集中分析日志、结果和指标

相关文档：

- 安全事件响应的日志记录策略
- 事件响应云功能定义

相关示例：

- 使用 Amazon GuardDuty 和 Amazon Detective 进行威胁检测和响应
- Security Hub 研讨会
- 使用 Amazon Inspector 进行漏洞管理

SEC10-BP07 运行模拟

随着组织不断发展壮大，威胁形势也会不断变化，因此务必要持续评估组织的事件响应能力。运行模拟（也称为实际演练）是可用于执行这种评估的一种方法。模拟过程使用现实世界中的安全事件场景，旨在模仿威胁主体采取的战术、技术和程序（TTP），让组织通过响应现实中可能发生的模拟网络事件，来练习和评估自己的事件响应能力。

建立这种最佳实践的好处：模拟有多种好处：

- 检验网络准备情况，有助于事件响应人员树立信心。
- 测试工具和工作流程的准确性和有效性。
- 完善沟通和上报环节，使之与您的事件响应计划相吻合。
- 提供机会来应对不太常见的攻击载体。

在未建立这种最佳实践的情况下暴露的风险等级：中等

实施指导

模拟主要分为三种类型：

- 桌面演练：桌面演练模拟方法是一种基于讨论的会议，让各个事件响应利益相关者参与进来，练习角色和职责，以及练习使用既定的沟通工具和行动手册。通常是用一整天的时间在虚拟场地和/或实地中协调完成演练。由于桌面演练以讨论为基础，因此侧重于流程、人员和协作。在讨论中，技术是必不可少的一部分，但事件响应工具或脚本的实际使用通常不包括在桌面演练中。
- 紫队演练：紫队演练可提高事件响应人员（蓝队）和模拟威胁主体（红队）之间的协作能力。蓝队由安全运营中心（SOC）的成员组成，但也可以包括在实际网络事件中会参与进来的其他利益相关者。红队由渗透测试团队或接受过攻击安全培训的关键利益相关者组成。在设计场景时，红队会与演练协调员相互协作，以确保场景的准确性与可行性。在紫队演练中，主要的关注点是支持事件响应工作的检测机制、工具和标准操作程序（SOP）。
- 红队演练：在红队演练中，进攻方（红队）模拟进行攻击，从而在预定范围内实现特定目标或一系列目标。防御方（蓝队）不一定知道演练的范围和持续时间，如此，可以更真实地评估他们应对真实事件的能力。由于红队的演练可能是侵入性测试，因此务必谨慎行事，并实施控制措施，以确保该演练不会对环境造成实际破坏。

请考虑定期协调开展网络模拟。对于参与者和整个组织而言，每种演练类型都可以带来独特的好处，因此您可以选择从不太复杂的模拟类型（例如桌面演练）入手，然后再慢慢过渡到较为复杂的模拟类型（红队演练）。您应根据自身的安全成熟度、资源和预期结果选择模拟类型。由于红队演练的复杂性和成本，一些客户可能不会选择进行红队演练。

实施步骤

无论您选择哪种模拟类型，模拟通常都遵循以下实施步骤：

1. 确定核心演练要素：确定模拟场景和模拟要达成的目标。这两者都应该得到领导层的认同。
2. 确定关键利益相关者：演练至少需要演练协调员和参与者。根据具体的场景，可能会涉及其他利益相关者，例如法务、通信或行政等领域的领导层。
3. 构建和测试场景：如果有特定要素不可行，则可能需要在构建时重新定义该场景。本阶段的预期结果是最终确定的场景。
4. 协调开展模拟：采用的模拟类型决定了所需的协调工作（书面讨论场景对比高技术含量的模拟场景）。协调员应根据演练目标调整其协调战术，并应尽可能让所有演练参与者都参与进来，以实现最大利益。

5. 撰写事后报告（AAR）：确定哪些方面进展较为顺利、哪些方面需要改进以及可能存在的差距。AAR 应衡量模拟的有效性，并记录团队对模拟事件的响应情况，以便在将来的模拟中可以不断跟踪进度。

资源

相关文档：

- AWS 事件响应指南

相关视频：

- AWS 实际演练 – 安全版

运营

运营是执行事件响应的核心。这是响应和修复安全事件的操作发生的地方。运营包括以下五个阶段：检测、分析、 控制、根除和 恢复访问 AWS 资源。下表中提供了这些阶段和目标的描述。

阶段	目标
检测	识别潜在的安全事件。
分析	确定安全事件是否为意外事件，并评估事件的影响范围。
控制	尽量减小和限制安全事件的影响范围。
根除	移除与安全事件相关的未经授权的资源或构件。实施可消除安全事件的缓解措施。
恢复	将系统恢复到已知安全的状态并监控这些系统，以确认威胁不会再次出现。

在您应对和处理安全事件时，应使用这些阶段作为指导，以便有效且可靠地进行响应。您采取的实际操作将因事件而异。例如，涉及勒索软件的事件要遵循的响应步骤与涉及公共 Amazon S3 存储桶的事件

不同。此外，这些阶段不一定按顺序发生。在控制和根除之后，您可能需要重新分析，以了解您的操作是否有效。

在人员、流程和技术方面做好充分的准备是有效运营的关键。因此，请遵循 准备 部分中的最佳实践，以便能够有效地响应活动的安全事件。

如需了解更多信息，请参阅 运营 部分（在《AWS 安全事件响应指南》中）。

事件后活动

威胁形势在不断变化，您的组织必须具备同样的动态性，以有效保护您的环境。持续改进的关键在于对事件和模拟的结果进行迭代，以提高有效检测、响应和调查可能的安全事件的能力，减少潜在漏洞，缩短响应时间，并恢复安全运营。以下机制有助于验证您的组织是否已经准备就绪，可以利用最新的功能和知识有效应对任何情形。

最佳实践

- SEC10-BP08 建立从事件中吸取经验教训的框架

SEC10-BP08 建立从事件中吸取经验教训的框架

实现 经验教训总结 框架和根本原因分析能力不仅有助于提高事件响应能力，还有助于防止事件再次发生。通过从每次事件中吸取教训，您可以避免重复同样的错误、泄露或错误配置，这不仅可以改善您的安全态势，还可以最大限度地减少因可预防的情况而损失的时间。

未建立这种最佳实践的情况下暴露的风险等级： 中

实施指导

重要的是要实现一个 经验教训总结 框架，大体上确立并实现以下几点：

- 何时总结经验教训？
- 总结经验教训的过程涉及什么？
- 如何总结经验教训？
- 谁参与了这个过程，具体情况如何？
- 如何确定需要改进的领域？
- 如何确保有效跟踪和实施改进措施？

该框架不应关注或指责个人，而应侧重于改进工具和流程。

实施步骤

除了前面列出的大体上的成果外，重要的是要确保提出正确的问题，以便从流程中获得最大价值（可以带来切实可行的改进的信息）。请考虑以下问题，以便于您启动经验教训讨论：

- 发生了什么事件？
- 何时首次发现该事件？
- 是如何发现的？
- 哪些系统针对该活动发出了警报？
- 涉及哪些系统、服务和数据？
- 具体发生了什么？
- 哪些地方做得好？
- 哪些地方做得不好？
- 哪些流程或程序出现问题或未能扩展以应对事件？
- 以下方面有哪些地方有待改进：
  - 人员
    - 需要联系的人是否真的可以联系上，联系名单是否是最新名单？
    - 相应人员是否缺少有效应对和调查事件所需的培训或能力？
    - 相应的资源是否已就绪并随时可用？
  - 流程
    - 是否遵循了流程和程序？
    - 是否针对这种事件记录并提供了流程和程序？
    - 是否缺少必要的流程和程序？
    - 响应人员是否能够及时获得所需的信息来处理问题？
  - 技术
    - 现有警报系统是否能有效识别活动并发出警报？
    - 我们如何将检测时间缩短 50%？
    - 现有警报是否需要改进，或者是否需要针对这种事件设置新的警报？
    - 现有工具是否允许对事件进行有效调查（搜索/分析）？
    - 怎样才能更快地识别这种事件？
    - 如何防止这种事件再次发生？

- 谁是改进计划的负责人，如何检验改进计划的执行情况？
- 实施和测试额外监控或预防性控制和流程的时间表是怎样的？

此列表并非详尽无遗，但旨在作为一个起点，确定组织和业务需求是什么，以及如何分析这些需求，以便最有效地从事件中吸取经验教训，并不断改进您的安全态势。最重要的是，该列表开始将经验教训作为事件响应流程、文档和利益相关者期望的标准组成部分。

资源

相关文档：

- AWS 安全事件响应指南：建立从事件中吸取经验教训的框架
- NCSC CAF 指南：总结经验教训

应用程序安全性

应用程序安全性（AppSec）介绍了如何设计、构建和测试所开发工作负载的安全属性的整个过程。您的组织中应该有经过适当培训的人员，了解构建和发布基础设施的安全属性，并使用自动化来识别安全问题。

在软件开发生命周期（SDLC）和发布后流程的常规部分采用应用程序安全性测试，有助于确保您拥有一种结构化的机制来识别、修复和防止应用程序安全性问题进入生产环境。

在设计、构建、部署和操作工作负载时，应用程序开发方法应该包括安全控制机制。在此过程中，协调流程以持续减少缺陷并尽可能减少技术债务。例如，在设计阶段使用威胁建模有助于及早发现设计缺陷，这使得缺陷更易于修复，修复的成本更低，而不是等到以后再来缓解这些缺陷。

在 SDLC 中，越早的阶段，解决缺陷的成本和复杂性通常就会越低。解决问题最简单的方法就是从一开始就不要有问题，所以从威胁模型开始有助于您在设计阶段专注于实现正确的结果。随着 AppSec 计划日渐成熟，您可以增加使用自动化执行的测试数量，提高向构建者提出的反馈的准确性，并减少安全审查所需的时间。所有这些操作都可以提高所构建软件的质量，并加快将新功能推向生产环境的速度。

这些实施指南侧重于四个方面：组织和文化、管道本身的安全、管道中的安全以及依赖项管理。每个方面提供了一组可以实施的原则，并提供了有关如何设计、开发、构建、部署和操作工作负载的端到端视图。

在 AWS 中，可以使用很多方法来处理应用程序安全性计划。其中一些方法依赖于技术，而另一些方法侧重于应用程序安全性计划的人员和组织方面。

最佳实践

- SEC11-BP01 应用程序安全性培训
- SEC11-BP02 在整个开发和发布生命周期中执行自动化测试
- SEC11-BP03 定期执行渗透测试
- SEC11-BP04 人工代码审查
- SEC11-BP05 集中管理服务，方便获取软件包和依赖项
- SEC11-BP06 以编程方式部署软件
- SEC11-BP07 定期评测管道的安全属性
- SEC11-BP08 建立规程，让工作负载团队负责安全领域

SEC11-BP01 应用程序安全性培训

向贵组织的构建者提供培训，使其了解有关安全开发和操作应用程序的常见做法。在开发时采取安全至上的做法，有助于降低到安全审查阶段才会检测出问题的可能性。

期望结果：依照安全的理念设计和构建软件。如果组织中的构建者接受始于威胁建模的安全开发实践培训，就可以提高所开发软件的整体质量和安全性。因为经过安全审查阶段之后所需的返工较少，所以此方法可以减少交付软件或功能的时间。

就本最佳实践而言，安全开发是指正在编写的软件以及支持软件开发生命周期（SDLC）的工具或系统。

常见反模式：

- 等到安全审查阶段才考虑系统的安全属性。
- 将所有安全决策交给安全团队。
- 未能传达在 SDLC 中作出的决策如何与组织的总体安全期望或策略相关联。
- 太迟参与安全审查过程。

建立此最佳实践的好处：

- 在开发周期的早期更好地了解组织对安全性的要求。
- 能够更快地识别和修复潜在的安全问题，从而更快地交付功能。
- 提高软件和系统的质量。

在未建立这种最佳实践的情况下暴露的风险等级：中等

实施指导

为组织中的构建者提供培训。从威胁建模课程开始，为帮助进行安全培训打下了良好的基础。理想情况下，构建者应该能够自助访问与其工作负载相关的信息。这种访问有助于他们就所构建系统的安全属性作出明智的决策，而不需要询问另一个团队。应该明确地定义让安全团队参与审查的流程，且这个流程应该简单易行。安全培训中应包括审查流程中的步骤。如果提供了已知的实施模式或模板，应该很容易找到它们并与总体安全要求关联起来。考虑使用 AWS CloudFormation、AWS Cloud Development Kit (AWS CDK) Constructs、Service Catalog 或其他模板工具减少对自定义配置的需求。

实施步骤

- 让构建者从有关威胁建模的课程开始，打下很好的基础，并帮助培训他们如何思考安全性。
- 让他们可以访问 AWS 培训与认证、行业或 AWS 合作伙伴培训。
- 提供有关组织安全审查流程的培训，明确安全团队、工作负载团队和其他利益相关方之间的职责分工。
- 发布有关如何满足安全要求的自助服务指南，包括代码示例和模板（如可用）。
- 定期从构建者团队那里获得有关他们在安全审查流程和培训方面的体验反馈，并利用这些反馈来作出改进。
- 使用实际试用或错误大扫除活动来帮助减少问题数量，以及增强构建者的技能。

资源

相关最佳实践：

- SEC11-BP08 建立规程，让工作负载团队负责安全领域

相关文档：

- AWS 培训和认证
- 如何看待云安全治理
- 如何处理威胁建模
- 加速培训 – AWS Skills Guild

相关视频：

- 主动式安全性：注意事项和方法

相关示例：

- 有关威胁建模的研讨会
- 开发人员的行业意识

相关服务：

- AWS CloudFormation
- AWS Cloud Development Kit (AWS CDK)（AWS CDK）Constructs
- Service Catalog
- AWS BugBust

SEC11-BP02 在整个开发和发布生命周期中执行自动化测试

在整个开发和发布生命周期中自动测试安全属性。自动化使得在发布软件前，更加容易始终如一反复识别软件中可能存在的问题，进而减少所提供的软件中存在安全问题的风险。

期望结果：自动测试的目标是提供一种程序化方式，在整个开发生命周期中尽早常常检测潜在问题。自动执行回归测试时，您可以重新运行功能测试和非功能测试，确认以前测试过的软件在更改后仍按预期执行。定义安全性单元测试以检查常见的错误配置（如身份验证中断或缺失）时，可以在开发过程的早期识别并修复这些问题。

测试自动化根据应用程序的要求和期望的功能，使用专门构建的测试用例进行应用程序验证。自动测试的结果基于将生成的测试输出与其各自的预期输出进行比较，从而加快整个测试生命周期。回归测试和单元测试套件等测试方法最适合自动化。自动执行安全属性的测试使构建者无需等待安全审查即可自动接收反馈。静态或动态代码分析形式的自动化测试可以提高代码质量，并帮助在开发生命周期的早期检测潜在的软件问题。

常见反模式：

- 不传达自动化测试的测试用例和测试结果。
- 就在发布之前执行自动化测试。
- 使用自动化测试用例来应对经常变化的需求。
- 未能就如何处理安全测试的结果提供指导。

建立此最佳实践的好处：

- 减少对评估系统安全属性的人员的依赖。
- 在多个工作流程中得到一致的结果可提高一致性。
- 降低在生产软件中引入安全问题的可能性。
- 由于及早发现软件问题，可以缩短检测和修复之间的时间。
- 增加多个工作流中的系统或重复行为的可见性，可用于促进组织范围内的改进。

在未建立这种最佳实践的情况下暴露的风险等级：中等

实施指导

在构建软件时，采用各种软件测试机制，以确保根据应用程序的业务逻辑测试应用程序的功能要求和非功能要求（重点关注应用程序可靠性、性能和安全性）。

静态应用程序安全性测试（SAST）分析源代码是否存在异常安全模式，并指出容易出现缺陷的代码。SAST 依赖于文档（需求规范、设计文档和设计规范）和应用程序源代码等静态输入来测试一系列已知的安全问题。静态代码分析器可以帮助加快大量代码的分析。NIST Quality Group 对源代码安全性分析器进行了比较，包括针对字节码扫描器和二进制码扫描器的开源工具。

动态分析安全测试（DAST）方法针对正在运行的应用程序执行测试，以识别潜在的意外行为，能够对静态测试作出补充。动态测试可用于检测通过静态分析无法检测到的潜在问题。通过在代码存储库、构建和管道阶段进行测试，您可以检查出不同类型的潜在问题，防止这些问题进入到代码中。Amazon CodeWhisperer 在构建器的 IDE 中提供代码建议，包括安全扫描。Amazon CodeGuru Reviewer 可以识别应用程序开发过程中的关键问题、安全问题以及难以发现的错误，并提供可提高代码质量的建议。

通过开发人员安全性研讨会，学会使用 AWS 开发人员工具（例如，AWS CodeBuild、AWS CodeCommit 和 AWS CodePipeline）来自动执行发布管道，包括 SAST 和 DAST 测试方法。

在 SDLC 中，建立一个迭代过程，其中包括与安全团队一起定期审查应用程序。在发布准备情况审查过程中，应该处理和验证从这些安全审查中收集到的反馈。这些审查建立了健壮的应用程序安全态势，并为构建者提供切实可行的反馈，以解决潜在问题。

实施步骤

- 实施一致的 IDE、代码审查和 CI/CD 工具，其中包括安全测试。
- 考虑在 SDLC 中的哪个阶段适合阻塞管道，而不仅仅是通知构建者需要修复问题。
- 开发人员安全性研讨会提供了将静态和动态测试集成到发布管道的示例。
- 使用自动化工具（例如，与开发人员 IDE 集成的 Amazon CodeWhisperer，以及用于在提交时扫描代码的 Amazon CodeGuru Reviewer）执行测试或代码分析，帮助构建者适时获得反馈。
- 使用 AWS Lambda 构建应用程序时，您可以使用 Amazon Inspector 来扫描函数中的应用程序代码。
- 利用 AWS CI/CD 研讨会作为在 AWS 上构建 CI/CD 管道的起点。
- 当 CI/CD 管道中包括自动化测试时，您应该使用工单系统来跟踪软件问题的通知和修正。
- 对于可能生成结果的安全测试，链接到补救指南可帮助构建者提高代码质量。

- 定期分析使用自动化工具获得的结果，以确定下一个自动化、构建者培训或认知宣传活动的优先级。

资源

相关文档：

- 持续交付和持续部署

- AWS DevOps 能力合作伙伴
- 适用于应用程序安全性的 AWS 安全能力合作伙伴
- 选择 Well-Architected CI/CD 方法
- 使用 Amazon EventBridge 和 Amazon CloudWatch Events 监控 CodeCommit 事件
- Amazon CodeGuru 审查中的密钥检测
- 通过有效的治理加快 AWS 上的部署
- AWS 方法如何自动执行安全、不需要人工介入的部署

相关视频：

- 不需要人工介入：在亚马逊自动实现持续交付管道
- 自动执行跨账户 CI/CD 管道

相关示例：

- 开发人员的行业意识
- AWS CodePipeline 治理（GitHub）
- 开发人员安全性研讨会
- AWS CI/CD 研讨会

SEC11-BP03 定期执行渗透测试

定期对软件执行渗透测试。此机制有助于识别无法通过自动化测试或人工代码审查检测到的潜在软件问题。它还有助于了解检测控制的有效性。渗透测试应设法确定软件是否会以意外方式执行，例如公开应受保护的数据，或者授予比预期更广泛的权限。

期望结果：使用渗透测试来检测、修复和验证应用程序的安全属性。在软件开发生命周期（SDLC）中应定期执行计划的渗透测试。在发布软件之前应处理渗透测试的结果。您应该分析渗透测试的结果，以确定是否存在使用自动化可以发现的问题。拥有包括主动反馈机制的定期且可重复渗透测试流程，有助于为构建者提供指导并提高软件质量。

常见反模式：

- 仅对已知或普遍存在的安全问题进行渗透测试。
- 未使用相关的第三方工具和库对应用程序执行渗透测试。
- 仅对软件包安全问题进行渗透测试，而不评估已实施的业务逻辑。

建立此最佳实践的好处：

- 在发布之前增强对软件安全属性的信心。
- 有机会确定首选的应用程序模式，从而提高软件质量。
- 获得一个反馈环路，在开发周期早期确定自动化或额外培训可以在哪些方面改进软件的安全属性。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

渗透测试是一项结构化安全测试练习，让您可以运行计划的安全漏洞方案，以便检测、修复和验证安全控制机制。渗透测试从侦察开始，在这个过程中，根据应用程序的当前设计及其依赖项收集数据。生成并运行特定于安全方面的测试方案的精选列表。这些测试的主要目的是发现应用程序中的安全问题，有人会利用这些安全问题来获得对环境的非预期访问，或未经授权访问数据。当推出新功能时，或者应用程序的功能或技术实施方面发生重大变更时，您应该执行渗透测试。

您应该确定在开发生命周期的哪个阶段执行渗透测试最为合适。应当尽量晚些时候执行此测试，以便系统功能接近预期的发布状态，但也要留有足够的时间来修复任何问题。

实施步骤

- 采用结构化流程来确定渗透测试的范围，让这个流程基于威胁模型是保留场景相关性的好方法。
- 确定在开发周期的什么阶段执行渗透测试较为合适。这个阶段应该是在应用程序预期改动很细微，但仍留有足够时间进行修复的时候。
- 为构建者提供以下方面的培训：从渗透测试结果中可以期待获得什么，以及如何获得有关修复的信息。

- 使用工具自动执行常见或可重复的测试，从而加快渗透测试的速度。
- 分析渗透测试结果，以便确定系统性安全问题，并使用此数据为额外的自动化测试和正在进行的构建者培训提供信息。

资源

相关最佳实践：

- SEC11-BP01 应用程序安全性培训
- SEC11-BP02 在整个开发和发布生命周期中执行自动化测试

相关文档：

- AWS 渗透测试提供有关 AWS 上的渗透测试的详细指导
- 通过有效的治理加快 AWS 上的部署
- AWS 安全能力合作伙伴
- 使 AWS Fargate 上的渗透测试架构实现现代化改造
- AWS Fault injection Simulator

相关示例：

- 使用 AWS CodePipeline 自动执行 API 测试（GitHub）
- 自动安全助手（GitHub）

SEC11-BP04 人工代码审查

对您制作的软件执行人工代码审查。此流程有助于确保编写代码的人不是唯一检查代码质量的人。

期望结果：在开发过程中纳入人工代码审查步骤可以提高所编写的软件的质量，帮助团队中缺乏经验的成员提升自身技能，并且有助于确定哪些情况可以使用自动化。自动化工具和测试可以支持人工代码审查。

常见反模式：

- 在部署前不执行代码审查。

- 让同一个人编写和审查代码。
- 不使用自动化来协助或编排代码审查。
- 在审查代码之前没有对构建者进行应用程序安全方面的培训。

建立此最佳实践的好处：

- 提高代码质量。
- 通过重复利用通用方法提高代码开发的一致性。
- 减少在渗透测试和后续阶段发现的问题。
- 改进团队内部的知识传授。

在未建立这种最佳实践的情况下暴露的风险等级：中等

实施指导

在整个代码管理流程中应实施审查步骤。具体细节取决于分支、拉取请求和合并请求所用的方法。您可以使用 AWS CodeCommit 或第三方解决方案，例如 GitHub、GitLab 或 Bitbucket。无论使用哪种方法，务必要确认在将代码部署到生产环境中之前，您的流程需要进行代码审查。使用 Amazon CodeGuru Reviewer 等工具可以更轻松地编排代码审查过程。

实施步骤

- 在代码管理流程中实施人工审查步骤，并在继续之前执行此审查。
- 考虑使用 Amazon CodeGuru Reviewer 来管理和协助代码审查工作。
- 实施审批流程，要求在代码需要完成代码审查后方可进入下一阶段。
- 验证是否存在这样一个流程：识别在人工代码审查期间发现并可以自动检测到的问题。
- 根据您的代码开发实践集成人工代码审查步骤。

资源

相关最佳实践：

- SEC11-BP02 在整个开发和发布生命周期中执行自动化测试

相关文档：

- 在 AWS CodeCommit 存储库中使用拉取请求
- 在 AWS CodeCommit 中使用审批规则模板
- 关于 GitHub 中的拉取请求
- 使用 Amazon CodeGuru Reviewer 自动审查代码
- 使用 Amazon CodeGuru Reviewer CLI 自动检测 CI/CD 管道中的安全漏洞和错误

相关视频：

- 使用 Amazon CodeGuru 持续改进代码质量

相关示例：

- 开发人员安全性研讨会

SEC11-BP05 集中管理服务，方便获取软件包和依赖项

提供集中管理的服务，方便构建者团队获取软件包和其他依赖项。通过采取这种做法，可以在将软件包纳入所编写的软件之前，对软件包进行验证；另外，还可以为分析贵组织所使用的软件提供数据来源。

期望结果：除了正在编写的代码之外，软件还包含一组其他软件包。这样就可以轻松实施可重复使用的功能，例如 JSON 解析器或加密库。从逻辑上将这些软件包和依赖项的来源集中在一起，从而为安全团队提供了一种机制，可以在使用软件包之前对其属性进行验证。这种方法还降低了由于现有软件包中的更改或构建者团队（包括直接来自互联网的任意软件包）所引起的意外问题的风险。使用此方法与手动和自动测试流程相结合，增加对所开发软件质量的信心。

常见反模式：

- 从互联网上的任意存储库中提取软件包。
- 在将新软件包提供给构建者之前不进行测试。

建立此最佳实践的好处：

- 更好地了解正在构建的软件中使用了哪些软件包。
- 了解谁使用了哪些软件包后，在需要更新软件包时，能够向工作负载团队发出通知。
- 降低软件中存在问题软件包的风险。

在未建立这种最佳实践的情况下暴露的风险等级：中等

实施指导

以构建者易于使用的方式为软件包和依赖项提供集中管理服务。集中管理服务可以在逻辑上集中，而不用作为一个整体系统来实施。利用此方法，您可以通过满足构建者需求的方法来提供服务。您应该实施一种有效的方法：在发生更新或出现新需求时将软件包添加到存储库。AWS CodeArtifact 等 AWS 服务或类似的 AWS 合作伙伴解决方案提供了一种实现此功能的方法。

实施步骤：

- 实施可在用于开发软件的所有环境中使用的逻辑集中式存储库服务。
- 在 AWS 账户 账户分配过程中包括对存储库的访问权限。
- 构建自动化以在存储库中发布软件包之前对其进行测试。
- 维护最常用软件包、语言和更改量最大的团队的指标。
- 为构建者团队提供一种自动化机制来请求新软件包和提供反馈。
- 定期扫描存储库中的软件包，以确定新发现的问题的潜在影响。

资源

相关最佳实践：

- SEC11-BP02 在整个开发和发布生命周期中执行自动化测试

相关文档：

- 通过有效的治理加快 AWS 上的部署
- 使用 CodeArtifact Package Origin Control 工具包加强软件包的安全性
- 使用 Amazon CodeGuru Reviewer 检测日志记录中的安全问题
- 软件构件的供应链级别（SLSA）

相关视频：

- 主动式安全性：注意事项和方法
- AWS 安全理念（re:Invent 2017）

- 当安全、保障和紧迫性都很重要时：处理 Log4Shell

相关示例：

- 多区域软件包发布管道（GitHub）
- 使用 AWS CodePipeline 在 AWS CodeArtifact 上发布 Node.js 模块（GitHub）
- AWS CDK Java CodeArtifact 管道示例（GitHub）
- 使用 AWS CodeArtifact 分发专用 .NET NuGet 包（GitHub）

SEC11-BP06 以编程方式部署软件

尽可能以编程方式部署软件。通过采取这种做法，可以降低由于人为错误导致部署失败或引入意外问题的可能性。

期望结果：让人们远离数据是在 AWS Cloud 中安全构建的一项关键原则。此原则包括如何部署软件。

不依赖人来部署软件的好处是，您可以更加确信，您测试的内容就是部署的内容，并且确信每次都一致地执行部署。无需更改软件即可在不同的环境中运行。使用十二要素应用程序开发原则，特别是配置的外部化，使您无需更改即可将相同的代码部署到多个环境。对软件包进行加密签名可以很好地确认不同环境之间什么也没有改变。这种方法的总体结果是降低更改过程中的风险以及提升软件版本的一致性。

常见反模式：

- 手动将软件部署到生产环境中。
- 手动对软件进行更改，以适应不同的环境。

建立此最佳实践的好处：

- 增强对软件发布过程的信心。
- 降低了失败的更改对业务功能造成影响的风险。
- 由于更改风险降低，从而加快了发布节奏。
- 针对部署过程中的意外事件的自动回滚功能。
- 能够以加密方式证明所测试的软件是部署的软件。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

构建 AWS 账户 结构时减少持续的人类访问环境的情况，并使用 CI/CD 工具来进行部署。适当地设计应用程序，以便从外部源（例如，AWS Systems Manager Parameter Store）获得特定于环境的配置数据。在测试软件包后对其进行签名，并在部署期间验证这些签名。配置 CI/CD 管道以推送应用程序代码，并使用金丝雀来确认已成功部署。使用 AWS CloudFormation 或 AWS CDK 等工具来定义基础设施，然后使用 AWS CodeBuild 和 AWS CodePipeline 来执行 CI/CD 操作。

实施步骤

- 构建明确定义的 CI/CD 管道，以便简化部署过程。
- 使用 AWS CodeBuild 和 AWS Code Pipeline 来提供 CI/CD 功能，从而更容易将安全测试集成到管道中。
- 遵循使用多个账户组织 AWS 环境白皮书中有关环境分离的指导。
- 确认在运行生产工作负载的环境中没有持续的人类访问。
- 设计应用程序以支持配置数据外部化。
- 考虑使用蓝绿部署模式进行部署。
- 实施金丝雀以验证软件是否成功部署。
- 使用 AWS Signer 或 AWS Key Management Service（AWS KMS）等加密工具为部署的软件包签名和进行验证。

资源

相关最佳实践：

- SEC11-BP02 在整个开发和发布生命周期中执行自动化测试

相关文档：

- AWS CI/CD 研讨会
- 通过有效的治理加快 AWS 上的部署
- 自动执行安全、不需要人工介入的部署
- 使用 AWS Certificate Manager Private CA 和 AWS Key Management Service 非对称密钥进行代码签名
- 代码签名，用于 AWS Lambda 的可信度和完整性控制措施

相关视频：

- 不需要人工介入：在亚马逊自动实现持续交付管道

相关示例：

- 使用 AWS Fargate 进行蓝绿部署

SEC11-BP07 定期评测管道的安全属性

对您的管道运用 Well-Architected 安全性支柱原则，尤其注意权限分离。定期评测管道基础设施的安全属性。通过有效管理管道的安全性，可以确保通过管道的软件的安全性。

期望结果：用于构建和部署软件的管道应遵循与环境中任何其他工作负载相同的推荐做法。正在使用测试的构建者应该不能编辑在管道中实施的测试。管道应该只拥有它们正在执行的部署所需的权限，并应实施保护措施以避免部署到错误的环境。管道不应该依赖长期凭证，且应配置为发出状态，以便可以验证构建环境的完整性。

常见反模式：

- 构建者可以绕过安全测试。
- 用于部署管道的权限过于宽松。
- 未将管道配置为验证输入。
- 不定期审查与 CI/CD 基础设施关联的权限。
- 使用长期或硬编码凭证。

建立此最佳实践的好处：

- 对通过管道构建和部署的软件的完整性有了更大的信心。
- 在出现可疑活动时可以停止部署。

在未建立这种最佳实践的情况下暴露的风险等级：高

实施指导

从支持 IAM 角色的托管 CI/CD 服务开始，可以降低凭证泄露的风险。将安全性支柱原则应用到 CI/CD 管道基础设施有助于确定可以在哪些方面作出安全改进。遵循 AWS 部署管道参考架构是构建 CI/CD

环境的一个很好的起点。定期审查管道实施和分析日志以了解意外行为，这样有助于您了解用于部署软件的管道的使用模式。

实施步骤

- 从 AWS 部署管道参考架构开始。
- 考虑使用 AWS IAM Access Analyzer 以编程方式生成管道的最低权限 IAM 策略。
- 将管道与监控和警报集成在一起，以便在发生意外或异常活动时您会得到通知，对于 AWS 托管服务，Amazon EventBridge 允许您将数据路由到目标，例如 AWS Lambda 或 Amazon Simple Notification Service（Amazon SNS）。

资源

相关文档：

- AWS 部署管道参考架构
- 监控 AWS CodePipeline
- AWS CodePipeline 的安全最佳实践

相关示例：

- DevOps 监控控制面板（GitHub）

SEC11-BP08 建立规程，让工作负载团队负责安全领域

建立规程或机制，使构建者团队能够针对创建的软件作出安全决策。这些决策仍然需要由安全团队通过审查加以验证，但让构建者团队负责安全领域可以构建速度更快、安全性更高的工作负载。此机制还可促进负责任文化，进而对所构建系统的运营产生积极影响。

期望结果：为了让构建者团队负责安全性和决策制定，您可以向构建者团队提供安全思维方式方面的培训，或者让安全人员加入构建者团队或与构建者团队联系在一起，从而增强他们的培训。这两种方法都有效，并且可以让团队在开发周期的早期作出更高质量的安全决策。这种负责任模式基于应用程序安全性培训。从特定工作负载的威胁建模开始，有助于将设计思维集中在合适的场景中。拥有一个专注于安全的构建者社区，或者拥有一组与构建者团队合作的安全工程师带来的另一项好处是，您可以更深入地了解如何编写软件。这种了解帮助您确定自动化能力的下一个改进领域。

常见反模式：

- 将所有安全设计决策交给安全团队。
- 在开发过程中没有及早满足安全要求。
- 没有从构建者和安全人员那里获得关于计划运营的反馈。

建立此最佳实践的好处：

- 缩短完成安全审查的时间。
- 减少等到安全审查阶段才检测到安全问题的情况。
- 提高所编写软件的整体质量。
- 有机会识别和了解系统性问题或高价值改进领域。
- 进行安全审查后，发现的问题可以在早期进行修复，从而减少所需的返工量。
- 提升对安全功能的认知。

在未建立这种最佳实践的情况下暴露的风险等级：低

实施指导

从SEC11-BP01 应用程序安全性培训中的指导开始。然后确定您认为可能最适合您组织的计划的运营模式。两个主要模式是对构建者进行培训，或在构建者团队中加入安全人员。确定初始方法后，应使用单个或一小组工作负载团队进行试点，以证明该模式适用于您的组织。来自组织的构建者和安全团队的领导层支持有助于计划的成功交付。在构建此计划时，重要的是选择可以用来显示项目价值的指标。了解 AWS 如何解决这个问题是一个很好的学习经验。这个最佳实践非常注重组织变革和文化。您使用的工具应支持构建者和安全社区之间的协作。

实施步骤

- 首先对构建者进行应用程序安全性培训。
- 创建一个社区和入门培训计划来对构建者进行培训。
- 为计划选择一个名称。通常使用守护者、拥护者或倡导者。
- 确定要使用的模式：培训构建者、加入安全工程师或具有相关性安全角色。
- 从安全性、构建者和可能的其他相关团体中确定项目发起人。
- 跟踪参与计划的人数、审查所花时间以及来自构建者和安全人员的反馈等指标。使用这些指标来作出改进。

资源

相关最佳实践：

- SEC11-BP01 应用程序安全性培训
- SEC11-BP02 在整个开发和发布生命周期中执行自动化测试

相关文档：

- 如何处理威胁建模
- 如何看待云安全治理

相关视频：

- 主动式安全性：注意事项和方法

总结

安全性是一项持续性的工作。事件发生时，应将其视为提高架构安全性的机会。拥有强大的身份控制、自动响应安全事件、在多个级别保护基础设施以及通过加密管理合理分类的数据，可以提供每个组织都应实施的深度防御。借助本白皮书中讨论的编程函数以及 AWS 功能和服务，您可以更加轻松地执行这项工作。

AWS 致力于帮助您构建和运营既能保护信息、系统和资产，又能提供商业价值的架构。

贡献者

以下是对本文做出贡献的个人和组织：

- Sarita Dharankar，Amazon Web Services Well-Architected 部门的 Security Pillar Lead
- Adam Cerini，Amazon Web Services Senior Solution Architect
- Bill Shinn，Amazon Web Services Office of the CISO 的 Senior Principal
- Brigid Johnson，Amazon Web Services 的 AWS Identity 部门的 Senior Software Development Manager
- Byron Pogson，Amazon Web Services Senior Solution Architect
- Charlie Hammell，Amazon Web Services Principal Enterprise Architect
- Darran Boyd，Amazon Web Services Financial Services 部门的 Principal Security Solutions Architect
- Dave Walker，Amazon Web Services Security and Compliance 部门的 Principal Specialist Solutions Architect
- John Formento，Amazon Web Services Senior Solution Architect
- Paul Hawkins，Amazon Web Services Office of the CISO 的 Principal
- Sam Elmalak，Amazon Web Services Senior Technology Leader
- Pat Gaw，Amazon Web Services Principal Security Consultant
- Daniel Begimher，Amazon Web Services Senior Consultant, Security
- Danny Cortegaca，Amazon Web Services Senior Security Solutions Architect
- Ana Malhotra，Amazon Web Services Security Solutions Architect
- Debashis Das，Amazon Web Services Office of the CISO 的 Principal
- Reef Dsouza，Amazon Web Services Principal Solutions Architect
- Brad Burnett，Amazon Web Services Identity 部门的 Security Solutions Architect
- Anna McAbee，Amazon Web Services Threat Detection and Incident Response 部门的 Senior Security Solutions Architect
- Jason Garman，Amazon Web Services Principal Security Solutinos Architect

延伸阅读

如需更多帮助，请查阅以下资源：

- AWS Well-Architected Framework 白皮书
- AWS Architecture Center

文档修订

要获得有关此白皮书的更新通知，请订阅 RSS 源。

变更	说明	日期
更新了最佳实践指南	根据以下领域的新指南更新了最佳实践：安全操作工作负载和保护传输中数据。	December 6, 2023
更新了最佳实践指南	对事件响应中的指南和最佳实践进行了重大更新。 更新了准备工作中的多项最佳实践。事件响应中增加了两个新领域：运营和事后活动。新增了最佳实践 SEC10-BP08 建立从事件中吸取经验教训的框架。	October 3, 2023
更新了最佳实践指南	根据以下领域的新指南更新了最佳实践：准备和模拟。	July 13, 2023
针对新框架进行了更新。	为最佳实践更新了规范性指南并增加了新的最佳实践。添加了应用程序安全性（AppSec）的新最佳实践领域。	April 10, 2023
已更新白皮书	为最佳实践更新了新的实施指导。	December 15, 2022
已更新白皮书	扩展了最佳实践并增加了改进计划。	October 20, 2022
次要更新	更新了 IAM 信息来反映最新的最佳实践。	June 28, 2022

次要更新	添加了其他 AWS PrivateLink 信息，并修复了错误的链接。	May 19, 2022
次要更新	添加了 AWS PrivateLink。	May 6, 2022
次要更新	删除了非包容性用语。	April 22, 2022
次要更新	添加了有关 VPC 网络访问分析器的信息。	February 2, 2022
次要更新	在简介中添加了可持续性支柱。	December 2, 2021
次要更新	修复错误的链接。	May 27, 2021
次要更新	贯穿全文的编辑性修改。	May 17, 2021
主要更新	添加了有关治理的部分，为各个部分添加了详细信息，并添加了新功能和服务。	May 7, 2021
次要更新	已更新链接。	March 10, 2021
次要更新	修复错误的链接。	July 15, 2020
针对新框架进行了更新	已更新有关账户、身份和权限管理的指导。	July 8, 2020
针对新框架进行了更新	已更新以扩展每个方面的建议、新的最佳实践、服务和功能。	April 30, 2020
已更新白皮书	反映新的 AWS 服务和功能以及最新参考的更新。	July 1, 2018
已更新白皮书	更新了“系统安全配置和维护”一节，以反映新的 AWS 服务和功能。	May 1, 2017

原始版本	已发布安全性支柱 – AWS Well-Architected Framework。	November 1, 2016

声明

客户负责对本文档中的信息进行独立评估判断。本文档：（a）仅供参考，（b）代表 AWS 当前的产品服务和实践，如有变更，恕不另行通知，以及（c）不构成 AWS 及其附属公司、供应商或授权商的任何承诺或保证。AWS 产品或服务均“按原样”提供，没有任何明示或暗示的担保、声明或条件。AWS 对其客户的责任和义务由 AWS 协议规定，本文档与 AWS 和客户之间签订的任何协议无关，亦不影响任何此类协议。