aws

用户指南

Amazon Inspector

Amazon Inspector: 用户指南

Table of Contents

Amazon Inspector 是什么？

Amazon Inspector 是一项漏洞管理服务，它会持续扫描您的 AWS 工作负载，查找软件漏洞和意外的网络暴露。Amazon Inspector 会自动发现和扫描正在运行的 Amazon EC2 实例、Amazon Elastic Container Registry (Amazon ECR) 中的容器映像和 AWS Lambda 函数，以查找已知软件漏洞和意外网络暴露。

Amazon Inspector 会在发现软件漏洞或网络配置问题时创建调查发现。调查发现会描述漏洞，确定受影响的资源，对漏洞的严重性进行评级，并提供修复指导。可以使用 Amazon Inspector 控制台分析调查发现，也可以通过其他 AWS 服务 查看和处理调查发现。有关更多信息，请参阅了解 Amazon Inspector 中的调查发现。

主题

Amazon Inspector 的特征

集中管理多个 Amazon Inspector 账户

如果您的 AWS 环境包含多个账户，则可以使用 AWS 组织通过一个账户集中管理环境。使用此方法时，您可以将某一账户指定为 Amazon Inspector 的委托管理员账户。

只需单击一下即可为整个组织激活 Amazon Inspector。此外，您还可以在将来有成员加入组织时自动为他们激活服务。Amazon Inspector 委托管理员账户可以管理组织成员的调查发现数据和某些设置。这包括查看所有成员账户的汇总调查发现详细信息、为成员账户激活或停用扫描，以及查看 AWS 组织内扫描的资源。

持续扫描环境，查找漏洞和网络风险

有了 Amazon Inspector，便无需手动安排或配置评测扫描。Amazon Inspector 会自动发现并开始扫描符合条件的资源。Amazon Inspector 会自动重新扫描资源，以应对可能会引入新漏洞的变更，在资源的整个生命周期内持续评测您的环境，上述变更包括：在 EC2 实例中安装新程序包、安装补丁，以及发布影响资源的新常见漏洞和风险 (CVE)。与传统的安全扫描软件不同，Amazon Inspector 对机群性能的影响微乎其微。

当发现漏洞或开放的网络路径时，Amazon Inspector 会生成调查发现供您调查。调查发现包括有关漏洞、受影响资源和补救建议的全面详细信息。如果您对调查发现进行了适当的补救，Amazon Inspector 会自动检测到补救措施并关闭该调查发现。

使用 Amazon Inspector 风险评分准确评测漏洞

当 Amazon Inspector 通过扫描收集有关环境的信息时，它会提供专门针对您的环境量身定制的严重性评分。Amazon Inspector 会检查构成漏洞的国家漏洞数据库 (NVD) 基本评分的安全指标，并根据您的计算环境进行调整。例如，如果漏洞可通过网络利用，但互联网上没有通往相应 Amazon EC2 实例的开放网络路径，则该服务可能会降低该实例的调查发现的 Amazon Inspector 评分。该评分采用 CVSS 格式，是对 NVD 提供的基本通用漏洞评分系统 (CVSS) 评分的修改。

使用 Amazon Inspector 控制面板识别具有高影响力的调查发现

Amazon Inspector 控制面板可提供整个环境中调查发现的总体视图。您可以在此控制面板中查看调查发现的详细信息。此控制面板包含有关环境中扫描覆盖范围、最重要的调查发现以及调查发现最多的资源的简化信息。Amazon Inspector 控制面板中基于风险的补救面板显示了影响实例和映像数量最多的调查发现。通过此面板，您可以更轻松地确定对环境影响最大的调查发现，查看调查发现的详细信息以及建议的解决方案。

使用自定义视图管理调查发现

除了控制面板外，Amazon Inspector 控制台还提供了调查发现视图。此页面列出了您的环境的所有调查发现，并提供了各个调查发现的详细信息。您可以查看按类别或漏洞类型分组的调查发现。在每个视图中，您都可以使用筛选条件进一步自定义结果。您还可以使用筛选条件创建抑制规则，在视图中隐藏不需要的调查发现。

您可以使用筛选条件和抑制规则生成调查发现报告，展示所有调查发现或自定义的调查发现。报告可以使用 CSV 或 JSON 格式生成。

使用其他服务和系统监控和处理调查发现

为了支持与其他服务和系统的集成，Amazon Inspector 会将调查发现作为调查发现事件发布到 Amazon EventBridge。EventBridge 是一种无服务器事件总线服务，可以将调查发现数据路由到目标，例如 AWS Lambda 函数和 Amazon Simple Notification Service (Amazon SNS) 主题。借助 EventBridge，您可以近乎实时地监控和处理调查发现，并将其作为现有安全和合规工作流程的一部分。

如果已激活 AWS Security Hub，那么 Amazon Inspector 还会将调查发现发布到 Security Hub。Security Hub 服务提供了 AWS 环境中安全状况的全面视图，可帮您检查环境是否符合安全行业

标准和最佳实践。借助 Security Hub，您可以更轻松地监控和处理调查发现，并将其作为对 AWS 环境中组织安全状况的更广泛分析的一部分。

访问 Amazon Inspector

大多数 AWS 区域都可以使用 Amazon Inspector。有关当前可使用 Amazon Inspector 的区域的列表，请参阅 Amazon Web Services 一般参考中的 Amazon Inspector 端点和配额。要了解有关 AWS 区域的更多信息，请参阅 Amazon Web Services 一般参考中的管理 AWS 区域。在每个区域，您都可以通过以下方式使用 Amazon Inspector：

AWS 管理控制台

AWS Management Console是一个基于浏览器的界面，可用于创建和管理 AWS 资源。作为该控制台的一部分，Amazon Inspector 控制台提供对 Amazon Inspector 账户和资源的访问。您可以通过 Amazon Inspector 控制台执行 Amazon Inspector 任务。

AWS 命令行工具

借助 AWS 命令行工具，您可在系统的命令行中发出命令，执行 Amazon Inspector 任务。与控制台相比，使用命令行更快、更方便。如果要构建执行任务的脚本，命令行工具也会十分有用。

AWS 提供两组命令行工具：AWS Command Line Interface (AWS CLI) 和 AWS Tools for PowerShell。有关安装和使用 AWS CLI 的信息，请参阅 AWS 命令行界面用户指南。有关安装和使用 Tools for PowerShell 的信息，请参阅 AWS Tools for PowerShell 用户指南。

AWS SDK

AWS 提供的 SDK 包含用于各种编程语言和平台的库和示例代码，例如 Java、Go、Python、C++ 和 .NET。这些 SDK 可以提供对 Amazon Inspector 和其他 AWS 服务的编程式便捷访问。它们可以执行多种任务，例如以加密方式对请求进行签名、管理错误以及自动重试请求等。有关安装和使用 AWS SDK 的信息，请参阅在 AWS 上构建的工具。

Amazon Inspector REST API

Amazon Inspector REST API 让您能够以编程方式全面访问 Amazon Inspector 账户和资源。借助此 API，您可以直接向 Amazon Inspector 发送 HTTPS 请求。但是，与 AWS 命令行工具和 SDK 不同，使用此 API 需要您的应用程序处理底层细节，例如生成哈希值来签署请求。

Amazon Inspector 入门

本教程介绍了 Amazon Inspector 的实际操作。

第 1 步介绍如何在多账户环境中激活独立账户的 Amazon Inspector 扫描，或者以亚马逊检查员委托管理员的身份激活 Amazon Inspector 扫描。 AWS Organizations

第 2 步介绍如何了解控制台中的 Amazon Inspector 调查发现。

> **Note**
>
> 在本教程中，您将以当前状态完成任务 AWS 区域。要在其他区域设置 Amazon Inspector，您必须在每个区域完成这些步骤。

主题

开始前的准备工作

Amazon Inspector 是一项漏洞管理服务，可持续扫描您的 Amazon EC2 实例、Amazon ECR 容器映像和 AWS Lambda 函数，以查找软件漏洞和意外网络泄露。

在激活 Amazon Inspector 之前，请注意以下几点：

- Amazon Inspector 是一项区域服务，数据存储在您使用该服务 AWS 区域 的地方。您要使用 Amazon Inspector 监控的每个 AWS 区域 配置过程都必须重复您在本教程中完成的任何配置过程。
- Amazon Inspector 使您可以灵活地激活 Amazon EC2 实例、亚马逊 ECR 容器镜像和 AWS Lambda 函数扫描。您可以在 Amazon Inspector 控制台的账户管理页面或使用 Amazon Inspector API 管理扫描类型。
- 只有安装并激活了 Amazon EC2 Systems Manager (SSM) 代理，Amazon Inspector 才能为 EC2 实例提供常见脆弱性和风险 (CVE) 数据。许多 EC2 实例上都预装了此代理，但您可能需要手动激活它。不管 SSM 代理状态如何，都会对所有 EC2 实例进行网络暴露问题扫描。有关为 Amazon EC2

配置扫描的更多信息，请参阅扫描 Amazon EC2 实例。Amazon ECR 和 AWS Lambda 功能扫描不需要使用代理。

- 具有管理员权限的 IAM 用户身份 AWS 账户 可以启用 Amazon Inspector。出于数据保护目的，我们建议您保护您的凭证并使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 设置个人用户。这样，每个用户只能获得管理 Amazon Inspector 所需的权限。有关启用 Amazon Inspector 所需的权限的信息，请参阅AWS 托管策略：AmazonInspector2FullAccess。
- 在任何区域首次激活 Amazon Inspector 时，都会为您的账户在全球范围内创建一个名为 `AWSServiceRoleForAmazonInspector2` 的服务相关角色。该角色包括相应权限和信任策略，允许 Amazon Inspector 收集程序包详细信息并分析 Amazon VPC 配置以生成脆弱性调查发现。有关更多信息，请参阅对 Amazon Inspector 使用服务相关角色。有关服务相关角色的更多信息，请参阅使用服务相关角色。

步骤 1：激活 Amazon Inspector

使用 Amazon Inspector 的第一步是为 AWS 账户激活它。激活任意 Amazon Inspector 扫描类型后，Amazon Inspector 会立即开始发现和扫描所有符合条件的资源。

如果您希望通过集中式管理员账户管理组织内多个账户的 Amazon Inspector，则必须为 Amazon Inspector 分配一名委托管理员。选择以下选项之一，了解如何为您的环境激活 Amazon Inspector。

Standalone account environment

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 选择开始使用。
3. 选择激活 Amazon Inspector。

在独立账户中激活 Amazon Inspector 时，默认情况下会激活所有扫描类型。您可以在 Amazon Inspector 控制台的账户管理页面或使用 Amazon Inspector API 管理激活的扫描类型。激活 Amazon Inspector 后，它会自动发现并开始扫描所有符合条件的资源。查看以下扫描类型信息，了解默认情况下哪些资源符合条件：

Amazon EC2 扫描

为了向您的 EC2 实例提供常见漏洞和风险敞口 (CVE) 数据，Amazon Inspector 要求安装并激活 S AWS ystems Manager (SSM) 代理。许多 EC2 实例上都预装了此代理，但您可能需要手

动激活它。不管 SSM 代理状态如何，都会对所有 EC2 实例进行网络暴露问题扫描。有关为 Amazon EC2 配置扫描的更多信息，请参阅使用 Amazon Inspector 扫描 Amazon EC2 实例。

Amazon ECR 扫描

激活 Amazon ECR 扫描后，Amazon Inspector 会将您的私有注册表中为 Amazon ECR 提供的默认基本扫描配置的所有容器存储库转换为使用持续扫描的增强扫描。您也可以选择将此设置配置为仅在推送时扫描或通过包含规则扫描特定存储库。过去 30 天内推送的所有映像都将安排生命周期扫描，此 Amazon ECR 扫描设置可以随时更改。有关为 Amazon ECR 配置扫描的更多信息，请参阅使用 Amazon Inspector 扫描 Amazon ECR 容器映像。

AWS Lambda 功能扫描

当您激活 AWS Lambda 函数扫描时，Amazon Inspector 会发现您账户中的 Lambda 函数并立即开始扫描这些函数以查找漏洞。Amazon Inspector 会在部署新的 Lambda 函数和层时对其进行扫描，并在它们更新或新的常见脆弱性和风险 (CVE) 发布时对其进行重新扫描。Amazon Inspector 提供两种不同级别的 Lambda 函数扫描。默认情况下，首次激活 Amazon Inspector 时，会激活 Lambda 标准扫描，这会扫描函数中的程序包依赖项。此外，您还可以激活 Lambda 代码扫描，以扫描函数中的开发者代码是否存在代码脆弱性。有关配置 Lambda 函数扫描的更多信息，请参阅使用 Amazon Inspector 进行扫描 AWS Lambda。

Multi-account environment

Important

要完成这些步骤，您必须与要管理的所有账户属于同一个组织，并且有权访问 AWS Organizations 管理账户，才能在组织内委派 Amazon Inspector 管理员。委托管理员可能需要其他权限。有关更多信息，请参阅指定委托管理员所需的权限。

Note

要以编程方式为多个区域的多个账户启用 Amazon Inspector，可以使用 Amazon Inspector 开发的 shell 脚本。有关使用此脚本的更多信息，请参阅 insp ector2-enablement-with-cli on。 GitHub

为 Amazon Inspector 委托管理员

1. 登录 AWS Organizations 管理账号。

2. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
3. 在委派管理员窗格中，输入您要指定为该组织的 Amazon Inspector 委托管理员的十二位数 ID。 AWS 账户 然后选择委托。然后，在确认窗口中，再次选择委托。

> **Note**
>
> 委托管理员时，您的账户会激活 Amazon Inspector。

添加成员账户

作为委托管理员，您可以为与组织管理账户关联的所有成员激活扫描。此工作流程会为所有成员账户激活所有扫描类型。不过，成员也可以为自己的账户激活 Amazon Inspector，或者委托管理员可以有选择地激活对服务的扫描。有关更多信息，请参阅管理多个账户。

1. 登录委托管理员账户。
2. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
3. 在导航窗格中，选择账户管理。账户表显示了与组织管理账户关联的所有成员账户。
4. 在账户管理页面上，您可以从顶部横幅中选择 “激活对所有账户的扫描”，以激活 EC2 实例、ECR 容器映像以及对组织中所有账户的 AWS Lambda 功能扫描。或者，您可以在账户表中选择要添加为成员的账户。然后从激活菜单中选择全部扫描。
5. （可选）打开为新成员账户自动激活 Inspector 功能，然后选择要包含的扫描类型，以为添加到组织中的所有新成员账户激活这些扫描。

Amazon Inspector 目前提供对 EC2 实例、ECR 容器镜像和 AWS Lambda 函数的扫描。激活 Amazon Inspector 后，它会自动开始发现和扫描所有符合条件的资源。查看以下扫描类型信息，了解默认情况下哪些资源符合条件：

Amazon EC2 扫描

要为您的 EC2 实例提供 CVE 漏洞数据，Amazon Inspector 需要安装并激活 S AWS ystems Manager (SSM) 代理。许多 EC2 实例上都预装了此代理，但您可能需要手动激活它。不管 SSM 代理状态如何，都会对所有 EC2 实例进行网络暴露问题扫描。有关为 Amazon EC2 配置扫描的更多信息，请参阅使用 Amazon Inspector 扫描 Amazon EC2 实例。

Amazon ECR 扫描

激活 Amazon ECR 扫描后，Amazon Inspector 会将您的私有注册表中为 Amazon ECR 提供的默认基本扫描配置的所有容器存储库转换为使用持续扫描的增强扫描。您也可以选择将此设置配置为仅在推送时扫描或通过包含规则扫描特定存储库。过去 30 天内推送的所有映像都将安排生命周期扫描。委派管理员可以随时更改 Amazon ECR 扫描设置。有关为 Amazon ECR 配置扫描的更多信息，请参阅使用 Amazon Inspector 扫描 Amazon ECR 容器映像。

AWS Lambda 功能扫描

当您激活 AWS Lambda 函数扫描时，Amazon Inspector 会发现您账户中的 Lambda 函数并立即开始扫描这些函数以查找漏洞。Amazon Inspector 会在部署新的 Lambda 函数和层时对其进行扫描，并在它们更新或新的常见脆弱性和风险 (CVE) 发布时对其进行重新扫描。有关配置 Lambda 函数扫描的更多信息，请参阅使用 Amazon Inspector 进行扫描 AWS Lambda。

步骤 2：查看 Amazon Inspector 调查发现

您可以在 Amazon Inspector 控制台中或通过 API 查看您的环境的调查发现。所有调查结果也会推送到 Amazon EventBridge 和 AWS Security Hub （如果已激活）。此外，容器映像调查发现会推送到 Amazon ECR。

Amazon Inspector 控制台为调查发现提供了多种不同的查看格式。Amazon Inspector 控制面板为您提供了环境风险的总体概况，而调查发现表提供了特定调查发现的详细信息。

在此步骤中，您可以使用调查发现表和调查发现控制面板浏览调查发现的详细信息。有关 Amazon Inspector 控制面板的更多信息，请参阅了解控制面板。

要在 Amazon Inspector 控制台中查看您的环境的调查发现的详细信息，请执行以下操作：

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 从导航窗格中选择控制面板。您可以选择控制面板中的任意链接，导航到 Amazon Inspector 控制台中的页面，了解相应项目的更多详细信息。
3. 从导航窗格中选择调查发现。
4. 默认情况下，您将看到 “所有调查结果” 选项卡，其中显示您的环境的所有 EC2 实例、ECR 容器映像和 AWS Lambda 函数查找结果。
5. 在调查发现列表中，在标题列中选择一个调查发现名称，打开其详细信息窗格。所有调查发现都有一个调查发现详细信息选项卡。您可以通过以下方式与调查发现详细信息选项卡进行交互：

- 要了解有关脆弱性的更多详细信息，请点击脆弱性详细信息部分中的链接，打开该脆弱性的文档。
- 要进一步调查资源，请点击受影响的资源部分中的资源 ID 链接，打开受影响资源的服务控制台。

程序包脆弱性类型的调查发现也具有 Inspector 评分和脆弱性情报选项卡，此选项卡解释了该调查发现的 Amazon Inspector 评分是如何计算的，并提供了与该调查发现相关的常见脆弱性和风险 (CVE) 的信息。有关调查发现类型的更多信息，请参阅Amazon Inspector 中的调查发现类型。

了解 Amazon Inspector 控制面板

Amazon Inspector 控制面板提供当前 AWS 区域内 AWS 资源汇总统计数据的快照。这些统计数据包括资源覆盖范围和活动漏洞的关键指标。控制面板还会显示您账户的汇总调查发现数据组，例如包含最重要调查发现的 Amazon Elastic Compute Cloud (Amazon EC2) 实例、Amazon Elastic Container Registry (Amazon ECR) 和 AWS Lambda 函数。要进行更深入的分析，您可以查看控制面板项目的支持数据。

如果您的账户是组织的 Amazon Inspector 委托管理员账户，则控制面板将包含您组织中所有账户（包括您自己的账户）的账户覆盖率、汇总统计数据和调查发现数据。

显示控制面板

控制面板显示环境覆盖范围和重要调查发现的概览。

要显示控制面板，请执行以下操作：

1. 打开 Amazon Inspector 控制台：https://console.aws.amazon.com/inspector/v2/home。
2. 在导航窗格中，选择控制面板。
3. 您可以通过以下方式与控制面板交互：

   - 控制面板每 5 分钟自动刷新一次。不过，您可以选择页面右上角的刷新图标，手动刷新数据。
   - 要查看控制面板上某个项目的支持数据，请选择相应项目。
   - 如果您作为 Amazon Inspector 委托管理员管理 AWS 组织的多个账户，则控制面板会显示您的成员账户的汇总统计数据。要对控制面板应用筛选条件，以仅显示特定账户的数据，请在账户框中输入账户 ID。

了解控制面板组件并解释数据

Amazon Inspector 控制面板的每个部分都提供了对关键指标或活动调查发现数据的见解，可以帮助您了解当前 AWS 区域内 AWS 资源的漏洞状况。

环境覆盖范围

环境覆盖范围部分提供有关 Amazon Inspector 扫描的资源的统计数据。在该部分中，您可以查看 Amazon Inspector 扫描的 Amazon EC2 实例、Amazon ECR 映像和 AWS Lambda 函数的数量和百分比。如果您作为 Amazon Inspector 委托管理员通过 AWS Organizations 管理多个账户，您还

将看到组织账户总数、激活 Amazon Inspector 的账户数量以及由此产生的组织覆盖率数据。您还可以使用此部分来确定哪些资源不在 Amazon Inspector 的覆盖范围内。这些资源可能包含漏洞，这些漏洞可能会被人利用，使您的组织面临风险。有关更多信息，请参阅评测 Amazon Inspector 对 AWS 环境的覆盖率。

选择一个覆盖范围组后，可进入所选组的账户管理页面。账户管理页面显示了有关 Amazon Inspector 覆盖哪些账户、Amazon EC2 实例和 Amazon ECR 存储库的详细信息。

覆盖范围组如下所示：

- 账户
- 实例
- 容器存储库
- 容器映像
- Lambda

重要调查发现

重要调查发现部分提供了环境中重要漏洞的数量以及环境中所有调查发现的总数。在本节中，数量按资源和评测类型显示。有关重要调查发现以及 Amazon Inspector 如何确定重要性的更多信息，请参阅了解 Amazon Inspector 中的调查发现。

选择一个重要调查发现组后，会进入所有调查发现页面，并自动应用筛选条件，显示与所选分组匹配的所有重要调查发现。

重要调查发现组如下所示：

- ECR 容器映像调查发现
- Amazon EC2 调查发现
- 网络可达性调查发现
- AWS Lambda 函数调查发现

基于风险的补救

基于风险的补救部分显示前五个存在严重漏洞的程序包，这些漏洞会影响环境中的大部分资源。修复这些程序包可以显著减少环境面临的关键风险数量。选择程序包名称以查看相关的漏洞详细信息和受影响的资源。

包含最重要调查发现的账户

包含最重要调查发现的账户部分显示环境中包含最重要调查发现的前五个 AWS 账户，以及该账户的调查发现总数。只有当 Amazon Inspector 通过 AWS Organizations 配置为多账户扫描时，才能

通过委托管理员账户查看此部分。此视图可帮助委托管理员了解组织内的哪些账户面临的风险最大。

选择账户 ID 以查看有关受影响成员账户的更多信息。

包含最重要调查发现的 Amazon ECR 存储库

包含最重要调查发现的 Elastic Container Registry (ECR) 存储库部分显示环境中包含最重要容器映像调查发现的前五个 Amazon ECR 存储库。该视图显示存储库名称、AWS 账户标识符、存储库创建日期、严重漏洞数量和漏洞总数。此视图可帮助您确定哪些存储库面临的风险最大。

选择存储库名称以查看有关受影响存储库的更多信息。

包含最重要调查发现的容器映像

包含最重要调查发现的容器映像部分显示环境中包含最重要调查发现的前五个容器映像。该视图显示映像标签数据、存储库名称、映像摘要、AWS 账户标识符、严重漏洞数量和漏洞总数。此视图可帮助应用程序所有者确定哪些容器映像可能需要重建和重新启动。

选择容器映像可查看有关受影响的容器映像的更多信息。

包含最重要调查发现的实例

包含最重要调查发现的实例部分显示包含最重要调查发现的前五个 Amazon EC2 实例。该视图显示实例标识符、AWS 账户标识符、亚马逊机器映像 (AMI) 标识符、严重漏洞数量和漏洞总数。此视图可帮助基础设施所有者确定哪些实例可能需要修补。

选择实例 ID 以查看有关受影响的 Amazon EC2 实例的更多信息。

包含最重要调查发现的亚马逊机器映像 (AMI)

包含最重要调查发现的亚马逊机器映像 (AMI) 部分显示环境中包含最重要调查发现的前五个 AMI。该视图显示 AMI 标识符、AWS 账户标识符、在环境中运行的受影响 EC2 实例的数量、AMI 创建日期、AMI 操作系统平台、严重漏洞的数量和漏洞总数。此视图可帮助基础设施所有者确定哪些 AMI 可能需要重建。

选择受影响实例以查看有关从受影响的 AMI 启动的实例的更多信息。

包含最重要调查发现的 AWS Lambda 函数

包含最重要调查发现的 AWS Lambda 函数部分显示环境中包含最重要调查发现的前五个 Lambda 函数。该视图显示 Lambda 函数名称、AWS 账户标识符、运行时系统环境、严重漏洞的数量、高危漏洞的数量和漏洞总数。此视图可帮助基础设施所有者确定哪些 Lambda 函数可能需要修复。

选择函数名称以查看有关受影响的 AWS Lambda 函数的更多信息。

了解 Amazon Inspector 中的调查发现

调查结果是关于影响您的 AWS 资源之一的漏洞的详细报告。调查结果以检测到的漏洞命名，并提供严重性等级、有关受影响资源的信息以及描述如何修复报告的漏洞的详细信息。

每当 Amazon Inspector 检测到 Amazon EC2 实例、亚马逊 ECR 存储库中的容器镜像或 AWS Lambda 函数中存在漏洞时，它都会生成调查结果。Amazon Inspector 会持续扫描您的计算环境并存储您的所有活动发现，直到您对其进行补救。

当您修复发现时，该发现会自动关闭，Amazon Inspector 会在 7 天后删除该发现。当您删除资源时，Amazon Inspector 会在 30 天后删除与该资源相关的所有发现。

如果您禁用 Amazon Inspector，则搜索结果将在 24 小时后删除。如果您的账户被 AWS 暂停，90 天后将删除发现的结果。

调查结果按以下状态之一进行分类：

处于活动状态

Amazon Inspector 将尚未补救的发现确定为 “有效”。

已抑制

Amazon Inspector 将受一项或多项禁止规则约束的发现识别为 “已禁止”。您可以在 “隐藏的查找结果” 列表中找到隐藏的查找结果。有关更多信息，请参阅使用抑制规则抑制 Amazon Inspector 调查发现。

已关闭

修复漏洞后，Amazon Inspector 会自动检测漏洞并将发现的状态更改为 “已关闭”。已关闭的调查结果将在 7 天后删除。

主题

Amazon Inspector 中的调查发现类型

Amazon Inspector 可为 Amazon Elastic Compute Cloud (Amazon EC2) 实例、Amazon Elastic Container Registry (Amazon ECR) 存储库中的容器映像以及 AWS Lambda 函数生成调查发现。Amazon Inspector 可以生成以下类型的调查发现。

程序包漏洞

程序包漏洞调查发现可识别 AWS 环境中暴露于常见漏洞和风险 (CVE) 的程序包。攻击者可以利用这些未修补的漏洞来破坏数据的保密性、完整性或可用性，或访问其他系统。CVE 系统提供了针对公共已知的信息安全漏洞和风险的参考方法。有关更多信息，请访问 https://www.cve.org/。

针对 Linux 的 CVE 检测会在供应商安全公告发布后 24 小时内添加到 Amazon Inspector 中。针对 Windows 的 CVE 检测会在微软发布后 48 小时内添加到 Amazon Inspector 中。您可以使用 Amazon Inspector 漏洞数据库搜索 来查看 CVE 检测是否受支持。

Amazon Inspector 可以为 EC2 实例、ECR 容器映像和 Lambda 函数生成程序包漏洞调查发现。程序包漏洞调查发现具有该调查发现类型所特有的额外详细信息，即 Inspector 评分和漏洞情报。

代码漏洞

代码漏洞调查发现可识别代码中可能被攻击者利用的代码行。代码漏洞包括注入缺陷、数据泄露、弱加密或代码中缺少加密。

Amazon Inspector 会使用自动推理和机器学习来评估 Lambda 函数应用程序代码，分析应用程序代码的总体安全合规性。它基于与 Amazon CodeGuru 合作开发的内部检测器来识别违反政策的行为和漏洞。有关可能的检测的列表，请参阅 CodeGuru 检测器库。

> **Important**
>
> Amazon Inspector 代码扫描可捕获代码片段以突出显示检测到的漏洞。这些片段可能以纯文本形式显示硬编码的凭证或其他敏感材料。

如果您已激活 Amazon Inspector Lambda 代码扫描，则 Amazon Inspector 可以为 Lambda 函数生成代码漏洞调查发现。

CodeGuru 服务会存储检测到的与代码漏洞相关的代码片段。默认情况下，会使用 CodeGuru 控制的 AWS 自有密钥对代码进行加密，不过，您也可以通过 Amazon Inspector API 使用自己的客户托管密钥进行加密。有关更多信息，请参阅对调查发现中的代码进行静态加密。

网络可达性

网络可达性调查发现表明，您的环境中存在通往 Amazon EC2 实例的开放网络路径。当您的 TCP 和 UDP 端口可以从 VPC 边缘（如互联网网关，包括应用程序负载均衡器或经典负载均衡器后的实例）、VPC 对等连接或使用虚拟网关的 VPN 到达时，就会出现这些调查发现。这些调查发现突出显示了可能过于宽松的网络配置，例如管理不善的安全组、访问控制列表或互联网网关，或者网络配置可能允许潜在的恶意访问。

Amazon Inspector 仅针对 Amazon EC2 实例生成网络可达性调查发现。Amazon Inspector 每 24 小时对网络可达性调查发现进行一次扫描。

Amazon Inspector 在扫描网络路径时会评估以下配置：

- Amazon EC2 实例
- AWS Lambda 函数
- 应用程序负载均衡器
- Direct Connect
- Elastic Load Balancer
- 弹性网络接口
- 互联网网关
- 网络访问控制列表
- 路由表
- 安全组
- 子网
- Virtual Private Cloud
- 虚拟专用网关
- VPC 端点
- VPC 网关端点
- VPC 对等连接
- VPN 连接

查找和查看 Amazon Inspector 调查结果

本节中的程序介绍如何通过亚马逊 Inspector 控制台和 API 在 Amazon Inspector 中查找和查看调查结果。查找的详细信息因发现类型、漏洞类型和受影响的资源而异。有关更多信息，请参阅Amazon Inspector 调查发现详细信息。

Console

在控制台中查看调查发现

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 从导航窗格中选择 “调查结果”。您将被定向到 “调查结果” 屏幕，您可以在其中查看所有发现。在 “调查结果” 表中，您可以通过在 “标题” 列下选择查找结果的名称来选择该查找结果。
3. （可选）您还可以查看按类别分组的调查结果。从导航窗格中选择 F indings，然后选择以下类别之一：
   - 按漏洞分类
   - 按实例划分

     > **Note**
     >
     > 按实例分组的发现结果不包括有关网络可用性的信息。

   - 按容器镜像
   - 按容器存储库排序
   - 通过 Lambda 函数

API

运行 ListFindings API 操作。在请求中，您可以指定 `filterCriteria` 返回特定的调查发现。

Amazon Inspector 调查发现详细信息

在 Amazon Inspector 控制台中，您可以查看每项调查发现的详细信息。调查发现的详细信息因调查发现类型而异。

查看调查发现的详细信息

1. 打开 Amazon Inspector 控制台：https://console.aws.amazon.com/inspector/v2/home
2. 选择要查看调查发现的区域。
3. 在导航窗格中，选择调查发现以显示调查发现列表
4. （可选）使用筛选栏选择特定的调查发现。有关更多信息，请参阅筛选 Amazon Inspector 调查发现。
5. 选择一个调查发现，查看其详细信息面板。

调查发现详细信息面板包含调查发现的基本识别特征。这包括调查发现的标题以及已发现脆弱性的基本描述、补救建议和严重性评分。有关评分的信息，请参阅Amazon Inspector 调查发现的严重性级别。

调查发现的详细信息因调查发现类型和受影响的资源而异。

所有发现都包含发现结果的 AWS 账户 ID 号、严重性、发现类型、查找结果的创建日期，以及包含该资源详细信息的 “受资源影响” 部分。

调查发现类型决定了可用于该调查发现的补救措施和脆弱性情报信息。根据查找类型的不同，可用的查找结果详细信息也有所不同。

程序包脆弱性

程序包脆弱性调查发现适用于 EC2 实例、ECR 容器映像和 Lambda 函数。有关更多信息，请参阅程序包漏洞。

程序包脆弱性调查发现还包括 Amazon Inspector 分数和漏洞情报。

此调查发现类型具有以下详细信息：

- 修复可用 – 表示脆弱性是否已在受影响程序包的更新版本中修复。具有下列值之一：
  - YES，这意味着所有受影响的程序包都有已修复脆弱性的版本。
  - NO，这意味着受影响的程序包没有已修复脆弱性的版本。
  - PARTIAL，这意味着一个或多个（但不是全部）受影响的程序包具有已修复脆弱性的版本。
- 攻击脆弱性可用 – 表示脆弱性具有已知的攻击风险。
  - YES，这意味着在您的环境中发现的脆弱性具有已知的攻击风险。Amazon Inspector 无法查看环境中攻击脆弱性的利用情况。
  - NO，这意味着脆弱性没有已知的攻击风险。

- 受影响的程序包 – 列出了调查发现中标记为易受攻击的各个程序包，以及每个程序包的详细信息：
- 文件路径 – 与调查发现关联的 EBS 卷 ID 和分区号。此字段出现在使用 无代理扫描 扫描的 EC2 实例的调查发现中。
- 已安装版本/已修复版本 – 检测到脆弱性的当前已安装程序包的版本号。将已安装的版本号与斜杠 (/) 后的值进行比较。第二个值是修复了检测到的脆弱性的程序包版本号，该版本号由与此调查发现相关的常见脆弱性和风险 (CVE) 或公告提供。如果脆弱性已在多个版本中得到修复，则此字段将列出包含修复的最新版本。如果修复不可用，则此值为 `None available`。

  > **Note**
  >
  > 如果调查发现是在 Amazon Inspector 开始将此字段纳入调查发现之前检测到的，则此字段的值为空。不过，可能提供相应修复。

- 程序包管理器 – 用于配置此程序包的程序包管理器。
- 补救 – 如果可通过更新程序包或编程库进行修复，则此部分将包含可以运行的更新命令。您可以复制提供的命令并在环境中运行它。

  > **Note**
  >
  > 补救命令由供应商数据源提供，可能因系统配置而异。请查看调查发现参考资料或操作系统文档，获取更具体的指导。

- 脆弱性详细信息 – 针对调查发现中确定的 CVE，提供指向相应 Amazon Inspector 首选来源的链接，例如美国国家漏洞数据库 (NVD)、REDHAT 或其他操作系统供应商。此外，您还可以看到调查发现的严重性评分。有关严重性评分的更多信息，请参阅 Amazon Inspector 调查发现的严重性级别。提供以下评分，以及每项评分的评分向量：
  - EPSS 分数
  - Inspector 分数
  - 来自 Amazon CVE 的 CVSS 3.1
  - 来自 NVD 的 CVSS 3.1
  - 来自 NVD 的 CVSS 2.0（如适用，适用于较早的 CVE）
- 相关脆弱性 – 指定与调查发现相关的其他脆弱性。通常，这些是影响相同程序包版本的其他 CVE，或者与调查发现 CVE 属于同一组的其他 CVE，具体由供应商确定。

代码脆弱性

代码脆弱性调查发现仅适用于 Lambda 函数。有关更多信息，请参阅代码漏洞。此调查发现类型具有以下详细信息：

- 修复可用 – 对于代码脆弱性，此值始终为 YES。
- 检测器名称-用于 CodeGuru 检测代码漏洞的检测器的名称。有关可能检测的列表，请参阅探CodeGuru 测器库。
- 探测器标签-与探测器关联的标签 CodeGuru 使用标签对检测进行分类。 CodeGuru
- 相关的 CWE – 与代码脆弱性相关的常见缺陷枚举 (CWE) 的 ID。
- 文件路径 – 代码脆弱性的文件位置。
- 脆弱性位置 – 对于 Lambda 代码扫描代码脆弱性，此字段显示 Amazon Inspector 发现脆弱性的确切代码行。
- 建议的补救措施 – 这提供了如何编辑代码来补救调查发现的建议。

网络可达性

网络可达性调查发现仅适用于 EC2 实例。有关更多信息，请参阅网络可达性。此调查发现类型具有以下详细信息：

- 开放端口范围 – 可以访问 EC2 实例的端口范围。
- 开放网络路径 – 显示 EC2 实例的开放访问路径。选择路径上的项目可获取更多信息。
- 补救 – 推荐关闭开放网络路径的方法。

Amazon Inspector 分数和漏洞情报

在 Amazon Inspector 控制台中，选择一项调查发现时，可以查看 Inspector 分数和漏洞情报选项卡，这里显示了程序包漏洞调查发现的评分详情以及漏洞情报详细信息。这些详细信息仅适用于程序包漏洞调查发现。

Amazon Inspector 评分

Amazon Inspector 评分是 Amazon Inspector 为每个 EC2 实例调查发现创建的情境化评分。Amazon Inspector 评分是通过将基本的 CVSS v3.1 分数信息与扫描期间从计算环境中收集的信息（例如网络可达性结果和可利用性数据）相关联来确定的。例如，如果脆弱性可通过网络利用，但 Amazon Inspector 确定互联网上没有通往易受攻击实例的开放网络路径，则该调查发现的 Amazon Inspector 评分可能低于基础评分。

调查发现的基础评分是供应商提供的 CVSS v3.1 基础评分。我们支持 RHEL、Debian 或亚马逊供应商基础评分，对于其他供应商或者供应商未提供评分的情况，Amazon Inspector 使用美国国家漏洞数据库（NVD）中的基础评分。Amazon Inspector 使用通用漏洞评分系统 3.1 版计算器来计算评分。可以在漏洞详细信息下的调查发现详细信息中查看单个调查发现的基础评分来源，即漏洞源（或调查发现 JSON 中的 packageVulnerabilityDetails.source）。

> **Note**
>
> Amazon Inspector 评分不适用于运行 Ubuntu 的 Linux 实例。这是因为 Ubuntu 自行定义的漏洞严重性可能与相关的 CVE 严重性不同。

Amazon Inspector 评分详细信息

打开调查发现的详细信息页面后，您可以选择 Inspector 分数和漏洞情报选项卡。此面板显示基础评分与 Inspector 分数之间的差异。本节介绍 Amazon Inspector 如何根据程序包的 Amazon Inspector 评分和供应商评分的组合来分配严重性评级。如果评分不同，此面板会显示相应原因。

在 CVSS 分数指标部分，您可以看到一个表格，其中包含 CVSS 基础评分指标与 Inspector 分数的对比情况。参与对比的指标是 first.org 维护的 CVSS 规范文档中定义的基础指标。基础指标汇总如下：

攻击向量

可利用脆弱性的环境。对于 Amazon Inspector 调查发现，这可以是网络、相邻网络或本地。

攻击复杂性

这描述了攻击者在利用脆弱性时面临的难度级别。低评分意味着攻击者只需满足很少或根本不需要满足其他条件即可利用该脆弱性。高评分意味着攻击者需要投入大量精力才能成功利用此脆弱性进行攻击。

所需的权限

这描述了攻击者利用脆弱性所需的权限级别。

用户互动

该指标说明成功利用此脆弱性进行攻击是否需要除攻击者之外的其他真人用户。

范围

这说明一个易受攻击组件中的脆弱性是否会影响易受攻击组件安全范围以外的组件中的资源。如果此值为不变，则受影响的资源不会影响其他资源。如果此值为已更改，则表明可利用易受攻击的组件来影响由不同安全机构管理的资源。

机密性

这衡量当脆弱性被利用时，对资源内数据机密性的影响程度。其范围为从无（不影响机密性）到高（资源中的所有信息都会泄露，或者密码或加密密钥等机密信息会泄露）。

完整性

这衡量当脆弱性被利用时，对受影响资源内数据完整性的影响程度。当攻击者修改受影响资源内的文件时，完整性就会受到威胁。评分范围为从无（即攻击者无法通过脆弱性修改任何信息）到高（如果脆弱性被利用，攻击者将可以修改任意或所有文件，或者可能被修改的文件会产生严重后果）。

可用性

这衡量当脆弱性被利用时，对受影响资源可用性的影响程度。评分范围为从无（脆弱性完全不影响可用性）到高（如果脆弱性被利用，攻击者可以完全拒绝对资源的访问或导致服务不可用）。

脆弱性情报

本节总结了 Amazon 提供的有关 CVE 的可用情报以及行业标准的安全情报来源，例如 Recorded Future 以及美国网络安全与基础设施安全局 (CISA)。

> **Note**
>
> 来自 CISA、亚马逊或 Recorded Future 的情报并非适用于所有 CVE。

可以在控制台中或使用 BatchGetFindingDetails API 查看漏洞情报详细信息。控制台提供以下详细信息：

ATT&CK

此部分显示与 CVE 相关联的 MITRE 战术、技术和程序 (TTP)。关联的 TTP 都会显示，如果有两个以上适用的 TTP，则可以选择链接以查看完整列表。选择一种战术或技术，可在 MITRE 网站上打开有关相应战术或技术的信息。

CISA

此部分包含与脆弱性相关联的日期。美国网络安全和基础设施安全局 (CISA) 根据活动利用情况的证据，将脆弱性添加到已知被利用的脆弱性目录中的日期，以及 CISA 预计系统修复脆弱性的截止日期。此信息来自 CISA。

已知恶意软件

此部分列出了利用此漏洞的已知利用包和工具。

证据

此部分总结了涉及此脆弱性的最严重安全事件。如果有 3 个以上的事件具有相同的严重性级别，则会显示最近的三个事件。

上次报告时间

此部分显示脆弱性的最后一次已知公开利用日期。

Amazon Inspector 调查发现的严重性级别

当 Amazon Inspector 生成脆弱性调查发现时，它会自动为调查发现指定严重性。调查发现的严重性反映了它的主要特征，因此可以帮助您评测调查发现并确定其优先级。调查发现的严重性并不意味着或以其他方式表明受影响的资源可能对您的组织具有的关键程度或重要性。

调查发现的严重性评级由与以下严重性级别之一相对应的数字评分决定：提醒、低、中、高或严重。

Amazon Inspector 确定严重性的方法因调查发现类型而异。请参阅以下章节，详细了解 Amazon Inspector 如何确定每种调查发现类型的严重性评级。

程序包脆弱性严重性

Amazon Inspector 使用 NVD/CVSS 评分作为程序包脆弱性严重性评分的基础。NVD/CVSS 评分是 NVD 发布并由 CVSS 定义的脆弱性严重性评分。NVD/CVSS 评分由攻击复杂性、脆弱性代码成熟度和所需权限等安全指标组成。Amazon Inspector 会生成一个从 1 到 10 的数字评分，以反映脆弱性的严重性。Amazon Inspector 将其归类为基础评分，因为它根据脆弱性的内在特征反映了脆弱性的严重性，而这些特征不会随着时间的推移改变。该评分还假定了不同部署环境中合理的最坏影响。CVSS v3 标准将 CVSS 评分对应以下严重性评级。

评分	评级
0	Informational
0.1–3.9	Low
4.0–6.9	Medium

7.0–8.9	High
9.0–10.0	Critical

程序包脆弱性调查发现的严重性也可能是未分类。这意味着供应商尚未为检测到的脆弱性设置脆弱性评分。在这种情况下，我们建议使用调查发现的参考 URL 来研究脆弱性并做出相应的响应。

程序包脆弱性调查发现包括以下评分和相关的评分向量，这些都属于调查发现的详细信息：

- EPSS 分数
- Inspector 分数
- 来自 Amazon CVE 的 CVSS 3.1
- 来自 NVD 的 CVSS 3.1
- 来自 NVD 的 CVSS 2.0（如适用）

代码脆弱性严重性

对于代码漏洞发现，Amazon Inspector 使用生成该发现的亚马逊 CodeGuru 探测器定义的严重性级别。使用 CVSS v3 评分系统为每个检测器分配一个严重性。有关严重性 CodeGuru 使用的说明，请参阅 CodeGuru 指南中的严重性定义。要查看按严重性分列的检测器列表，请从以下支持的编程语言中进行选择：

- 按严重性划分的 Python 检测器
- 按严重性划分的 Java 检测器

网络可达性严重性

Amazon Inspector 根据暴露的服务、端口和协议以及开放路径的类型来确定网络可达性脆弱性的严重性。下表定义了这些严重性评级。开放路径评级列中的值表示来自虚拟网关、对等 VPC 和 AWS Direct Connect 网络的开放路径。所有其他暴露的服务、端口和协议的严重性评级均为“提醒”。

服务	TCP 端口	UDP 端口	互联网路径评级	开放路径评级
DHCP	67, 68, 546, 547	67, 68, 546, 547	Medium	Informational
Elasticsearch	9300, 9200	NA	Medium	Informational

FTP	21	21	High	Medium
Global catalog LDAP	3268	NA	Medium	Informational
Global catalog LDAP over TLS	3269	NA	Medium	Informational
HTTP	80	80	Low	Informational
HTTPS	443	443	Low	Informational
Kerberos	88, 464, 543, 544, 749, 751	88, 464, 749, 750, 751, 752	Medium	Informational
LDAP	389	389	Medium	Informational
LDAP over TLS	636	NA	Medium	Informational
MongoDB	27017, 27018, 27019, 28017	NA	Medium	Informational
MySQL	3306	NA	Medium	Informational
NetBIOS	137, 139	137, 138	Medium	Informational
NFS	111, 2049, 4045, 1110	111, 2049, 4045, 1110	Medium	Informational
Oracle	1521, 1630	NA	Medium	Informational
PostgreSQL	5432	NA	Medium	Informational
Print services	515	NA	High	Medium
RDP	3389	3389	Medium	Low
RPC	111, 135, 530	111, 135, 530	Medium	Informational
SMB	445	445	Medium	Informational
SSH	22	22	Medium	Low

SQL Server	1433	1434	Medium	Informational
Syslog	601	514	Medium	Informational
Telnet	23	23	High	Medium
WINS	1512, 42	1512, 42	Medium	Informational

管理 Amazon Inspector 中的调查发现

Amazon Inspector 提供了多种对调查发现进行排序、分组和管理的方法。这些功能可帮助您根据自己的环境量身定制调查结果，通过不同的视图汇总调查结果，并重点关注特定 AWS 环境的漏洞。

调查发现根据其状态显示在不同的视图中：活动、隐藏或已关闭。默认情况下，每个视图仅显示活动的调查发现。活动的调查发现表示 Amazon Inspector 检测到的潜在安全问题，表明存在脆弱性或潜在威胁。隐藏的调查发现是使用抑制规则排除的活动调查发现。当 Amazon Inspector 检测到调查发现已修复时，它会自动将该调查发现的状态设置为已关闭。不需要手动关闭调查发现。

您还可以在中 AWS Security Hub查看调查结果，该服务可全面了解您的 AWS 环境中的安全状态。有关更多信息，请参阅Amazon Inspector 与 AWS Security Hub 集成。Amazon ECR 控制台中还提供了容器映像调查结果，您可以使用 AWS Command Line Interface (AWS CLI) 或 API 查看所有资源的调查结果。

主题

查看 Amazon Inspector 调查发现

Amazon Inspector 控制台根据相关分组在选项卡视图中显示调查发现。每个视图都包含可帮助您分析特定脆弱性、识别最脆弱的资源以及衡量环境中脆弱性的总体影响的信息。您可以通过在调查发现导航侧面板下选择相应选项来导航到不同的调查发现视图。您还可以在每个视图中创建筛选条件，以重点关注特定类型的调查发现。有关使用筛选条件的更多信息，请参阅筛选 Amazon Inspector 调查发现。

调查发现可以按以下参数进行分组：

- 按脆弱性 – 列出在您的环境中检测到的最严重的脆弱性。在该视图中选择一个脆弱性标题，即可打开包含更多信息的详细信息窗格。
- 按账户 – 列出您的账户、每个账户的 Amazon Inspector 扫描覆盖率以及每个账户的重大和高严重性的调查发现总数。只有委托管理员才能使用该分组。
- 按实例 – 列出环境中最脆弱的 Amazon EC2 实例。

- 按容器映像 – 列出环境中最脆弱的 Amazon ECR 容器映像。
- 按容器存储库 – 显示脆弱性最多的存储库。
- 按 Lambda 函数 – 显示脆弱性最多的 Lambda 函数。
- 所有调查发现 – 显示您环境的调查发现的完整列表。这是导航到调查发现页面时的默认视图。在此视图中，您可以按活动、隐藏和已关闭的调查发现进行筛选。

您可以根据筛选条件创建抑制规则，将部分调查发现排除在调查发现视图之外。有关更多信息，请参阅使用抑制规则抑制 Amazon Inspector 调查发现。

筛选 Amazon Inspector 调查发现

借助调查发现筛选条件，您可以只查看符合您指定标准的调查发现。不符合筛选条件的调查发现将从您的视图中排除。您可以使用 Amazon Inspector 控制台创建调查发现筛选条件。要使用这些筛选条件自动隐藏现有和未来的调查发现，请参阅使用抑制规则抑制 Amazon Inspector 调查发现。

在 Amazon Inspector 控制台中创建筛选条件

在每个调查发现视图中，您都可以使用筛选功能来查找具有特定特征的调查发现。移至其他选项卡视图时，筛选条件将被移除。

筛选条件包含一个条件，其中包括配对的筛选属性与筛选值。不符合筛选条件的调查发现将从调查发现列表中排除。例如，要查看与您的管理员帐户关联的所有结果，您可以选择 AWS 账户 ID 属性并将其与您的十二位数 AWS 账户 ID 的值配对。

有些筛选条件适用于所有调查发现，而有些筛选条件仅适用于特定的资源类型或调查发现类型。

对调查发现视图应用筛选条件

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 在导航窗格中，选择调查发现。默认视图会显示所有状态为活动的调查发现。
3. 要按条件筛选调查发现，请选择添加筛选条件栏，查看该视图适用的所有筛选条件的列表。在不同的视图中可使用的筛选条件不同。
4. 从列表中选择您要筛选的条件。
5. 在标准输入窗格中输入所需的筛选值以定义条件。
6. 选择应用，将该筛选条件应用于当前结果。您可以再次选择筛选条件输入栏，继续添加其他筛选条件。

7. （可选）要查看隐藏或已关闭的调查发现，请在筛选条件栏中选择活动，然后选择隐藏或已关闭。选择显示全部，可在同一个视图中查看活动、隐藏和已关闭的调查发现。

使用抑制规则抑制 Amazon Inspector 调查发现

使用抑制规则排除符合标准的搜索结果。例如，您可以创建一条规则，抑制所有漏洞分数较低的发现，这样您就可以只关注最关键的发现。

> **Note**
>
> 禁止规则仅用于筛选您的调查结果列表，不会对调查结果产生任何影响，也不会阻止 Amazon Inspector 生成调查结果。

如果 Amazon Inspector 生成的调查结果与禁止规则相匹配，则结果将设置为 “已隐藏”。默认情况下，符合禁止规则的搜索结果不会出现在您的列表中。

Amazon Inspector 会存储隐藏的调查结果，直到它们得到补救。Amazon Inspector 检测到已修正的调查结果。当 Amazon Inspector 检测到已修正的发现时，它会将该发现设置为 “已关闭”，并将其存储 7 天。

禁止显示的调查结果将 EventBridge 作为事件发布到 Amazon AWS Security Hub 和 Amazon。通过使用 EventBridge 规则更改查找结果的状态，可以自动抑制 Security Hub 中不需要的发现。有关更多信息，请参阅中的如何创建自动禁止规则。 AWS Security Hub

您无法创建关闭或修复发现结果的抑制规则。您只能创建抑制规则来筛选列表中显示的结果。您可以随时在 Amazon Inspector 控制台中查看隐藏的调查发现。

> **Note**
>
> 组织中的成员账户无法创建或管理禁止规则。

创建抑制规则

您可以创建抑制规则来筛选默认显示的调查发现列表。您可以使用 CreateFilterAPI 并指定SUPRESS为的值，以编程方式创建禁止规则。`action`

> **Note**
>
> 只有独立账户和 Amazon Inspector 授权的管理员才能创建和管理抑制规则。组织中的成员不会在导航窗格中看到抑制规则的选项。

要创建抑制规则（控制台）

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 在导航窗格中，选择抑制规则。然后，选择创建规则。
3. 对于每个条件，请执行以下操作：
   - 选择筛选栏以查看可以添加到抑制规则中的筛选条件的列表。
   - 为您的抑制规则选择筛选条件。
4. 添加完条件后，输入规则的名称，还可选择输入相应描述。
5. 选择保存规则。Amazon Inspector 会立即应用新的抑制规则，并隐藏所有符合条件的调查发现。

查看隐藏的调查发现

默认情况下，Amazon Inspector 不会在 Amazon Inspector 控制台中显示隐藏的调查发现。不过，您可以查看特定规则抑制的调查发现。

要查看隐藏的调查发现

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 在导航窗格中，选择抑制规则。
3. 在抑制规则列表中，选择规则的标题。

更改抑制规则

您可以随时更改抑制规则。

要修改抑制规则

1. 打开 Amazon Inspector 控制台：https://console.aws.amazon.com/inspector/v2/home
2. 在导航窗格中，选择抑制规则。

3. 选择要修改的抑制规则的标题。
4. 进行所需的更改，然后选择保存以更新规则。

删除抑制规则

抑制规则可以删除。删除抑制规则后，Amazon Inspector 将不再隐藏符合规则标准且未被其他规则抑制的新调查发现和现有调查发现。

删除抑制规则后，符合该规则条件的新调查发现和现有调查发现的状态均变为活动。这意味着它们默认显示在 Amazon Inspector 控制台中。此外，Amazon Inspector 还会将这些发现 EventBridge 作为事件发布 AWS 给 Security Hub 和亚马逊。

要删除抑制规则

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 在导航窗格中，选择抑制规则。
3. 选中要删除的抑制规则标题旁边的复选框。
4. 选择删除，然后确认您的选择以永久删除规则。

从 Amazon Inspector 导出调查发现报告

除了向 Amazon EventBridge 和 AWS Security Hub 发送调查发现外，您还可以选择将调查发现以调查发现报告的形式，导出到 Amazon Simple Storage Service (Amazon S3) 存储桶。调查发现报告是一个 CSV 或 JSON 文件，其中包含您选择包含在报告中的调查发现的详细信息。它提供了特定时间点调查发现的详细快照。对于每项调查发现，该文件都包含各种详细信息，例如受影响资源的 Amazon Resource Name (ARN)、调查发现的创建日期和时间、相关的常见漏洞和风险 (CVE) ID 以及调查发现的严重性、状态以及 Amazon Inspector 和 CVSS 分数。

配置调查发现报告时，首先要指定要在报告中包含哪些调查发现。默认情况下，Amazon Inspector 包含当前 AWS 区域内状态为有效的所有调查发现的数据。如果您是某组织的 Amazon Inspector 委托管理员，这还包括贵组织中所有成员账户的调查发现数据。

您可以选择通过筛选数据来自定义报告。使用筛选条件时，您可以添加或排除具有特定特征的调查发现的数据，例如，在特定时间范围内创建的所有关键调查发现、特定资源的所有活动调查发现或特定类型的所有关键调查发现。如果您是某组织的 Amazon Inspector 管理员，则可以使用筛选条件创建包含组织中特定 AWS 账户的调查发现的报告，例如，某账户中所有状态为活动且已有相应修复的关键调查发现。然后，您可以与账户所有者共享报告以进行修复。

> **Note**
>
> 使用 CreateFindingsReport API 导出调查发现报告时，默认情况下您只会看到活动的调查发现。要查看已隐藏或已关闭的调查发现，必须将 findingStatus 筛选条件的值指定为 SUPPRESSED 或 CLOSED。

导出调查发现报告时，Amazon Inspector 会使用您指定的 AWS Key Management Service (AWS KMS) 密钥对数据进行加密，并将该报告添加到您指定的 S3 存储桶中。加密密钥必须是客户管理的、位于当前 AWS 区域的 AWS Key Management Service (AWS KMS) 对称加密密钥。此外，密钥政策必须允许 Amazon Inspector 使用该密钥。S3 存储桶还必须位于当前区域，并且存储桶的策略必须允许 Amazon Inspector 向存储桶添加对象。

在 Amazon Inspector 完成对您的报告的加密和存储后，您可以从您指定的 S3 存储桶下载报告或将其移动到其他位置。或者，您可以将报告保留在这一 S3 存储桶中，并将该存储桶用作随后导出的调查发现报告的存储库。

本主题将指导您完成使用 AWS Management Console 导出调查发现报告的过程。该过程包括验证您是否拥有所需的权限，配置所需的资源，然后配置和导出报告。

> **Note**
>
> 每次只能导出一份调查发现报告。如果当前正在导出报告，请等到导出完成后再尝试导出其他报告。

任务

- 步骤 1：验证您的权限
- 步骤 2：配置 S3 存储桶
- 步骤 3：配置 AWS KMS key
- 步骤 4：配置和导出调查发现报告
- 排查导出错误

首次导出调查发现报告后，步骤 1–3 则成为可选的。这主要取决于您是否要将相同的 S3 存储桶和 AWS KMS key 用于后续报告。

如果您更喜欢在步骤 1-3 之后以编程方式导出报告，请使用 Amazon Inspector API 的 CreateFindingsReport 操作。

步骤 1：验证您的权限

在从 Amazon Inspector 导出调查发现报告之前，请确认您拥有导出调查发现报告以及配置用于加密和存储报告的资源所需的权限。要验证权限，请使用 AWS Identity and Access Management (IAM) 查看附加到您的 IAM 身份的 IAM policy。然后，将这些策略中的信息与以下导出调查发现报告时您需要执行的操作的列表进行比较。

Amazon Inspector

对于 Amazon Inspector，请确认您可以执行以下操作：

- `inspector2:ListFindings`
- `inspector2:CreateFindingsReport`

这些操作允许您检索账户的调查发现数据，并将这些数据导出到调查发现报告中。

如果您计划以编程方式导出大型报告，则还可以验证您是否可以执行以下操作：`inspector2:GetFindingsReportStatus`，用于检查报告的状态；`inspector2:CancelFindingsReport`，用于取消正在进行的导出。

AWS KMS

对于 AWS KMS，请验证您是否可以执行以下操作：

- `kms:GetKeyPolicy`
- `kms:PutKeyPolicy`

这些操作允许您检索和更新您希望 Amazon Inspector 用来加密报告的 AWS KMS key 密钥策略。

要使用 Amazon Inspector 控制台导出报告，还要确认您是否可以执行以下 AWS KMS 操作：

- `kms:DescribeKey`
- `kms:ListAliases`

这些操作允许您检索和显示有关您账户的 AWS KMS keys 的信息。然后，您可以选择其中一种密钥来加密报告。

如果您计划创建新的 KMS 密钥来加密报告，则还需要能够执行 `kms:CreateKey` 操作。

Amazon S3

对于 Amazon S3，请验证您是否可以执行以下操作：

- s3:CreateBucket
- s3:DeleteObject
- s3:PutBucketAcl
- s3:PutBucketPolicy
- s3:PutBucketPublicAccessBlock
- s3:PutObject
- s3:PutObjectAcl

这些操作允许您创建和配置 Amazon Inspector 用于存储报告的 S3 存储桶。它们还允许您在存储桶中添加和删除对象。

如果要使用 Amazon Inspector 控制台导出报告，还需要验证您是否可以执行 s3:ListAllMyBuckets 和 s3:GetBucketLocation 操作。这些操作允许您检索和显示有关您账户的 S3 存储桶的信息。然后，您可以选择其中一个存储桶来存储报告。

如果您无法执行一项或多项必要的操作，请在继续下一步之前向 AWS 管理员寻求帮助。

步骤 2：配置 S3 存储桶

验证权限后，就可以配置用于存储调查发现报告的 S3 存储桶了。它可以是您自己账户的现有存储桶，也可以是其他 AWS 账户拥有且允许您访问的现有存储桶。如果您想将报告存储在新存储桶中，请在继续操作之前创建存储桶。

S3 存储桶必须与您要导出的调查发现数据位于同一 AWS 区域。例如，如果您在美国东部（弗吉尼亚州北部）区域使用 Amazon Inspector，并且想要导出该区域的调查发现数据，则存储桶也必须位于美国东部（弗吉尼亚州北部）区域。

此外，存储桶的策略必须允许 Amazon Inspector 向存储桶添加对象。本主题说明了如何更新存储桶策略，并提供了要添加到策略中的语句的示例。有关添加和更新存储桶策略的详细信息，请参阅《Amazon Simple Storage Service 用户指南》中的使用存储桶策略。

如果您想将报告存储在其他账户拥有的 S3 存储桶中，请与该存储桶的所有者合作，更新存储桶策略。同时还要获取存储桶的 URI。导出报告时，需要输入此 URI。

更新存储桶策略

1. 通过以下网址打开 Amazon S3 控制台：https://console.aws.amazon.com/s3。

2. 在导航窗格中，选择桶。
3. 选择要存储调查发现报告的 S3 存储桶。
4. 选择权限选项卡。
5. 在存储桶策略部分中，选择编辑。
6. 将以下示例语句复制到剪贴板：

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "allow-inspector",
   "Effect": "Allow",
   "Principal": {
    "Service": "inspector2.amazonaws.com"
   },
   "Action": [
    "s3:PutObject",
    "s3:PutObjectAcl",
    "s3:AbortMultipartUpload"
   ],
   "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*",
   "Condition": {
    "StringEquals": {
     "aws:SourceAccount": "111122223333"
    },
    "ArnLike": {
     "aws:SourceArn": "arn:aws:inspector2:Region:111122223333:report/*"
    }
   }
  }
 ]
}
```

7. 在 Amazon S3 控制台的存储桶策略编辑器中，将上述语句粘贴到策略中，以将其添加到策略中。

   添加语句时，请确保语法有效。存储桶策略使用 JSON 格式。这意味着需要在语句之前或之后添加一个逗号，具体取决于在策略中添加语句的位置。如果将语句添加在最后，请在前一语句的右大括号后面添加一个逗号。如果将语句添加为第一个语句，或添加在两个现有语句之间，请在语句的右大括号后面添加一个逗号。
8. 使用适合您环境的正确值更新语句，其中：

- *DOC-EXAMPLE-BUCKET* 是存储桶的名称。
- *111122223333* 是您的 AWS 账户的账户 ID。
- *##*是指您使用 Amazon Inspector 并希望允许 Amazon Inspector 向存储桶添加报告的 AWS 区域。例如，us-east-1 表示美国东部（弗吉尼亚州北部）区域。

> **Note**
>
> 如果您在手动启用的 AWS 区域使用 Amazon Inspector，还需要在 Service 字段的值中添加相应的区域代码。此字段指定了 Amazon Inspector 的服务主体。
> 例如，如果您在中东（巴林）区域使用 Amazon Inspector，该区域的区域代码为 me-south-1，则需要将语句中的 inspector2.amazonaws.com 替换为 inspector2.me-south-1.amazonaws.com。

请注意，示例语句定义了使用两个 IAM 全局条件键的条件：

- aws:SourceAccount – 此条件仅允许 Amazon Inspector 为您的账户向存储桶添加报告。它可以防止 Amazon Inspector 为其他账户向存储桶中添加报告。更具体地说，该条件指定了哪个账户可以将存储桶用于 aws:SourceArn 条件指定的资源和操作。

  要为其他账户在存储桶中存储报告，请在此条件中添加其他每个账户的账户 ID。例如：

  ```
  "aws:SourceAccount": [111122223333,444455556666,123456789012]
  ```

- aws:SourceArn – 此条件根据要添加到存储桶中的对象的来源，限制对存储桶的访问权限。它可以防止其他 AWS 服务向存储桶添加对象。它还可以防止 Amazon Inspector 在为您的账户执行其他操作时向存储桶添加对象。更具体地说，只有当对象是调查发现报告，并且这些报告由条件中指定的账户在条件中指定的区域创建时，该条件才允许 Amazon Inspector 向存储桶添加对象。

  要允许 Amazon Inspector 为其他账户执行指定操作，请在此条件中添加其他每个账户的 Amazon 资源名称 (ARN)。例如：

  ```
  "aws:SourceArn": [
      "arn:aws:inspector2:Region:111122223333:report/*",
      "arn:aws:inspector2:Region:444455556666:report/*",
      "arn:aws:inspector2:Region:123456789012:report/*"
  ```

```
]
```

aws:SourceAccount 和 aws:SourceArn 条件指定的账户应匹配。

这两个条件都有助于防止 Amazon Inspector 在 Amazon S3 事务期间被用作混淆代理。您可以从存储桶策略中删除这些条件，但我们并不建议您这样做。

9. 完成存储桶策略更新后，选择保存更改。

步骤 3：配置 AWS KMS key

验证权限并配置 S3 存储桶后，应确定AWS KMS key您希望 Amazon Inspector 用来加密调查发现报告的 。密钥必须是客户管理的对称加密 KMS 密钥。此外，密钥必须与您配置用于存储报告的 S3 存储桶位于同一 AWS 区域。

密钥可以是您自己账户中的现有 KMS 密钥，也可以是其他账户拥有的现有 KMS 密钥。如果要使用新的 KMS 密钥，请在继续之前创建密钥。如果您希望使用其他账户拥有的现有密钥，请获取该密钥的 Amazon 资源名称 (ARN)。从 Amazon Inspector 中导出报告时，需要输入此 ARN。有关创建和查看 KMS 密钥设置的信息，请参阅 AWS Key Management Service 开发人员指南中的管理密钥。

确定要使用哪个 KMS 密钥后，向 Amazon Inspector 授予使用该密钥的权限。否则，Amazon Inspector 将无法加密和导出报告。要授予 Amazon Inspector 使用密钥的权限，请更新密钥的密钥策略。有关密钥策略和管理 KMS 密钥访问权限的详细信息，请参阅 AWS Key Management Service 开发人员指南中的 AWS KMS 密钥策略。

更新密钥策略

> **Note**
>
> 以下步骤用于更新现有密钥以允许 Amazon Inspector 使用它。如果您还没有现有得密钥，请参阅https://docs.aws.amazon.com/kms/latest/developerguide/create-keys.html，获取有关创建密钥的指南。

1. 从 https://console.aws.amazon.com/kms 打开 AWS KMS 控制台。
2. 要更改 AWS 区域，请使用页面右上角的区域选择器。
3. 在导航窗格中，选择客户托管密钥。
4. 选择要用于加密报告的 KMS 密钥。该密钥必须为对称加密 (SYMMETRIC_DEFAULT) 密钥。

5. 在密钥策略选项卡上，选择编辑。如果您没有看到带有编辑按钮的密钥策略，则必须先选择切换到策略视图。
6. 将以下示例语句复制到剪贴板：

```
{
    "Sid": "Allow Amazon Inspector to use the key",
    "Effect": "Allow",
    "Principal": {
        "Service": "inspector2.amazonaws.com"
    },
    "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey*"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "111122223333"
        },
        "ArnLike": {
            "aws:SourceArn": "arn:aws:inspector2:Region:111122223333:report/*"
        }
    }
}
```

7. 在 AWS KMS 控制台的密钥策略编辑器中，将上述语句粘贴到密钥策略中，以将其添加到策略中。

   添加语句时，请确保语法有效。密钥策略使用 JSON 格式。这意味着需要在语句之前或之后添加一个逗号，具体取决于在策略中添加语句的位置。如果将语句添加在最后，请在前一语句的右大括号后面添加一个逗号。如果将语句添加为第一个语句，或添加在两个现有语句之间，请在语句的右大括号后面添加一个逗号。
8. 使用适合您环境的正确值更新语句，其中：

   - *111122223333* 是您的 AWS 账户的账户 ID。
   - *##*是指您希望允许 Amazon Inspector 使用密钥加密报告的 AWS 区域。例如，us-east-1 表示美国东部（弗吉尼亚州北部）区域。

> **Note**
>
> 如果您在手动启用的 AWS 区域使用 Amazon Inspector，还需要在 `Service` 字段的值中添加相应的区域代码。例如，如果您在中东（巴林）区域使用 Amazon Inspector，请将 `inspector2.amazonaws.com` 替换为 `inspector2.me-south-1.amazonaws.com`。

与上一步中存储桶策略的示例语句一样，此示例中的 `Condition` 字段也使用两个 IAM 全局条件键：

- aws:SourceAccount – 此条件仅允许 Amazon Inspector 为您的账户执行指定的操作。更具体地说，它决定了哪个账户可以为 `aws:SourceArn` 条件指定的资源和操作执行指定的操作。

  要允许 Amazon Inspector 为其他账户执行指定操作，请在此条件中添加其他每个账户的账户 ID。例如：

  ```
  "aws:SourceAccount": [111122223333,444455556666,123456789012]
  ```

- aws:SourceArn – 此条件可防止其他 AWS 服务执行指定的操作。它还可以防止 Amazon Inspector 在为您的账户执行其他操作时使用密钥。换句话说，只有当对象是调查发现报告，并且这些报告由条件中指定的账户在条件中指定的区域创建时，该条件才允许 Amazon Inspector 使用密钥加密 S3 对象。

  要允许 Amazon Inspector 为其他账户执行指定操作，请在此条件中添加其他每个账户的 ARN。例如：

  ```
  "aws:SourceArn": [
      "arn:aws:inspector2:us-east-1:111122223333:report/*",
      "arn:aws:inspector2:us-east-1:444455556666:report/*",
      "arn:aws:inspector2:us-east-1:123456789012:report/*"
  ]
  ```

  `aws:SourceAccount` 和 `aws:SourceArn` 条件指定的账户应匹配。

这些条件有助于防止 Amazon Inspector 在 AWS KMS 事务期间被用作混淆代理。您可以从语句中删除这些条件，但我们并不建议您这样做。

9. 完成密钥策略更新后，选择保存更改。

步骤 4：配置和导出调查发现报告

验证权限并配置资源以加密和存储调查发现报告后，就可以配置和导出报告了。

要配置和导出调查发现报告

1. 打开 Amazon Inspector 控制台：https://console.aws.amazon.com/inspector/v2/home。
2. 在导航窗格中的调查发现下，选择所有调查发现。
3. （可选）使用调查发现表上方的筛选栏，添加筛选条件，指定要在报告中包含哪些调查发现。添加条件时，Amazon Inspector 会更新表格，使其仅包含符合条件的调查发现。该表提供了报告所含数据的预览。

   > **Note**
   >
   > 建议添加筛选条件。如果不这样做，则该报告将包含当前 AWS 区域中状态为活动的所有调查发现的数据。如果您是某组织的 Amazon Inspector 管理员，则这将包含贵组织中所有成员账户的调查发现数据。
   > 如果报告包含所有或多个调查发现的数据，则生成和导出报告可能需要很长时间，而且一次只能导出一份报告。

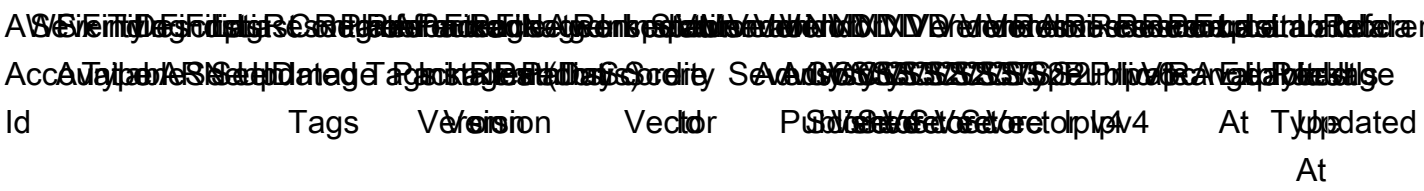
4. 选择导出调查发现。
5. 在导出设置部分的导出文件类型中，指定报告的文件格式：

   - 要创建包含相应数据的 JavaScript 对象表示法 (.json) 文件，请选择 JSON。

     如果您选择 JSON 选项，则报告将包含每个调查发现的所有字段。有关可能的 JSON 字段的列表，请参阅 Amazon Inspector API 参考中的调查发现数据类型。

   - 要创建包含相应数据的逗号分隔值 (.csv) 文件，请选择 CSV。

     如果您选择 CSV 选项，则报告将仅包含每个调查发现的部分字段，即报告调查发现关键属性的大约 45 个字段。这些字段包括：调查发现类型、标题、严重性、状态、描述、首次查看、上次查看、修复可用、AWS 账户 ID、资源 ID、资源标签和补救措施。除此之外，还有一些字段可以捕获每个调查发现的评分详细信息和参考 URL。以下是调查发现报告中 CSV 标题的示例：



6. 在导出位置下，针对 S3 URI，指定要存储报告的 S3 存储桶：

   - 要将报告存储在您的账户拥有的存储桶中，请选择浏览 S3。Amazon Inspector 会列出您的账户的 S3 存储桶。选择所需的存储桶所在的行，然后单击选择。

     > **Tip**
     >
     > 要同时为报告指定 Amazon S3 路径前缀，请在 S3 URI 框中的值后面附加斜杠 (/) 和前缀。然后，Amazon Inspector 会在将报告添加到存储桶时添加前缀，Amazon S3 会生成由该前缀指定的路径。
     > 例如，如果您想使用自己的 AWS 账户 ID 作为前缀，并且您的账户 ID 为 111122223333，请在 S3 URI 框中的值后面附加 **/111122223333**。
     > 前缀类似于 S3 存储桶内的目录路径。它使您可以将相似的对象组合到一个存储桶中，就像将相似文件一起存储在文件系统的一个文件夹中一样。有关详细信息，请参阅 Amazon Simple Storage Service 用户指南中的使用文件夹在 Amazon S3 控制台中组织对象。

   - 要将报告存储在其他账户拥有的存储桶中，请输入相应存储桶的 URI，例如 **s3://DOC-EXAMPLE_BUCKET**，其中 DOC-EXAMPLE_BUCKET 是存储桶的名称。存储桶所有者可以在存储桶属性中帮您找到这些信息。

7. 对于 KMS 密钥，请指定要用于加密报告的 AWS KMS key：

   - 要使用自己账户中的密钥，请从列表中选择相应密钥。该列表显示了您账户的客户托管对称加密 KMS 密钥。
   - 要使用其他账户拥有的密钥，请输入相应密钥的 Amazon 资源名称 (ARN)。密钥所有者可以在密钥属性中帮您找到这些信息。有关详细信息，请参阅 AWS Key Management Service 开发人员指南中的查找密钥 ID 和 ARN。

8. 选择导出。

Amazon Inspector 生成调查发现报告，使用您指定的 KMS 密钥对其进行加密，然后将其添加到您指定的 S3 存储桶中。根据您选择在报告中包含的调查发现数量，此过程可能需要几分钟或几小时。导出

完成后，Amazon Inspector 会显示一条消息，表明您的调查发现报告已成功导出。（可选）在消息中选择查看报告，可导航到 Amazon S3 中的报告。

请注意，每次只能导出一份报告。如果当前正在导出报告，请等到导出完成后再尝试导出其他报告。

排查导出错误

如果导出调查发现报告时出现错误，Amazon Inspector 会显示一条描述错误的消息。您可以使用本主题中的信息作为指南，找出错误的可能原因和解决方案。

例如，验证 S3 存储桶位于当前 AWS 区域，并且存储桶的策略允许 Amazon Inspector 向存储桶添加对象。此外，验证当前区域已启用 AWS KMS key，并确保密钥策略允许 Amazon Inspector 使用密钥。

修复错误后，请尝试再次导出报告。

不能有多个报告错误

如果您要创建报告时，Amazon Inspector 已经在生成报告，则会收到一条错误消息，其内容为原因：无法同时处理多个报告。之所以出现此错误，是因为 Amazon Inspector 每次只能为一个账户生成一份报告。

要解决错误，您可以等待其他报告完成或取消报告，然后再请求新报告。

您可以使用 GetFindingsReportStatus 操作检查报告的状态，该操作会返回当前正在生成的报告的报告 ID。

如果需要，可以使用 `GetFindingsReportStatus` 操作提供的报告 ID 取消当前正在进行的导出，方法是使用 CancelFindingsReport 操作。

使用 Amazon EventBridge 创建对 Amazon Inspector 调查发现的自定义响应

Amazon Inspector 会针对新生成的调查发现、新汇总的调查发现以及调查发现状态的变化，为 Amazon EventBridge 创建一个事件。除了对 `updatedAt` 和 `lastObservedAt` 字段进行更改之外，其他任何内容更改都将发布新事件。这意味着，当您采取诸如重启资源或更改与资源关联的标签之类的操作时，就会生成针对调查发现的新事件。但是，`id` 字段中的调查发现 ID 保持不变。尽最大努力发布事件。

> **Note**
>
> 如果您的账户是 Amazon Inspector 委托管理员，EventBridge 除了会将事件发布到事件产生的成员账户外，还会将事件发布到您的账户。

将 EventBridge 事件与 Amazon Inspector 结合使用时，您可以自动执行任务，帮助您响应通过 Amazon Inspector 调查发现揭示的安全问题。

Amazon Inspector 会向同一地区的默认事件总线发送事件。这意味着您必须为运行 Amazon Inspector 的每个区域配置事件规则，才能查看该区域的事件。

要接收有关基于 EventBridge 事件的 Amazon Inspector 调查发现的通知，您必须为 Amazon Inspector 创建 EventBridge 规则和目标。此规则允许 EventBridge 将 Amazon Inspector 生成的调查发现的通知发送到在规则中指定的目标。有关更多信息，请参阅 Amazon EventBridge 用户指南中的 Amazon EventBridge 规则。

事件架构

以下是 EC2 调查发现事件的 Amazon Inspector 事件格式示例。有关其他调查发现类型和事件类型的示例架构，请参阅EventBridge 架构。


```
{
    "version": "0",
    "id": "66a7a279-5f92-971c-6d3e-c92da0950992",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-19T22:46:15Z",
    "region": "us-east-1",
    "resources": ["i-0c2a343f1948d5205"],
    "detail": {
        "awsAccountId": "111122223333",
        "description": "\n It was discovered that the sound subsystem in the Linux
 kernel contained a\n race condition in some situations. A local attacker could use
 this to cause\n a denial of service (system crash).",
        "exploitAvailable": "YES",
        "exploitabilityDetails": {
            "lastKnownExploitAt": "Oct 24, 2022, 11:08:59 PM"
        },
        "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/FINDING_ID",
```

```
        "firstObservedAt": "Jan 19, 2023, 10:46:15 PM",
        "fixAvailable": "YES",
        "lastObservedAt": "Jan 19, 2023, 10:46:15 PM",
        "packageVulnerabilityDetails": {
            "cvss": [{
                "baseScore": 4.7,
                "scoringVector": "CVSS:3.1/AV:L/AC:H/PR:L/UI:N/S:U/C:N/I:N/A:H",
                "source": "NVD",
                "version": "3.1"
            }],
            "referenceUrls": ["https://lore.kernel.org/all/
CAFcO6XN7JDM4xSXGhtusQfS2mSBcx50VJKwQpCq=WeLt57aaZA@mail.gmail.com/", "https://
ubuntu.com/security/notices/USN-5792-1", "https://ubuntu.com/security/notices/
USN-5791-2", "https://ubuntu.com/security/notices/USN-5791-1", "https://ubuntu.com/
security/notices/USN-5793-2", "https://git.kernel.org/pub/scm/linux/kernel/git/
torvalds/linux.git/commit/?id=8423f0b6d513b259fdab9c9bf4aaa6188d054c2d", "https://
ubuntu.com/security/notices/USN-5793-1", "https://ubuntu.com/security/notices/
USN-5792-2", "https://ubuntu.com/security/notices/USN-5791-3", "https://ubuntu.com/
security/notices/USN-5793-4", "https://ubuntu.com/security/notices/USN-5793-3",
 "https://git.kernel.org/linus/8423f0b6d513b259fdab9c9bf4aaa6188d054c2d(6.0-rc5)",
 "https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2022-3303"],
            "relatedVulnerabilities": [],
            "source": "UBUNTU_CVE",
            "sourceUrl": "https://people.canonical.com/~ubuntu-security/cve/2022/
CVE-2022-3303.html",
            "vendorCreatedAt": "Sep 27, 2022, 11:15:00 PM",
            "vendorSeverity": "medium",
            "vulnerabilityId": "CVE-2022-3303",
            "vulnerablePackages": [{
                "arch": "X86_64",
                "epoch": 0,
                "fixedInVersion": "0:5.15.0.1027.31~20.04.16",
                "name": "linux-image-aws",
                "packageManager": "OS",
                "remediation": "apt update && apt install --only-upgrade linux-image-
aws",
                "version": "5.15.0.1026.30~20.04.16"
            }]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
```

```
        "resources": [{
            "details": {
                "awsEc2Instance": {
                    "iamInstanceProfileArn": "arn:aws:iam::111122223333:instance-
profile/AmazonSSMRoleForInstancesQuickSetup",
                    "imageId": "ami-0b7ff1a8d69f1bb35",
                    "ipV4Addresses": ["172.31.85.212", "44.203.45.27"],
                    "ipV6Addresses": [],
                    "launchedAt": "Jan 19, 2023, 7:53:14 PM",
                    "platform": "UBUNTU_20_04",
                    "subnetId": "subnet-8213f2a3",
                    "type": "t2.micro",
                    "vpcId": "vpc-ab6650d1"
                }
            },
            "id": "i-0c2a343f1948d5205",
            "partition": "aws",
            "region": "us-east-1",
            "type": "AWS_EC2_INSTANCE"
        }],
        "severity": "MEDIUM",
        "status": "ACTIVE",
        "title": "CVE-2022-3303 - linux-image-aws",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Jan 19, 2023, 10:46:15 PM"
    }
}
```


创建 EventBridge 规则以通知您 Amazon Inspector 的调查发现

为了提高 Amazon Inspector 调查发现的可见性，您可以使用 EventBridge 设置自动调查发现提醒，并将其发送到消息中心。本主题向您展示如何向电子邮件、Slack 或 Amazon Chime 发送严重性为 CRITICAL 和 HIGH 的调查发现提醒。您将学习如何设置 Amazon Simple Notification Service 主题，然后将该主题关联到 EventBridge 事件规则。

第 1 步。设置 Amazon SNS 主题和端点

要设置自动警报，必须首先在 Amazon Simple Notification Service 中设置一个主题并添加一个端点。有关更多信息，请参阅 SNS 指南。

此过程可确定要将 Amazon Inspector 调查发现数据发送到何处。SNS 主题可在创建事件规则期间或之后添加到 EventBridge 事件规则中。

Email setup

创建 SNS 主题

1. 登录 Amazon SNS 控制台：https://console.aws.amazon.com/sns/v3/home。
2. 从导航窗格中选择主题，然后选择创建主题。
3. 在创建主题部分中，选择标准。接下来，输入主题名称，如 **`Inspector_to_Email`**。其他详细信息是可选的。
4. 选择创建主题。这将打开一个包含新主题详细信息的新面板。
5. 在订阅部分中，选择创建订阅。
6. a. 从协议菜单中选择电子邮件。
   b. 在端点字段中，添加您想要用于接收通知的电子邮件地址。

   > **Note**
   >
   > 创建订阅后，您需要通过电子邮件客户端确认订阅。

   c. 选择创建订阅。
7. 在收件箱中查收订阅消息，然后选择确认订阅。

Slack setup

创建 SNS 主题

1. 登录 Amazon SNS 控制台：https://console.aws.amazon.com/sns/v3/home。
2. 从导航窗格中选择主题，然后选择创建主题。
3. 在创建主题部分中，选择标准。接下来，输入主题名称，如 **`Inspector_to_Slack`**。其他详细信息是可选的。选择创建主题以完成端点的创建。

配置 AWS Chatbot 客户端

1. 导航到 AWS Chatbot 控制台，网址为：https://console.aws.amazon.com/chatbot/。
2. 从配置的客户端面板中选择配置新的客户端。
3. 选择 Slack，然后选择配置以确认。

> **Note**
>
> 选择 Slack 时，必须通过选择允许来确认 AWS Chatbot 访问通道的权限。

4. 选择配置新通道以打开配置详细信息窗格。
   a. 输入通道的名称。
   b. 对于 Slack 通道，选择您要使用的通道。
   c. 在 Slack 中，右键单击通道名称并选择复制链接，复制私有通道的通道 ID。
   d. 在AWS Management Console的 AWS Chatbot 窗口中，将从 slack 复制的通道 ID 粘贴到私有通道 ID 字段。
   e. 在权限中，如果您还没有角色，请选择使用模板创建 IAM 角色。
   f. 对于策略模板，请选择通知权限。这是 AWS Chatbot 的 IAM policy 模板。此策略能够为 CloudWatch 警报、事件和日志以及 Amazon SNS 主题，提供必要的读取和列出权限。
   g. 要查看通道防护机制策略，请选择 AmazonInspector2ReadOnlyAccess。
   h. 选择您之前创建 SNS 主题的区域，然后选择您创建的用于向 Slack 通道发送通知的 Amazon SNS 主题。
5. 选择配置。

Amazon Chime setup

创建 SNS 主题

1. 登录 Amazon SNS 控制台：https://console.aws.amazon.com/sns/v3/home。
2. 从导航窗格中选择主题，然后选择创建主题。
3. 在创建主题部分中，选择标准。接下来，输入主题名称，如 **`Inspector_to_Chime`**。其他详细信息是可选的。选择创建主题以完成。

配置 AWS Chatbot 客户端

1. 导航到 AWS Chatbot 控制台，网址为：https://console.aws.amazon.com/chatbot/。
2. 从已配置的客户端面板中选择配置新客户端。
3. 选择 Chime，然后选择配置以确认。
4. 在配置详细信息窗格中，输入通道的名称。

5. 在 Amazon Chime 中打开所需的聊天室。
   a. 选择右上角的齿轮图标，然后选择管理 Webhook 和自动程序。
   b. 选择复制 URL 以将 Webhook URL 复制到剪贴板。
6. 在AWS Management Console的 AWS Chatbot 窗口中，将复制的 URL 粘贴到 Webhook URL 字段中。
7. 在权限中，如果您还没有角色，请选择使用模板创建 IAM 角色。
8. 对于策略模板，请选择通知权限。这是 AWS Chatbot 的 IAM policy 模板。此模板能够为 CloudWatch 警报、事件和日志以及 Amazon SNS 主题，提供必要的读取和列出权限。
9. 选择您之前创建 SNS 主题的区域，然后选择您创建的用于向 Amazon Chime 聊天室发送通知的 Amazon SNS 主题。
10. 选择配置。

第 2 步。为 Amazon Inspector 调查发现创建 EventBridge 规则

1. 访问 https://console.aws.amazon.com/events/，打开 Amazon EventBridge 控制台。
2. 从导航窗格中选择规则，然后选择创建规则。
3. 输入规则名称，另还可选择输入描述。
4. 选择具有事件模式的规则，然后选择下一步。
5. 在事件模式窗格中，选择自定义模式（JSON 编辑器）。
6. 将下面的 JSON 粘贴到编辑器中。

```
{
  "source": ["aws.inspector2"],
  "detail-type": ["Inspector2 Finding"],
  "detail": {
    "severity": ["HIGH", "CRITICAL"],
    "status": ["ACTIVE"]
  }
}
```

> **Note**
>
> 此模式会针对 Amazon Inspector 检测到的各种严重性为 `CRITICAL` 或 `HIGH` 的活动调查发现发送通知。

输入完事件模式后，选择下一步。

7. 在选择目标页面上，选择 AWS 服务。然后，对于选择目标类型，选择 SNS 主题。
8. 对于选择主题，请选择您在第 1 步中创建的 SNS 主题的名称。然后选择下一步。
9. 根据需要添加可选标签，然后选择下一步。
10. 检查规则，然后选择创建规则。

适用于 Amazon Inspector 多账户环境的 EventBridge

如果您是 Amazon Inspector 委托管理员，则您的账户中显示的 EventBridge 规则取决于您的成员账户的适用调查发现。如果您在管理员账户中通过 EventBridge 设置了调查发现通知（如上一部分所述），您将收到有关多个账户的通知。换句话说，除了您自己账户生成的调查发现和事件的通知外，您还会收到您的成员账户生成的调查发现和事件的通知。

您可以使用调查发现的 JSON 详细信息中的 `accountId` 来识别产生 Amazon Inspector 调查发现的成员账户。

使用 Amazon Inspector 导出 SBOM

您可以使用 Amazon Inspector 控制台或 API 为您的资源生成软件材料清单 (SBOM)。SBOM 是您的代码库中所有开源和第三方软件组件的嵌套清单。Amazon Inspector 为环境中的各个资源提供 SBOM。从 Amazon Inspector 导出的 SBOM 可以帮助您了解有关软件供应的信息，例如您最常用的程序包以及整个组织的相关漏洞。

您可以为 Amazon Inspector 正在主动监测的所有受支持资源导出 SBOM。您可以通过 评测 Amazon Inspector 对 AWS 环境的覆盖率 查看资源的状态。

> **Note**
>
> Amazon Inspector 不支持为 Windows EC2 实例导出 SBOM。

Amazon Inspector

Amazon Inspector 支持以 CycloneDX 1.4 和 SPDX 2.3 兼容格式导出 SBOM。Amazon Inspector 将 SBOM 以 JSON 文件格式导出到您选择的 Amazon S3 存储桶。

> **Note**
>
> 从 Amazon Inspector 导出的 SPDX 格式与使用 SPDX 2.3 的系统兼容，但它们不包含无权利保留协议 (CC0) 字段。这是因为包含此字段将允许用户重新分发或编辑材料。

来自 Amazon Inspector 的 CycloneDX 1.4 SBOM 格式示例


```
                  {
"bomFormat": "CycloneDX",
"specVersion": "1.4",
"version": 1,
"metadata": {
  "timestamp": "2023-06-02T01:17:46Z",
  "component": null,
  "properties": [
    {
      "name": "imageId",
```

```
        "value":
 "sha256:c8ee97f7052776ef223080741f61fcdf6a3a9107810ea9649f904aa4269fdac6"
      },
      {
        "name": "architecture",
        "value": "arm64"
      },
      {
        "name": "accountId",
        "value": "111122223333"
      },
      {
        "name": "resourceType",
        "value": "AWS_ECR_CONTAINER_IMAGE"
      }
    ]
  },
  "components": [
    {
      "type": "library",
      "name": "pip",
      "purl": "pkg:pypi/pip@22.0.4?path=usr/local/lib/python3.8/site-packages/
pip-22.0.4.dist-info/METADATA",
      "bom-ref": "98dc550d1e9a0b24161daaa0d535c699"
    },
    {
      "type": "application",
      "name": "libss2",
      "purl": "pkg:dpkg/libss2@1.44.5-1+deb10u3?
arch=ARM64&epoch=0&upstream=libss2-1.44.5-1+deb10u3.src.dpkg",
      "bom-ref": "2f4d199d4ef9e2ae639b4f8d04a813a2"
    },
    {
      "type": "application",
      "name": "liblz4-1",
      "purl": "pkg:dpkg/liblz4-1@1.8.3-1+deb10u1?
arch=ARM64&epoch=0&upstream=liblz4-1-1.8.3-1+deb10u1.src.dpkg",
      "bom-ref": "9a6be8907ead891b070e60f5a7b7aa9a"
    },
    {
      "type": "application",
      "name": "mawk",
      "purl": "pkg:dpkg/mawk@1.3.3-17+b3?
arch=ARM64&epoch=0&upstream=mawk-1.3.3-17+b3.src.dpkg",
```

```
      "bom-ref": "c2015852a729f97fde924e62a16f78a5"
    },
    {
      "type": "application",
      "name": "libgmp10",
      "purl": "pkg:dpkg/libgmp10@6.1.2+dfsg-4+deb10u1?
arch=ARM64&epoch=2&upstream=libgmp10-6.1.2+dfsg-4+deb10u1.src.dpkg",
      "bom-ref": "52907290f5beef00dff8da77901b1085"
    },
    {
      "type": "application",
      "name": "ncurses-bin",
      "purl": "pkg:dpkg/ncurses-bin@6.1+20181013-2+deb10u3?
arch=ARM64&epoch=0&upstream=ncurses-bin-6.1+20181013-2+deb10u3.src.dpkg",
      "bom-ref": "cd20cfb9ebeeadba3809764376f43bce"
    }
  ],
  "vulnerabilities": [
    {
      "id": "CVE-2022-40897",
      "affects": [
        {
          "ref": "a74a4862cc654a2520ec56da0c81cdb3"
        },
        {
          "ref": "0119eb286405d780dc437e7dbf2f9d9d"
        }
      ]
    }
  ]
}
```

来自 Amazon Inspector 的 SPDX 2.3 SBOM 格式示例

```
{
 "name": "409870544328/EC2/i-022fba820db137c64/ami-074ea14c08effb2d8",
 "spdxVersion": "SPDX-2.3",
 "creationInfo": {
  "created": "2023-06-02T21:19:22Z",
  "creators": [
   "Organization: 409870544328",
```

```
    "Tool: Amazon Inspector SBOM Generator"
   ]
  },
  "documentNamespace": "EC2://i-022fba820db137c64/AMAZON_LINUX_2/null/x86_64",
  "comment": "",
  "packages": [{
    "name": "elfutils-libelf",
    "versionInfo": "0.176-2.amzn2",
    "downloadLocation": "NOASSERTION",
    "sourceInfo": "/var/lib/rpm/Packages",
    "filesAnalyzed": false,
    "externalRefs": [{
     "referenceCategory": "PACKAGE-MANAGER",
     "referenceType": "purl",
     "referenceLocator": "pkg:rpm/elfutils-libelf@0.176-2.amzn2?
arch=X86_64&epoch=0&upstream=elfutils-libelf-0.176-2.amzn2.src.rpm"
    }],
    "SPDXID": "SPDXRef-Package-rpm-elfutils-libelf-ddf56a513c0e76ab2ae3246d9a91c463"
   },
   {
    "name": "libcurl",
    "versionInfo": "7.79.1-1.amzn2.0.1",
    "downloadLocation": "NOASSERTION",
    "sourceInfo": "/var/lib/rpm/Packages",
    "filesAnalyzed": false,
    "externalRefs": [{
      "referenceCategory": "PACKAGE-MANAGER",
      "referenceType": "purl",
      "referenceLocator": "pkg:rpm/libcurl@7.79.1-1.amzn2.0.1?
arch=X86_64&epoch=0&upstream=libcurl-7.79.1-1.amzn2.0.1.src.rpm"
     },
     {
      "referenceCategory": "SECURITY",
      "referenceType": "vulnerability",
      "referenceLocator": "CVE-2022-32205"
     }
    ],
    "SPDXID": "SPDXRef-Package-rpm-libcurl-710fb33829bc5106559bcd380cddb7d5"
   },
   {
    "name": "hunspell-en-US",
    "versionInfo": "0.20121024-6.amzn2.0.1",
    "downloadLocation": "NOASSERTION",
    "sourceInfo": "/var/lib/rpm/Packages",
```

```
   "filesAnalyzed": false,
   "externalRefs": [{
    "referenceCategory": "PACKAGE-MANAGER",
    "referenceType": "purl",
    "referenceLocator": "pkg:rpm/hunspell-en-US@0.20121024-6.amzn2.0.1?
arch=NOARCH&epoch=0&upstream=hunspell-en-US-0.20121024-6.amzn2.0.1.src.rpm"
   }],
   "SPDXID": "SPDXRef-Package-rpm-hunspell-en-US-de19ae0883973d6cea5e7e079d544fe5"
  },
  {
   "name": "grub2-tools-minimal",
   "versionInfo": "2.06-2.amzn2.0.6",
   "downloadLocation": "NOASSERTION",
   "sourceInfo": "/var/lib/rpm/Packages",
   "filesAnalyzed": false,
   "externalRefs": [{
     "referenceCategory": "PACKAGE-MANAGER",
     "referenceType": "purl",
     "referenceLocator": "pkg:rpm/grub2-tools-minimal@2.06-2.amzn2.0.6?
arch=X86_64&epoch=1&upstream=grub2-tools-minimal-2.06-2.amzn2.0.6.src.rpm"
    },
    {
     "referenceCategory": "SECURITY",
     "referenceType": "vulnerability",
     "referenceLocator": "CVE-2021-3981"
    }
   ],
   "SPDXID": "SPDXRef-Package-rpm-grub2-tools-minimal-c56b7ea76e5a28ab8f232ef6d7564636"
  },
  {
   "name": "unixODBC-devel",
   "versionInfo": "2.3.1-14.amzn2",
   "downloadLocation": "NOASSERTION",
   "sourceInfo": "/var/lib/rpm/Packages",
   "filesAnalyzed": false,
   "externalRefs": [{
    "referenceCategory": "PACKAGE-MANAGER",
    "referenceType": "purl",
    "referenceLocator": "pkg:rpm/unixODBC-devel@2.3.1-14.amzn2?
arch=X86_64&epoch=0&upstream=unixODBC-devel-2.3.1-14.amzn2.src.rpm"
   }],
   "SPDXID": "SPDXRef-Package-rpm-unixODBC-devel-1bb35add92978df021a13fc9f81237d2"
  }
 ],
```

```
 "relationships": [{
   "spdxElementId": "SPDXRef-DOCUMENT",
   "relatedSpdxElement": "SPDXRef-Package-rpm-elfutils-libelf-
ddf56a513c0e76ab2ae3246d9a91c463",
   "relationshipType": "DESCRIBES"
  },
  {
   "spdxElementId": "SPDXRef-DOCUMENT",
   "relatedSpdxElement": "SPDXRef-Package-rpm-yajl-8476ce2db98b28cfab2b4484f84f1903",
   "relationshipType": "DESCRIBES"
  },
  {
   "spdxElementId": "SPDXRef-DOCUMENT",
   "relatedSpdxElement": "SPDXRef-Package-rpm-unixODBC-
devel-1bb35add92978df021a13fc9f81237d2",
   "relationshipType": "DESCRIBES"
  }
 ],
 "SPDXID": "SPDXRef-DOCUMENT"
}
```

SBOM 筛选条件

导出 SBOM 时，可以使用筛选条件，为特定资源子集创建报告。如果您不提供筛选条件，则会导出所有活动、受支持的资源的 SBOM。而且，如果您是委托管理员，这还包括所有成员的资源。可使用以下筛选条件：

- AccountID — 此筛选条件可用于导出与特定账户 ID 关联的资源的 SBOM。
- EC2 实例标签 — 此筛选条件可用于导出带有特定标签的 EC2 实例的 SBOM。
- 函数名称 — 此筛选条件可用于导出特定 Lambda 函数的 SBOM。
- 映像标签 — 此筛选条件可用于导出带有特定标签的容器映像的 SBOM。
- Lambda 函数标签 — 此筛选条件可用于导出带有特定标签的 Lambda 函数的 SBOM。
- 资源类型 — 此筛选条件可用于筛选资源类型：EC2/ECR/Lambda。
- 资源 ID — 此筛选条件可用于导出特定资源的 SBOM。
- 存储库名称 — 此筛选条件可用于为特定存储库中的容器映像生成 SBOM。

配置和导出 SBOM

要导出 SBOM，必须先配置 Amazon S3 存储桶和允许 Amazon Inspector 使用的 AWS KMS 密钥。您可以使用筛选条件为资源的特定子集导出 SBOM。要为 AWS 组织中的多个账户导出 SBOM，请在以 Amazon Inspector 委托管理员的身份登录后执行以下步骤。

先决条件

- Amazon Inspector 主动监测的受支持资源。
- 配置了策略的 Amazon S3 存储桶，允许 Amazon Inspector 向存储桶添加对象。有关配置策略的信息，请参阅配置导出权限。
- 配置了策略的 AWS KMS 密钥，允许 Amazon Inspector 使用该策略来加密报告。有关配置策略的信息，请参阅配置用于导出的 AWS KMS 密钥。

> **Note**
>
> 如果您之前配置了 Amazon S3 存储桶和用于调查发现导出的 AWS KMS 密钥，则可以将这些存储桶和密钥用于 SBOM 导出。

选择您的首选访问方法来导出 SBOM。

Console

1. 打开 Amazon Inspector 控制台：https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的 AWS 区域选择器选择要为其导出 SBOM 的资源所在的区域。
3. 在导航窗格中，选择导出 SBOM。
4. （可选）在导出 SBOM 页面中，使用添加筛选条件菜单选择要为其创建报告的资源子集。如果未提供筛选条件，Amazon Inspector 将导出所有活动资源的报告。如果您是委托管理员，这将包括您组织中的所有活动资源。
5. 在导出设置下，选择需要的 SBOM 格式。
6. 输入 Amazon S3 URI 或选择浏览 Amazon S3，选择一个 Amazon S3 位置来存储 SBOM。
7. 输入为 Amazon Inspector 配置的密钥，用于加密报告。AWS KMS

API

- 要以编程方式为资源导出 SBOM，请使用 Amazon Inspector API 的 CreateSbomExport 操作。

  在请求中，使用 `reportFormat` 参数指定 SBOM 输出格式，选择 `CYCLONEDX_1_4` 或 `SPDX_2_3`。`s3Destination` 参数是必填的，而且您必须指定一个配置了策略的 S3 存储桶，以允许 Amazon Inspector 对其进行写入。（可选）使用 `resourceFilterCriteria` 参数将报告的范围限制在特定的资源。

AWS CLI

- 要使用 AWS Command Line Interface 为您的资源导出 SBOM，请运行以下命令：

  ```
  aws inspector2 create-sbom-export --report-format
  FORMAT --s3-destination bucketName=DOC-EXAMPLE-
  BUCKET1,keyPrefix=PREFIX,kmsKeyArn=arn:aws:kms:Region:111122223333:key/123
  ```

  在您的请求中，将 *FORMAT* 替换为您选择的格式：`CYCLONEDX_1_4` 或 `SPDX_2_3`。然后，将 s3 目标的 *user input placeholders* 替换为要导出到的 S3 存储桶的名称、用于 S3 中输出的前缀，以及用于加密报告的 KMS 密钥的 ARN。

Amazon Inspector 漏洞数据库搜索

您可以在 Amazon Inspector 漏洞数据库中搜索漏洞和风险敞口 (CVE)。Amazon Inspector 使用漏洞数据库中的信息来生成与 CVE ID 相关的详细信息。您可以在 CVE 详细信息页面中访问这些详细信息。

本主题介绍如何使用 CVE ID 搜索 Amazon Inspector 漏洞数据库并解读 CVE 详细信息页面。有关调查结果的信息，请参见Amazon Inspector 调查发现详细信息。

> **Note**
>
> Amazon Inspector 会跟踪数据库中的其他软件漏洞并生成调查结果。但是，Amazon Inspector 仅支持 CVE 详情页面的 “检测平台” 部分列出的平台的 CVE。目前，CVE 搜索不支持 Microsoft Windows。

搜索漏洞数据库

本节介绍如何在控制台中和使用 Amazon Inspector API 搜索漏洞数据库。

> **Note**
>
> 您必须在当前版本中激活 Amazon Inspector， AWS 区域 然后才能搜索漏洞数据库。

Console

1. 打开 Amazon Inspector 控制台：https://console.aws.amazon.com/inspector/
2. 在导航窗格中，选择漏洞数据库搜索。
3. 在搜索栏中输入 CVE ID，然后选择搜索。

API

运行 Amazon Inspector SearchVulnerabilitiesAPI，并按`filterCriteria`以下格式提供单个 CVE ID：`CVE-<year>-<ID>`。

了解 CVE 详情

本节介绍如何解读 CVE 详细信息页面。

CVE 详情

CVE 详细信息部分包含以下信息：

- CVE 描述和 ID
- CVE 严重性
- 常见漏洞评分系统 (CVSS) 和漏洞利用预测评分系统 (EPSS) 分数
- 检测平台

> **Note**
>
> 如果此字段为空，则 Amazon Inspector 不支持检测您的 CVE ID。

- 常见弱点枚举 (CWE)
- 供应商创建和更新日期

漏洞情报

漏洞情报部分提供威胁情报数据，例如漏洞利用目标和上次已知的公开漏洞利用日期。

它还提供了来自网络安全和基础设施安全局（CISA）的数据，其中包括补救措施、CVE被添加到已知被利用漏洞目录的日期以及CISA期望联邦机构修复CVE的日期。

参考信息

参考部分提供资源链接，以获取有关 CVE 的更多信息。

用于 Amazon Inspector 事件的 Amazon EventBridge 事件架构

为了支持与其他应用程序、服务和系统（例如监控或事件管理系统）的集成，Amazon Inspector 会自动将调查发现作为事件发布到 Amazon EventBridge。EventBridge 是一项无服务器事件总线服务，可将来自应用程序和其他 AWS 服务的实时数据流传输到 AWS Lambda 函数、Amazon Simple Notification Service 主题和 Amazon Kinesis Data Streams 流等目标。有关 EventBridge 和 EventBridge 事件的更多信息，请参阅 Amazon EventBridge 用户指南。

Amazon Inspector 会发布调查发现、资源覆盖率变更和对单个资源的初始扫描等事件。每个事件都是一个符合 AWS 事件 EventBridge 架构的 JSON 对象。由于数据结构为 EventBridge 事件，因此您可以使用其他应用程序、服务和工具，更轻松地监控、处理调查发现和支持的 Amazon Inspector 事件，并根据它们采取行动。

主题

用于 Amazon Inspector 的 Amazon EventBridge 基本架构

以下是 Amazon Inspector 的 EventBridge 事件的基本架构示例。事件详情因事件类型而异。

```
{
    "version": "0",
    "id": "Event ID",
    "detail-type": "Inspector2 *event type*",
    "source": "aws.inspector2",
    "account": "AWS ## ID (string)",
    "time": "event timestamp (string)",
    "region": "AWS ## (string)",
    "resources": [
        *IDs or ARNs of the resources involved in the event*
    ],
    "detail": {
        *Details of an Amazon Inspector event type*
```

```
    }
}
```

Amazon Inspector 调查发现事件架构示例

以下是 Amazon Inspector 调查发现的 EventBridge 事件的架构示例。当 Amazon Inspector 发现您的某个资源中存在软件漏洞或网络问题时，就会创建调查发现事件。有关创建针对此类事件的通知的指南，请参阅使用 Amazon EventBridge 创建对 Amazon Inspector 调查发现的自定义响应。

以下字段可识别调查发现事件：

- `detail-type` 字段设置为 `Inspector2 Finding`。
- `detail` 对象描述调查发现。

从选项中进行选择，查看针对不同资源和调查发现类型的调查发现事件架构。

Amazon EC2 package vulnerability finding

```
{
    "version": "0",
    "id": "66a7a279-5f92-971c-6d3e-c92da0950992",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-19T22:46:15Z",
    "region": "us-east-1",
    "resources": ["i-0c2a343f1948d5205"],
    "detail": {
        "awsAccountId": "111122223333",
        "description": "\n It was discovered that the sound subsystem in the Linux
 kernel contained a\n race condition in some situations. A local attacker could use
 this to cause\n a denial of service (system crash).",
        "exploitAvailable": "YES",
        "exploitabilityDetails": {
            "lastKnownExploitAt": "Oct 24, 2022, 11:08:59 PM"
        },
        "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/
FINDING_ID",
        "firstObservedAt": "Jan 19, 2023, 10:46:15 PM",
        "fixAvailable": "YES",
```

```
        "lastObservedAt": "Jan 19, 2023, 10:46:15 PM",
        "packageVulnerabilityDetails": {
            "cvss": [{
                "baseScore": 4.7,
                "scoringVector": "CVSS:3.1/AV:L/AC:H/PR:L/UI:N/S:U/C:N/I:N/A:H",
                "source": "NVD",
                "version": "3.1"
            }],
            "referenceUrls": ["https://lore.kernel.org/all/
CAFcO6XN7JDM4xSXGhtusQfS2mSBcx50VJKwQpCq=WeLt57aaZA@mail.gmail.com/", "https://
ubuntu.com/security/notices/USN-5792-1", "https://ubuntu.com/security/notices/
USN-5791-2", "https://ubuntu.com/security/notices/USN-5791-1", "https://ubuntu.com/
security/notices/USN-5793-2", "https://git.kernel.org/pub/scm/linux/kernel/git/
torvalds/linux.git/commit/?id=8423f0b6d513b259fdab9c9bf4aaa6188d054c2d", "https://
ubuntu.com/security/notices/USN-5793-1", "https://ubuntu.com/security/notices/
USN-5792-2", "https://ubuntu.com/security/notices/USN-5791-3", "https://ubuntu.com/
security/notices/USN-5793-4", "https://ubuntu.com/security/notices/USN-5793-3",
 "https://git.kernel.org/linus/8423f0b6d513b259fdab9c9bf4aaa6188d054c2d(6.0-rc5)",
 "https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2022-3303"],
            "relatedVulnerabilities": [],
            "source": "UBUNTU_CVE",
            "sourceUrl": "https://people.canonical.com/~ubuntu-security/cve/2022/
CVE-2022-3303.html",
            "vendorCreatedAt": "Sep 27, 2022, 11:15:00 PM",
            "vendorSeverity": "medium",
            "vulnerabilityId": "CVE-2022-3303",
            "vulnerablePackages": [{
                "arch": "X86_64",
                "epoch": 0,
                "fixedInVersion": "0:5.15.0.1027.31~20.04.16",
                "name": "linux-image-aws",
                "packageManager": "OS",
                "remediation": "apt update && apt install --only-upgrade linux-
image-aws",
                "version": "5.15.0.1026.30~20.04.16"
            }]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [{
            "details": {
```

```
                "awsEc2Instance": {
                    "iamInstanceProfileArn": "arn:aws:iam::111122223333:instance-
profile/AmazonSSMRoleForInstancesQuickSetup",
                    "imageId": "ami-0b7ff1a8d69f1bb35",
                    "ipV4Addresses": ["172.31.85.212", "44.203.45.27"],
                    "ipV6Addresses": [],
                    "launchedAt": "Jan 19, 2023, 7:53:14 PM",
                    "platform": "UBUNTU_20_04",
                    "subnetId": "subnet-8213f2a3",
                    "type": "t2.micro",
                    "vpcId": "vpc-ab6650d1"
                }
            },
            "id": "i-0c2a343f1948d5205",
            "partition": "aws",
            "region": "us-east-1",
            "type": "AWS_EC2_INSTANCE"
        }],
        "severity": "MEDIUM",
        "status": "ACTIVE",
        "title": "CVE-2022-3303 - linux-image-aws",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Jan 19, 2023, 10:46:15 PM"
    }
}
```

Amazon EC2 network reachability finding

```
{
    "version": "0",
    "id": "d0384f63-1621-1b75-d014-a5e45628ef3e",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-20T09:17:57Z",
    "region": "us-east-1",
    "resources": ["i-0a96278c2206a8e4b"],
    "detail": {
        "awsAccountId": "111122223333",
```

```
        "description": "On the instance i-0a96278c2206a8e4b, the port range
 22-22 is reachable from the InternetGateway igw-72069c09 from an attached ENI
 eni-0976efe678170408f.",
        "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/
FINDING_ID",
        "firstObservedAt": "Jan 20, 2023, 9:17:57 AM",
        "lastObservedAt": "Jan 20, 2023, 9:17:57 AM",
        "networkReachabilityDetails": {
            "networkPath": {
                "steps": [{
                    "componentId": "igw-72069c09",
                    "componentType": "AWS::EC2::InternetGateway"
                }, {
                    "componentId": "acl-91d74eec",
                    "componentType": "AWS::EC2::NetworkAcl"
                }, {
                    "componentId": "sg-0aaed0af450bd0165",
                    "componentType": "AWS::EC2::SecurityGroup"
                }, {
                    "componentId": "eni-0976efe678170408f",
                    "componentType": "AWS::EC2::NetworkInterface"
                }, {
                    "componentId": "i-0a96278c2206a8e4b",
                    "componentType": "AWS::EC2::Instance"
                }]
            },
            "openPortRange": {
                "begin": 22,
                "end": 22
            },
            "protocol": "TCP"
        },
        "remediation": {
            "recommendation": {
                "text": "You can restrict access to your instance by modifying the
 Security Groups or ACLs in the network path."
            }
        },
        "resources": [{
            "details": {
                "awsEc2Instance": {
                    "iamInstanceProfileArn": "arn:aws:iam::111122223333:instance-
profile/AmazonSSMRoleForInstancesQuickSetup",
                    "imageId": "ami-0b5eea76982371e91",
```

```
                    "ipV4Addresses": ["3.89.90.19", "172.31.93.57"],
                    "ipV6Addresses": [],
                    "keyName": "example-inspector-test",
                    "launchedAt": "Jan 19, 2023, 7:25:02 PM",
                    "platform": "AMAZON_LINUX_2",
                    "subnetId": "subnet-8213f2a3",
                    "type": "t2.micro",
                    "vpcId": "vpc-ab6650d1"
                }
            },
            "id": "i-0a96278c2206a8e4b",
            "partition": "aws",
            "region": "us-east-1",
            "type": "AWS_EC2_INSTANCE"
        }],
        "severity": "MEDIUM",
        "status": "ACTIVE",
        "title": "Port 22 is reachable from an Internet Gateway",
        "type": "NETWORK_REACHABILITY",
        "updatedAt": "Jan 20, 2023, 9:17:57 AM"
    }
}
```

Amazon ECR package vulnerability finding

```
{
    "version": "0",
    "id": "5b52952e-26df-3a51-6d14-4dbe737e58ec",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-19T21:59:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ecr:us-east-1:111122223333:repository/inspector2/
sha256:98f0304b3a3b7c12ce641177a99d1f3be56f532473a528fda38d53d519cafb13"
    ],
    "detail": {
        "awsAccountId": "111122223333",
        "description": "libcurl would reuse a previously created connection even
 when a TLS or SSHrelated option had been changed that should have prohibited
```

```
 reuse.libcurl keeps previously used connections in a connection pool for
 subsequenttransfers to reuse if one of them matches the setup. However, several TLS
 andSSH settings were left out from the configuration match checks, making themmatch
 too easily.",
        "exploitAvailable": "NO",
        "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/
FINDING_ID",
        "firstObservedAt": "Jan 19, 2023, 9:59:00 PM",
        "fixAvailable": "YES",
        "inspectorScore": 7.5,
        "inspectorScoreDetails": {
            "adjustedCvss": {
                "adjustments": [],
                "cvssSource": "NVD",
                "score": 7.5,
                "scoreSource": "NVD",
                "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:N",
                "version": "3.1"
            }
        },
        "lastObservedAt": "Jan 19, 2023, 9:59:00 PM",
        "packageVulnerabilityDetails": {
            "cvss": [
                {
                    "baseScore": 5,
                    "scoringVector": "AV:N/AC:L/Au:N/C:N/I:P/A:N",
                    "source": "NVD",
                    "version": "2.0"
                },
                {
                    "baseScore": 7.5,
                    "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:N",
                    "source": "NVD",
                    "version": "3.1"
                }
            ],
            "referenceUrls": [
                "https://hackerone.com/reports/1555796",
                "https://security.gentoo.org/glsa/202212-01",
                "https://lists.debian.org/debian-lts-announce/2022/08/
msg00017.html",
                "https://www.debian.org/security/2022/dsa-5197"
            ],
            "relatedVulnerabilities": [],
```

```
            "source": "NVD",
            "sourceUrl": "https://nvd.nist.gov/vuln/detail/CVE-2022-27782",
            "vendorCreatedAt": "Jun 2, 2022, 2:15:00 PM",
            "vendorSeverity": "HIGH",
            "vendorUpdatedAt": "Jan 5, 2023, 5:51:00 PM",
            "vulnerabilityId": "CVE-2022-27782",
            "vulnerablePackages": [
                {
                    "arch": "X86_64",
                    "epoch": 0,
                    "fixedInVersion": "0:7.61.1-22.el8_6.3",
                    "name": "libcurl",
                    "packageManager": "OS",
                    "release": "22.el8",
                    "remediation": "yum update libcurl",
                    "sourceLayerHash":
"sha256:38a980f2cc8accf69c23deae6743d42a87eb34a54f02396f3fcfd7c2d06e2c5b",
                    "version": "7.61.1"
                },
                {
                    "arch": "X86_64",
                    "epoch": 0,
                    "fixedInVersion": "0:7.61.1-22.el8_6.3",
                    "name": "curl",
                    "packageManager": "OS",
                    "release": "22.el8",
                    "remediation": "yum update curl",
                    "sourceLayerHash":
"sha256:38a980f2cc8accf69c23deae6743d42a87eb34a54f02396f3fcfd7c2d06e2c5b",
                    "version": "7.61.1"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
                "details": {
                    "awsEcrContainerImage": {
                        "architecture": "amd64",
```

```
                        "imageHash":
 "sha256:98f0304b3a3b7c12ce641177a99d1f3be56f532473a528fda38d53d519cafb13",
                        "imageTags": [
                            "o3"
                        ],
                        "platform": "ORACLE_LINUX_8",
                        "pushedAt": "Jan 19, 2023, 7:38:39 PM",
                        "registry": "111122223333",
                        "repositoryName": "inspector2"
                    }
                },
                "id": "arn:aws:ecr:us-east-1:111122223333:repository/inspector2/
sha256:98f0304b3a3b7c12ce641177a99d1f3be56f532473a528fda38d53d519cafb13",
                "partition": "aws",
                "region": "us-east-1",
                "type": "AWS_ECR_CONTAINER_IMAGE"
            }
        ],
        "severity": "HIGH",
        "status": "ACTIVE",
        "title": "CVE-2022-27782 - libcurl, curl",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Jan 19, 2023, 9:59:00 PM"
    }
}
```

Lambda package vulnerability finding

```
{
    "version": "0",
    "id": "040bb590-3a12-353f-ecb1-05e54b0fbea7",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-19T19:20:25Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:lambda:us-east-1:111122223333:function:ExampleFunction:$LATEST"
    ],
    "detail": {
        "awsAccountId": "111122223333",
```

```
        "description": "Those using Woodstox to parse XML data may be vulnerable to
 Denial of Service attacks (DOS) if DTD support is enabled. If the parser is running
 on user supplied input, an attacker may supply content that causes the parser to
 crash by stackoverflow. This effect may support a denial of service attack.",
        "exploitAvailable": "NO",
        "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/
FINDING_ID",
        "firstObservedAt": "Jan 19, 2023, 7:20:25 PM",
        "fixAvailable": "YES",
        "inspectorScore": 7.5,
        "inspectorScoreDetails": {
            "adjustedCvss": {
                "cvssSource": "NVD",
                "score": 7.5,
                "scoreSource": "NVD",
                "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H",
                "version": "3.1"
            }
        },
        "lastObservedAt": "Jan 19, 2023, 7:20:25 PM",
        "packageVulnerabilityDetails": {
            "cvss": [
                {
                    "baseScore": 7.5,
                    "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H",
                    "source": "NVD",
                    "version": "3.1"
                }
            ],
            "referenceUrls": [
                "https://bugs.chromium.org/p/oss-fuzz/issues/detail?id=47434"
            ],
            "relatedVulnerabilities": [],
            "source": "NVD",
            "sourceUrl": "https://nvd.nist.gov/vuln/detail/CVE-2022-40152",
            "vendorCreatedAt": "Sep 16, 2022, 10:15:00 AM",
            "vendorSeverity": "HIGH",
            "vendorUpdatedAt": "Nov 25, 2022, 11:15:00 AM",
            "vulnerabilityId": "CVE-2022-40152",
            "vulnerablePackages": [
                {
                    "epoch": 0,
                    "filePath": "lib/woodstox-core-6.2.7.jar",
                    "fixedInVersion": "6.4.0",
```

```
                    "name": "com.fasterxml.woodstox:woodstox-core",
                    "packageManager": "JAR",
                    "remediation": "Update woodstox-core to 6.4.0",
                    "version": "6.2.7"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
                "details": {
                    "awsLambdaFunction": {
                        "architectures": [
                            "X86_64"
                        ],
                        "codeSha256": "+EwrOrht2um4fdVCD73gj
+O7HJIAUvUxi8AD0eKHSkc=",
                        "executionRoleArn": "arn:aws:iam::111122223333:role/
ExampleFunction-ExecutionRole",
                        "functionName": "Example-function",
                        "lastModifiedAt": "Nov 7, 2022, 8:29:27 PM",
                        "packageType": "ZIP",
                        "runtime": "JAVA_11",
                        "version": "$LATEST"
                    }
                },
                "id": "arn:aws:lambda:us-
east-1:111122223333:function:ExampleFunction:$LATEST",
                "partition": "aws",
                "region": "us-east-1",
                "tags": {
                    "TargetAlias": "DeploymentStack",
                    "SoftwareType": "Infrastructure"
                },
                "type": "AWS_LAMBDA_FUNCTION"
            }
        ],
        "severity": "HIGH",
        "status": "ACTIVE",
        "title": "CVE-2022-40152 - com.fasterxml.woodstox:woodstox-core",
```

```
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Jan 19, 2023, 7:20:25 PM"
    }
}
```

Lambda code vulnerability finding

```
{
   "version":"0",
   "id":"9df01cb1-df24-bc46-5650-085a4087e7aa",
   "detail-type":"Inspector2 Finding",
   "source":"aws.inspector2",
   "account":"111122223333",
   "time":"2023-12-07T22:14:45Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:lambda:us-east-1:111122223333:function:code-finding:$LATEST"
   ],
   "detail":{
      "awsAccountId":"111122223333",
      "codeVulnerabilityDetails":{
         "detectorId":"python/lambda-override-reserved@v1.0",
         "detectorName":"Override of reserved variable names in a Lambda function",
         "detectorTags":[
            "availability",
            "aws-python-sdk",
            "aws-lambda",
            "data-integrity",
            "maintainability",
            "security",
            "security-context",
            "python"
         ],
         "filePath":{
            "endLine":6,
            "fileName":"lambda_function.py",
            "filePath":"lambda_function.py",
            "startLine":6
         },
         "ruleId":"Rule-434311"
      },
```

```
      "description":"Overriding environment variables that are reserved by AWS
 Lambda might lead to unexpected behavior or failure of the Lambda function.",
      "findingArn":"arn:aws:inspector2:us-east-1:111122223333:finding/FINDING_ID",
      "firstObservedAt":"Aug 8, 2023, 7:33:58 PM",
      "lastObservedAt":"Dec 7, 2023, 10:14:45 PM",
      "remediation":{
         "recommendation":{
            "text":"Your code attempts to override an environment variable that is
 reserved by the Lambda runtime environment. This can lead to unexpected behavior
 and might break the execution of your Lambda function.\n\n[Learn more](https://
docs.aws.amazon.com/lambda/latest/dg/configuration-envvars.html#configuration-
envvars-runtime)"
         }
      },
      "resources":[
         {
            "details":{
               "awsLambdaFunction":{
                  "architectures":[
                     "X86_64"
                  ],
                  "codeSha256":"2mtfH+CgubesG6NYpb2zEqBja5WN6FfbH4AAYDuF8RE=",
                  "executionRoleArn":"arn:aws:iam::193043430472:role/service-role/
code-finding-role-7jgg3wan",
                  "functionName":"code-finding",
                  "lastModifiedAt":"Dec 7, 2023, 10:12:48 PM",
                  "packageType":"ZIP",
                  "runtime":"PYTHON_3_7",
                  "version":"$LATEST"
               }
            },
            "id":"arn:aws:lambda:us-east-1:193043430472:function:code-finding:
$LATEST",
            "partition":"aws",
            "region":"us-east-1",
            "type":"AWS_LAMBDA_FUNCTION"
         }
      ],
      "severity":"HIGH",
      "status":"ACTIVE",
      "title":"Overriding environment variables that are reserved by AWS Lambda
 might lead to unexpected behavior.",
      "type":"CODE_VULNERABILITY",
      "updatedAt":"Dec 7, 2023, 10:14:45 PM"
```

```
    }
}
```

> **Note**
>
> 详细信息值以对象形式返回单个调查发现的 JSON 详细信息。它不会返回整个调查发现响应语法，该语法支持数组中的多个调查发现。

Amazon Inspector 初始扫描完成事件架构示例

以下是 Amazon Inspector 完成初始扫描事件的 EventBridge 事件架构示例。当 Amazon Inspector 完成对您的某个资源的初始扫描时，会创建此事件。

以下字段可识别初始扫描完成事件：

- `detail-type` 字段设置为 `Inspector2 Scan`。
- `detail` 对象包含一个 `finding-severity-counts` 对象，该对象详细说明了适用严重性类别中调查发现的数量，例如 `CRITICAL`、`HIGH` 和 `MEDIUM`。

从选项中进行选择，按资源类型查看不同的初始扫描事件架构。

Amazon EC2 instance initial scan

```
{
    "version": "0",
    "id": "28a46762-6ac8-6cc4-4f55-bc9ab99af928",
    "detail-type": "Inspector2 Scan",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-20T22:52:35Z",
    "region": "us-east-1",
    "resources": [
        "i-087d63509b8c97098"
    ],
    "detail": {
        "scan-status": "INITIAL_SCAN_COMPLETE",
```

```
        "finding-severity-counts": {
            "CRITICAL": 0,
            "HIGH": 0,
            "MEDIUM": 0,
            "TOTAL": 0
        },
        "instance-id": "i-087d63509b8c97098",
        "version": "1.0"
    }
}
```

Amazon ECR image initial scan

```
{
    "version": "0",
    "id": "fdaa751a-984c-a709-44f9-9a9da9cd3606",
    "detail-type": "Inspector2 Scan",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-20T23:15:18Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ecr:us-east-1:111122223333:repository/inspector2"
    ],
    "detail": {
        "scan-status": "INITIAL_SCAN_COMPLETE",
        "repository-name": "arn:aws:ecr:us-east-1:111122223333:repository/
inspector2",
        "finding-severity-counts": {
            "CRITICAL": 0,
            "HIGH": 0,
            "MEDIUM": 0,
            "TOTAL": 0
        },
        "image-digest":
 "sha256:965fbcae990b0467ed5657caceaec165018ef44a4d2d46c7cdea80a9dff0d1ea",
        "image-tags": [
            "ubuntu22"
        ],
        "version": "1.0"
    }
```

```
}
```

Lambda function initial scan

```
{
  "version": "0",
  "id": "4f290a7c-361b-c442-03c8-a629f6f20d6c",
  "detail-type": "Inspector2 Scan",
  "source": "aws.inspector2",
  "account": "111122223333",
  "time": "2023-02-23T18:06:03Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:lambda:us-west-2:111122223333:function:lambda-example:$LATEST"
  ],
  "detail": {
    "scan-status": "INITIAL_SCAN_COMPLETE",
    "finding-severity-counts": {
      "CRITICAL": 0,
      "HIGH": 0,
      "MEDIUM": 0,
      "TOTAL": 0
    },
    "version": "1.0"
  }
}
```

Amazon Inspector 覆盖率事件架构示例

以下是 Amazon Inspector 覆盖率事件的 EventBridge 事件架构示例。当 Amazon Inspector 扫描资源的覆盖率发生变化时，会创建此事件。以下字段可识别覆盖率事件：

- `detail-type` 字段设置为 `Inspector2 Coverage`。
- `detail` 对象包含一个 `scanStatus` 对象，用于指示资源的新扫描状态。

```
{
    "version": "0",
    "id": "000adda5-0fbf-913e-bc0e-10f0376412aa",
    "detail-type": "Inspector2 Coverage",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-20T22:51:39Z",
    "region": "us-east-1",
    "resources": [
        "i-087d63509b8c97098"
    ],
    "detail": {
        "scanStatus": {
            "reason": "UNMANAGED_EC2_INSTANCE",
            "statusCodeValue": "INACTIVE"
        },
        "scanType": "PACKAGE",
        "eventTimestamp": "2023-01-20T22:51:35.665501Z",
        "version": "1.0"
    }
}
```

将 Amazon Inspector 扫描集成到 CI/CD 管道中

您可以将 Amazon Inspector 容器映像扫描直接集成到 CI/CD 管道中，以扫描软件漏洞并在构建结束时提供报告。Amazon Inspector 生成的漏洞报告允许您在部署之前调查和修复风险。

Amazon Inspector CI/CD 集成结合了 Amazon Inspector SBOM 生成器和 Amazon Inspector Scan API，可为容器映像生成漏洞报告。Amazon Inspector SBOM 生成器根据提供的容器映像创建软件物料清单（SBOM），然后，Amazon Inspector Scan API 会扫描该 SBOM 并创建一份报告，其中包含检测到的所有漏洞的详细信息。

您可以通过专为单个 CI/CD 解决方案构建并在相应市场上提供的 Amazon Inspector 插件实现 CI/CD 与 Amazon Inspector 的集成，也可以自行创建自定义扫描集成。

主题

插件集成

Amazon Inspector 为支持的 CI/CD 解决方案提供了插件。您可以从相应的市场安装这些插件，然后使用它们将 Amazon Inspector 扫描作为构建步骤添加到管道中。插件构建步骤对您提供的映像运行 Amazon Inspector SBOM 生成器，然后对生成的 SBOM 运行 Amazon Inspector Scan API。

以下概述了 Amazon Inspector CI/CD 集成如何通过插件发挥作用：

1. 您可以将配置为 AWS 账户 允许访问 Amazon Inspector Scan API。有关说明，请参阅设置 AWS 账户以使用 Amazon Inspector CI/CD 集成。
2. 您可以安装市场上提供的 Amazon Inspector 插件。

3. 您可以安装和配置 Amazon Inspector SBOM 生成器二进制文件。有关说明，请参阅Amazon Inspector SBOM 生成器。
4. 您可以将 Amazon Inspector 扫描作为构建步骤添加到 CI/CD 管道中，然后配置扫描。
5. 在您运行构建时，该插件会将容器映像作为输入，然后在映像上运行 Amazon Inspector SBOM 生成器，以生成与 CycloneDX 兼容的 SBOM。
6. 然后，该插件将生成的 SBOM 发送到 Amazon Inspector Scan API 端点，该端点会评估每个 SBOM 组件是否存在漏洞。
7. Amazon Inspector Scan API 响应将转换为 CSV、SBOM JSON 和 HTML 格式的漏洞报告。该报告包含有关 Amazon Inspector 发现的任何漏洞的详细信息。

支持的 CI/CD 解决方案

Amazon Inspector 目前支持以下 CI/CD 解决方案。有关使用插件设置 CI/CD 集成的完整说明，请选择适用于 CI/CD 解决方案的插件：

- Jenkins 插件
- TeamCity 插件

自定义集成

如果 Amazon Inspector 未提供适用于您的 CI/CD 解决方案的插件，您可以结合使用 Amazon Inspector SBOM 生成器和 Amazon Inspector Scan API 来自行创建自定义 CI/CD 集成。您还可以使用 Amazon Inspector SBOM 生成器中提供的选项，借助自定义集成来微调扫描。

以下概述了自定义 Amazon Inspector CI/CD 集成如何发挥作用：

1. 您可以将配置为 AWS 账户 允许访问 Amazon Inspector Scan API。有关说明，请参阅设置 AWS 账户以使用 Amazon Inspector CI/CD 集成。
2. 您可以安装和配置 Amazon Inspector SBOM 生成器二进制文件。有关说明，请参阅Amazon Inspector SBOM 生成器。
3. 您可以使用 Amazon Inspector SBOM 生成器为容器映像生成与 CycloneDX 兼容的 SBOM。
4. 您可以对生成的 SBOM 使用 Amazon Inspector Scan API，从而生成漏洞报告。

有关设置自定义集成的说明，请参阅使用 Amazon Inspector 扫描自行创建自定义 CI/CD 管道集成。

设置 AWS 账户以使用 Amazon Inspector CI/CD 集成

您必须注册才能使用 Amazon Ins AWS 账户 pector CI/CD 集成。 AWS 账户 必须具有 IAM 角色，该角色可向你的管道授予对 Amazon Inspector Scan API 的访问权限。

完成以下主题中的任务以注册 AWS 账户、创建管理员用户以及为 CI/CD 集成配置 IAM 角色。

> **Note**
>
> 如果您已经注册了 AWS 账户，则可以跳至为 CI/CD 集成配置 IAM 角色。

主题

注册获取 AWS 账户

如果您没有 AWS 账户，请完成以下步骤来创建一个。

要注册 AWS 账户

1. 打开 https://portal.aws.amazon.com/billing/signup。
2. 按照屏幕上的说明进行操作。

   在注册时，将接到一通电话，要求使用电话键盘输入一个验证码。

   当您注册时 AWS 账户，就会创建AWS 账户根用户一个。根用户有权访问该账户中的所有 AWS 服务 和资源。作为安全最佳实践，请 为管理用户分配管理访问权限，并且只使用根用户执行需要根用户访问权限的任务。

AWS 注册过程完成后会向您发送一封确认电子邮件。在任何时候，您都可以通过转至 https://aws.amazon.com/ 并选择我的账户来查看当前的账户活动并管理您的账户。

创建管理用户

注册后，请保护您的安全 AWS 账户 AWS 账户根用户 AWS IAM Identity Center，启用并创建管理用户，这样您就不会使用 root 用户执行日常任务。

保护你的 AWS 账户根用户

1. 选择 Root 用户并输入您的 AWS 账户 电子邮件地址，以账户所有者的身份登录。AWS Management Console在下一页上，输入您的密码。

   要获取使用根用户登录方面的帮助，请参阅《AWS 登录 用户指南》中的以根用户身份登录。
2. 为您的根用户启用多重身份验证 (MFA)。

   有关说明，请参阅 I A M 用户指南中的为 AWS 账户 根用户（控制台）启用虚拟 MFA 设备。

创建管理用户

1. 启用 IAM Identity Center

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的启用 AWS IAM Identity Center。
2. 在 IAM Identity Center 中，为管理用户授予管理访问权限。

   有关使用 IAM Identity Center 目录 作为身份源的教程，请参阅《用户指南》 IAM Identity Center 目录中的使用默认设置配置AWS IAM Identity Center 用户访问权限。

作为管理用户登录

- 要使用您的 IAM Identity Center 用户身份登录，请使用您在创建 IAM Identity Center 用户时发送到您的电子邮件地址的登录网址。

  有关使用 IAM Identity Center 用户登录的帮助，请参阅AWS 登录 用户指南中的登录 AWS 访问门户。

为 CI/CD 集成配置 IAM 角色

要将 Amazon Inspector 扫描集成到 CI/CD 管道中，您需要创建一项 IAM 策略，允许访问扫描软件物料清单（SBOM）的 Amazon Inspector Scan API。然后，您可以将该策略附加到 IAM 角色，让您的账户可以运行 Amazon Inspector Scan API。

1. 登录 AWS Management Console 并打开 IAM 控制台，网址为 https://console.aws.amazon.com/iam/。
2. 在 IAM 控制台的导航窗格中，选择策略，然后选择创建策略。
3. 在策略编辑器中，选择 JSON，然后粘贴下列语句：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "inspector-scan:ScanSbom",
            "Resource": "*"
        }
    ]
}
```

4. 选择下一步。
5. 为策略命名（如 `InspectorCICDscan-policy`），添加可选描述，然后选择创建策略。此策略将附加到后续步骤中创建的角色。
6. 在 IAM 控制台的导航窗格中，依次选择角色和创建新角色。
7. 对于可信实体类型，选择自定义信任策略，然后粘贴以下策略：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::{ACCOUNT_ID}:root"
            },
            "Action": "sts:AssumeRole",
            "Condition": {}
        }
    ]
}
```

8. 选择下一步。

9. 在添加权限页面上，搜索并选择您之前创建的策略，然后选择下一步。
10. 为角色命名（如 `InspectorCICDscan-role`），添加可选描述，然后选择`Create Role`。

Amazon Inspector SBOM 生成器

Amazon Inspector SBOM 生成器（Sbomgen）是一种二进制工具，可为容器映像生成软件物料清单（SBOM）。SBOM 是系统上安装的软件的收集清单。

Sbomgen 的工作原理是扫描已知包含已安装程序包信息的文件。如果找到其中一个文件，该工具就会提取程序包名称、版本和其他元数据。然后，它会将此程序包元数据转换为 CycloneDX SBOM。

Sbomgen 可作为独立工具使用，以文件或 STDOUT 的形式提供 CycloneDX SBOM。它也是 Amazon Inspector CI/CD 集成的一部分，在部署管道中自动扫描容器映像。有关更多信息，请参阅将 Amazon Inspector 扫描集成到 CI/CD 管道中。

支持的程序包和映像格式

目前，Sbomgen 可以为以下类型的程序包收集清单：

- Alpine APK
- Debian / Ubuntu DPKG
- Red Hat RPM
- 通过 `go.mod` 和 `go mod cache` 的 Go 程序包
- 通过 `pom.properties` 的 Java 程序包
- 通过 `node_modules` 内的 `package.json` 文件的 Node.js 程序包
- 通过 Nuget 文件（`.deps.json`、`csproj`、`Packages.config`、packages.lock.json）的 C# 程序包
- 通过 `installed.json` 和 `composer.lock` 的 PHP
- 通过 `requirements.txt`、`Pipfile.lock`、`poetry.lock`、和 `egg/wheel` 文件的 Python 程序包
- 通过 `Gemfile.lock`、`.gemspec` 和全局安装的 Gem 的 Ruby 程序包
- 通过 `Cargo.lock` 和 `Cargo.toml` 的 Rust 程序包

Sbomgen 支持映像的以下容器镜像清单格式：

- OCI 映像清单
- Docker Image Manifest V2 Schema 2
- Docker Image Manifest V2 Schema 1
- Docker Image Manifest V1

> **Important**
>
> 如果容器映像的大小超过 5 GB、层数超过 60 或已安装的程序包超过 2000 个，则 Sbomgen 无法对其进行扫描。

安装 Amazon Inspector SBOM 生成器（Sbomgen）

Sbomgen 仅适用于 Linux 操作系统。如果您使用它来分析容器映像，则必须安装容器服务，例如 Docker、 Podman 或 containerd。

为了获得最佳性能，我们建议在具有以下最低硬件规格的系统上运行二进制文件：

- 4 核 CPU
- 8 GB RAM

要安装 Sbomgen，请执行以下操作

1. 从架构的正确 URL 中下载 Sbomgen zip 文件：

   Linux AMD64：

   https://amazon-inspector-sbomgen.s3.amazonaws.com/latest/linux/amd64/inspector-sbomgen.zip

   Linux ARM64：

   https://amazon-inspector-sbomgen.s3.amazonaws.com/latest/linux/arm64/inspector-sbomgen.zip
2. 使用以下命令解压缩下载的文件：

   ```
   unzip inspector-sbomgen.zip
   ```
3. 检查归档中是否包含以下文件：

- `inspector-sbomgen` – 这是为生成 SBOM 而要执行的二进制文件。
- `README.txt` – 这是说明如何使用 Sbomgen 的文档。
- `LICENSE.txt` – 此文件包含 Sbomgen 的软件许可证。
- `licenses` – 此文件夹包含 Sbomgen 所使用的第三方程序包的许可证信息。
- `checksums.txt` – 此文件提供 Sbomgen 二进制文件的哈希值。
- `sbom.json` – 这是 Sbomgen 二进制文件的 CycloneDX SBOM。

4. （可选）使用以下命令验证该二进制文件的真实性和完整性：

   ```
   sha256sum < inspector-sbomgen
   ```

   - 将结果与 `checksums.txt` 文件的内容进行比较。

5. 使用以下命令为该二进制文件授予可执行权限。

   ```
   chmod +x inspector-sbomgen
   ```

6. 使用以下命令验证是否已成功安装 Sbomgen：

   ```
   ./inspector-sbomgen --version
   ```

   您应该可以看到类似于如下所示的输出内容：

   ```
   Version: 1.X.X
   ```

使用 Sbomgen

您可以使用 Sbomgen 为容器映像生成 SBOM。

您还可以通过排除特定文件或定义工具扫描哪些程序包等选项来自定义 SBOM 生成的结果。有关这些用例的示例以及更多内容，请运行以下命令：

```
./inspector-sbomgen list-examples
```

为容器映像生成 SBOM 并将结果输出至文件

对于这个示例，请将 *`image:tag`* 替换为映像的 ID，将 *`output_path.json`* 替换为要保存输出内容的路径：

```
./inspector-sbomgen container --image image:tag -o output_path.json
```

使用 Sbomgen 向私有注册表进行身份验证

您可以通过提供私有注册表身份验证凭证，从托管在私有注册表中的容器生成 SBOM。您可以通过多种方式提供凭证：通过缓存的凭证、通过交互式方法，或通过非交互式方法（在运行 Sbomgen 之前将凭据作为环境变量提供）。

使用缓存的凭证进行身份验证（推荐）

1. 如果代理上有缓存的凭证，Sbomgen 将尝试使用缓存的凭据。对于此方法，请先向容器注册表进行身份验证。例如，如果您使用的是 Docker，则可以使用 Docker `login` 命令向注册表进行身份验证：

   ```
   docker login
   ```

2. 然后，成功向私有注册表进行身份验证后，即可在该注册表中的容器映像上使用 Sbomgen。要使用以下示例，请将 *image:tag* 替换为要扫描的映像的名称：

   ```
   ./inspector-sbomgen container --image image:tag
   ```

使用交互式方法进行身份验证

- 对于这种方法，您需要提供用户名作为参数，Sbomgen 会在需要时提示您输入安全密码。要使用以下示例，请将 *image:tag* 替换为要扫描的映像的名称，将 *your_username* 替换为有权访问该映像的用户名：

  ```
  ./inspector-sbomgen container --image image:tag --username
  your_username
  ```

使用非交互式方法进行身份验证

- 要使用此方法，您应将密码或注册表令牌存储在只有当前用户才能读取的 .txt 文件中。该文本文件应只包含一行密码或令牌。要使用以下示例，请将 *your_username* 替换为您的用户名，将 *password.txt* 替换为包含密码或令牌的文件，将 *image:tag* 替换为要扫描的映像的名称：

  ```
  INSPECTOR_SBOMGEN_USERNAME=your_username\
  INSPECTOR_SBOMGEN_PASSWORD=`cat password.txt` \
  ./inspector-sbomgen container --image image:tag
  ```

来自 Sbomgen 的输出内容示例

下面是使用 Sbomgen 清点的容器映像的 SBOM 示例。

容器映像 SBOM


```
{
  "bomFormat": "CycloneDX",
  "specVersion": "1.5",
  "serialNumber": "urn:uuid:828875ef-8c32-4777-b688-0af96f3cf619",
  "version": 1,
  "metadata": {
    "timestamp": "2023-11-17T21:36:38Z",
    "tools": [
      {
        "vendor": "Amazon Web Services, Inc. (AWS)",
        "name": "Amazon Inspector SBOM Generator",
        "version": "1.0.0",
        "hashes": [
          {
            "alg": "SHA-256",
            "content":
 "10ab669cfc99774786301a745165b5957c92ed9562d19972fbf344d4393b5eb1"
          }
        ]
      }
    ],
    "component": {
      "bom-ref": "comp-1",
      "type": "container",
      "name": "fedora:latest",
      "properties": [
        {
          "name": "amazon:inspector:sbom_generator:image_id",
          "value":
 "sha256:c81c8ae4dda7dedc0711daefe4076d33a88a69a28c398688090c1141eff17e50"
        },
        {
          "name": "amazon:inspector:sbom_generator:layer_diff_id",
          "value":
 "sha256:eddd0d48c295dc168d0710f70364581bd84b1dda6bb386c4a4de0b61de2f2119"
        }
      ]
```

```
    }
  },
  "components": [
    {
      "bom-ref": "comp-2",
      "type": "library",
      "name": "dnf",
      "version": "4.18.0",
      "purl": "pkg:pypi/dnf@4.18.0",
      "properties": [
        {
          "name": "amazon:inspector:sbom_generator:source_file_scanner",
          "value": "python-pkg"
        },
        {
          "name": "amazon:inspector:sbom_generator:source_package_collector",
          "value": "python-pkg"
        },
        {
          "name": "amazon:inspector:sbom_generator:source_path",
          "value": "/usr/lib/python3.12/site-packages/dnf-4.18.0.dist-info/METADATA"
        },
        {
          "name": "amazon:inspector:sbom_generator:is_duplicate_package",
          "value": "true"
        },
        {
          "name": "amazon:inspector:sbom_generator:duplicate_purl",
          "value": "pkg:rpm/fedora/python3-dnf@4.18.0-2.fc39?
arch=noarch&distro=39&epoch=0"
        }
      ]
    },
    {
      "bom-ref": "comp-3",
      "type": "library",
      "name": "libcomps",
      "version": "0.1.20",
      "purl": "pkg:pypi/libcomps@0.1.20",
      "properties": [
        {
          "name": "amazon:inspector:sbom_generator:source_file_scanner",
          "value": "python-pkg"
        },
```

```
        {
          "name": "amazon:inspector:sbom_generator:source_package_collector",
          "value": "python-pkg"
        },
        {
          "name": "amazon:inspector:sbom_generator:source_path",
          "value": "/usr/lib64/python3.12/site-packages/libcomps-0.1.20-py3.12.egg-
info/PKG-INFO"
        },
        {
          "name": "amazon:inspector:sbom_generator:is_duplicate_package",
          "value": "true"
        },
        {
          "name": "amazon:inspector:sbom_generator:duplicate_purl",
          "value": "pkg:rpm/fedora/python3-libcomps@0.1.20-1.fc39?
arch=x86_64&distro=39&epoch=0"
        }
      ]
    }
  ]
}
```

使用 Amazon Inspector 扫描自行创建自定义 CI/CD 管道集成

如果您的 CI/CD 市场上提供 Amazon Inspector CI/CD 插件，我们建议您使用这些插件。有关可用插件的列表，请参阅支持的 CI/CD 解决方案。

如果 Amazon Inspector 未提供适用于您的 CI/CD 解决方案的插件，您可以结合使用 Amazon Inspector SBOM 生成器和 Amazon Inspector Scan API 来自行创建自定义 CI/CD 集成。您还可以使用 Amazon Inspector SBOM 生成器中提供的选项，借助自定义集成来微调扫描。

自行设置自定义集成

1. 配置 AWS 账户 以允许访问 Amazon Inspector 扫描 API。有关说明，请参阅设置 AWS 账户以使用 Amazon Inspector CI/CD 集成。
2. 安装和配置 Amazon Inspector SBOM 生成器二进制文件。有关说明，请参阅安装 Amazon Inspector SBOM 生成器（Sbomgen）。
3. 使用 SBOM 生成器为要扫描的容器映像创建 SBOM 文件。要使用以下示例，请将 *image:id* 替换为要扫描的映像的名称，将 *sbom_path.json* 替换为 SBOM 输出的保存位置：

```
./inspector-sbomgen container –image image:id -o sbom_path.json
```

4. 调用 inspector-scan API 以扫描生成的 SBOM 并提供漏洞报告。要使用以下示例，请将 *sbom_path.json* 替换为与 CycloneDX 兼容的有效 SBOM 文件的路径。*########## AWS ### ############ API ####### REGION #########*有关区域和端点的完整列表，请参阅 Amazon Inspector Scan API 的端点。

```
aws inspector-scan scan-sbom --sbom file://sbom_path.json --endpoint
"ENDPOINT" --region REGION
```

API 输出格式

Amazon Inspector Scan API 能够以 CycloneDX 1.5 格式或 Amazon Inspector 调查发现 JSON 格式输出漏洞报告。可以使用 --output-format 标志更改默认值。

CycloneDX 1.5 格式输出示例


```
{
  "status": "SBOM parsed successfully, 1 vulnerabilities found",
  "sbom": {
    "bomFormat": "CycloneDX",
    "specVersion": "1.5",
    "serialNumber": "urn:uuid:0077b45b-ff1e-4dbb-8950-ded11d8242b1",
    "metadata": {
      "properties": [
        {
          "name": "amazon:inspector:sbom_scanner:critical_vulnerabilities",
          "value": "1"
        },
        {
          "name": "amazon:inspector:sbom_scanner:high_vulnerabilities",
          "value": "0"
        },
        {
          "name": "amazon:inspector:sbom_scanner:medium_vulnerabilities",
          "value": "0"
        },
        {
          "name": "amazon:inspector:sbom_scanner:low_vulnerabilities",
          "value": "0"
        }
```

```
      ],
      "tools": [
        {
          "name": "CycloneDX SBOM API",
          "vendor": "Amazon Inspector",
          "version": "empty:083c9b00:083c9b00:083c9b00"
        }
      ],
      "timestamp": "2023-06-28T14:15:53.760Z"
    },
    "components": [
      {
        "bom-ref": "comp-1",
        "type": "library",
        "name": "log4j-core",
        "purl": "pkg:maven/org.apache.logging.log4j/log4j-core@2.12.1",
        "properties": [
          {
            "name": "amazon:inspector:sbom_scanner:path",
            "value": "/home/dev/foo.jar"
          }
        ]
      }
    ],
    "vulnerabilities": [
      {
        "bom-ref": "vuln-1",
        "id": "CVE-2021-44228",
        "source": {
          "name": "NVD",
          "url": "https://nvd.nist.gov/vuln/detail/CVE-2021-44228"
        },
        "references": [
          {
            "id": "SNYK-JAVA-ORGAPACHELOGGINGLOG4J-2314720",
            "source": {
              "name": "SNYK",
              "url": "https://security.snyk.io/vuln/SNYK-JAVA-
ORGAPACHELOGGINGLOG4J-2314720"
            }
          },
          {
            "id": "GHSA-jfh8-c2jp-5v3q",
            "source": {
```

```
              "name": "GITHUB",
              "url": "https://github.com/advisories/GHSA-jfh8-c2jp-5v3q"
            }
          }
        ],
        "ratings": [
          {
            "source": {
              "name": "NVD",
              "url": "https://www.first.org/cvss/v3-1/"
            },
            "score": 10.0,
            "severity": "critical",
            "method": "CVSSv31",
            "vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H"
          },
          {
            "source": {
              "name": "NVD",
              "url": "https://www.first.org/cvss/v2/"
            },
            "score": 9.3,
            "severity": "critical",
            "method": "CVSSv2",
            "vector": "AC:M/Au:N/C:C/I:C/A:C"
          },
          {
            "source": {
              "name": "EPSS",
              "url": "https://www.first.org/epss/"
            },
            "score": 0.97565,
            "severity": "none",
            "method": "other",
            "vector": "model:v2023.03.01,date:2023-06-27T00:00:00+0000"
          },
          {
            "source": {
              "name": "SNYK",
              "url": "https://security.snyk.io/vuln/SNYK-JAVA-
ORGAPACHELOGGINGLOG4J-2314720"
            },
            "score": 10.0,
            "severity": "critical",
```

```
            "method": "CVSSv31",
            "vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H/E:H"
          },
          {
            "source": {
              "name": "GITHUB",
              "url": "https://github.com/advisories/GHSA-jfh8-c2jp-5v3q"
            },
            "score": 10.0,
            "severity": "critical",
            "method": "CVSSv31",
            "vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H"
          }
        ],
        "cwes": [
          400,
          20,
          502
        ],
        "description": "Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security
 releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages,
 and parameters do not protect against attacker controlled LDAP and other JNDI related
 endpoints. An attacker who can control log messages or log message parameters can
 execute arbitrary code loaded from LDAP servers when message lookup substitution is
 enabled. From log4j 2.15.0, this behavior has been disabled by default. From version
 2.16.0 (along with 2.12.2, 2.12.3, and 2.3.1), this functionality has been completely
 removed. Note that this vulnerability is specific to log4j-core and does not affect
 log4net, log4cxx, or other Apache Logging Services projects.",
        "advisories": [
          {
            "url": "https://www.intel.com/content/www/us/en/security-center/advisory/
intel-sa-00646.html"
          },
          {
            "url": "https://support.apple.com/kb/HT213189"
          },
          {
            "url": "https://msrc-blog.microsoft.com/2021/12/11/microsofts-response-to-
cve-2021-44228-apache-log4j2/"
          },
          {
            "url": "https://logging.apache.org/log4j/2.x/security.html"
          },
          {
```

```
            "url": "https://www.debian.org/security/2021/dsa-5020"
          },
          {
            "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-479842.pdf"
          },
          {
            "url": "https://www.oracle.com/security-alerts/alert-cve-2021-44228.html"
          },
          {
            "url": "https://www.oracle.com/security-alerts/cpujan2022.html"
          },
          {
            "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-714170.pdf"
          },
          {
            "url": "https://lists.fedoraproject.org/archives/list/package-
announce@lists.fedoraproject.org/message/M5CSVUNV4HWZZXGOKNSK6L7RPM7BOKIB/"
          },
          {
            "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-397453.pdf"
          },
          {
            "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-661247.pdf"
          },
          {
            "url": "https://lists.fedoraproject.org/archives/list/package-
announce@lists.fedoraproject.org/message/VU57UJDCFIASIO35GC55JMKSRXJMCDFM/"
          },
          {
            "url": "https://www.oracle.com/security-alerts/cpuapr2022.html"
          },
          {
            "url": "https://twitter.com/kurtseifried/status/1469345530182455296"
          },
          {
            "url": "https://tools.cisco.com/security/center/content/
CiscoSecurityAdvisory/cisco-sa-apache-log4j-qRuKNEbd"
          },
          {
            "url": "https://lists.debian.org/debian-lts-announce/2021/12/msg00007.html"
          },
          {
            "url": "https://www.kb.cert.org/vuls/id/930724"
          }
```

```
        ],
        "created": "2021-12-10T10:15:00Z",
        "updated": "2023-04-03T20:15:00Z",
        "affects": [
          {
            "ref": "comp-1"
          }
        ],
        "properties": [
          {
            "name": "amazon:inspector:sbom_scanner:exploit_available",
            "value": "true"
          },
          {
            "name": "amazon:inspector:sbom_scanner:exploit_last_seen_in_public",
            "value": "2023-03-06T00:00:00Z"
          },
          {
            "name": "amazon:inspector:sbom_scanner:cisa_kev_date_added",
            "value": "2021-12-10T00:00:00Z"
          },
          {
            "name": "amazon:inspector:sbom_scanner:cisa_kev_date_due",
            "value": "2021-12-24T00:00:00Z"
          },
          {
            "name": "amazon:inspector:sbom_scanner:fixed_version:comp-1",
            "value": "2.15.0"
          }
        ]
      }
    ]
  }
}
```

Inspector 格式输出示例

```
                        {
  "status": "SBOM parsed successfully, 1 vulnerability found",
  "inspector": {
    "messages": [
      {
```

```
        "name": "foo",
        "purl": "pkg:maven/foo@1.0.0", // Will not exist in output if missing in sbom
        "info": "Component skipped: no rules found."
      }
    ],
    "vulnerability_count": {
      "critical": 1,
      "high": 0,
      "medium": 0,
      "low": 0
    },
    "vulnerabilities": [
      {
        "id": "CVE-2021-44228",
        "severity": "critical",
        "source": "https://nvd.nist.gov/vuln/detail/CVE-2021-44228",
        "related": [
          "SNYK-JAVA-ORGAPACHELOGGINGLOG4J-2314720",
          "GHSA-jfh8-c2jp-5v3q"
        ],
        "description": "Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security
 releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages,
 and parameters do not protect against attacker controlled LDAP and other JNDI related
 endpoints. An attacker who can control log messages or log message parameters can
 execute arbitrary code loaded from LDAP servers when message lookup substitution is
 enabled. From log4j 2.15.0, this behavior has been disabled by default. From version
 2.16.0 (along with 2.12.2, 2.12.3, and 2.3.1), this functionality has been completely
 removed. Note that this vulnerability is specific to log4j-core and does not affect
 log4net, log4cxx, or other Apache Logging Services projects.",
        "references": [
          "https://www.intel.com/content/www/us/en/security-center/advisory/intel-
sa-00646.html",
          "https://support.apple.com/kb/HT213189",
          "https://msrc-blog.microsoft.com/2021/12/11/microsofts-response-to-
cve-2021-44228-apache-log4j2/",
          "https://logging.apache.org/log4j/2.x/security.html",
          "https://www.debian.org/security/2021/dsa-5020",
          "https://cert-portal.siemens.com/productcert/pdf/ssa-479842.pdf",
          "https://www.oracle.com/security-alerts/alert-cve-2021-44228.html",
          "https://www.oracle.com/security-alerts/cpujan2022.html",
          "https://cert-portal.siemens.com/productcert/pdf/ssa-714170.pdf",
          "https://lists.fedoraproject.org/archives/list/package-
announce@lists.fedoraproject.org/message/M5CSVUNV4HWZZXGOKNSK6L7RPM7BOKIB/",
          "https://cert-portal.siemens.com/productcert/pdf/ssa-397453.pdf",
```

```
          "https://cert-portal.siemens.com/productcert/pdf/ssa-661247.pdf",
          "https://lists.fedoraproject.org/archives/list/package-
announce@lists.fedoraproject.org/message/VU57UJDCFIASIO35GC55JMKSRXJMCDFM/",
          "https://www.oracle.com/security-alerts/cpuapr2022.html",
          "https://twitter.com/kurtseifried/status/1469345530182455296",
          "https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-
sa-apache-log4j-qRuKNEbd",
          "https://lists.debian.org/debian-lts-announce/2021/12/msg00007.html",
          "https://www.kb.cert.org/vuls/id/930724"
        ],
        "created": "2021-12-10T10:15:00Z",
        "updated": "2023-04-03T20:15:00Z",
        "properties": {
          "cisa_kev_date_added": "2021-12-10T00:00:00Z",
          "cisa_kev_date_due": "2021-12-24T00:00:00Z",
          "cwes": [
            400,
            20,
            502
          ],
          "cvss": [
            {
              "source": "NVD",
              "severity": "critical",
              "cvss3_base_score": 10.0,
              "cvss3_base_vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H",
              "cvss2_base_score": 9.3,
              "cvss2_base_vector": "AC:M/Au:N/C:C/I:C/A:C"
            },
            {
              "source": "SNYK",
              "severity": "critical",
              "cvss3_base_score": 10.0,
              "cvss3_base_vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H/E:H"
            },
            {
              "source": "GITHUB",
              "severity": "critical",
              "cvss3_base_score": 10.0,
              "cvss3_base_vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H"
            }
          ],
          "epss": 0.97565,
          "exploit_available": true,
```

```
          "exploit_last_seen_in_public": "2023-03-06T00:00:00Z"
        },
        "affects": [
          {
            "installed_version": "pkg:maven/org.apache.logging.log4j/log4j-
core@2.12.1",
            "fixed_version": "2.15.0",
            "path": "/home/dev/foo.jar"
          }
        ]
      }
    ]
  }
}
```

使用 Amazon Inspector Jenkins 插件

该Jenkins插件利用 Amazon Inspector SBOM 生成器二进制文件和 Amazon Inspector Scan API 在构建结束时生成详细的报告，因此您可以在部署之前调查和修复风险。

Amazon Inspector 是一项漏洞管理服务，它基于 CVE 扫描容器映像中是否存在操作系统和编程语言包漏洞。

使用 Amazon Ins Jenkins pector 插件，您可以将 Amazon Inspector 漏洞扫描添加到您的Jenkins管道中。

> **Note**
>
> 可以根据检测到的漏洞数量和严重性，将 Amazon Inspector 漏洞扫描配置为通过或失败管道执行。

你可以在Jenkins市场上查看该Jenkins插件的最新版本，网址为 https://plugins.jenkins.io/amazon-inspector-image-scanner/。

以下步骤描述了如何设置 Amazon Ins Jenkins pector 插件。

> **Important**
>
> 在完成以下步骤之前，必须将 Jenkins 升级到 2.387.3 或更高版本才能运行该插件。

第 1 步。设置一个 AWS 账户

AWS 账户 使用允许访问 Amazon Inspector Scan API 的 IAM 角色进行配置。有关说明，请参阅设置 AWS 账户以使用 Amazon Inspector CI/CD 集成。

第 2 步。安装 Amazon Inspector Jenkins 插件

以下过程描述了如何从Jenkins控制面板安装 Amazon Inspector Jenkins 插件。

1. 在 Jenkins 控制面板中，选择 “管理 Jenkins”，然后选择 “管理插件”。
2. 选择 “可用”。
3. 从 “可用” 选项卡中搜索 Amazon Inspector 扫描件，然后安装该插件。

（可选）步骤 3。将 docker 凭据添加到 Jenkins

> **Note**
>
> 仅当 docker 镜像位于私有存储库中时，才添加 docker 凭据。否则，请跳过此步骤。

以下过程介绍如何Jenkins从控制面板向中添加 docker 凭证。Jenkins

1. 在 Jenkins 控制面板中，选择 “管理 Jenkins”、“凭据”，然后选择 “系统”。
2. 选择 “全局凭据”，然后选择 “添加凭据”。
3. 在 “种类” 中，选择带密码的用户名。
4. 对于 “范围”，选择 “全局”（Jenkins、节点、项目、所有子项目等）。
5. 输入您的详细信息，然后选择 “确定”。

（可选）步骤 4。添加 AWS 凭证

> **Note**
>
> 仅当您想要基于 IAM 用户进行身份验证时，才添加 AWS 证书。否则，请跳过此步骤。

以下过程介绍如何从Jenkins仪表板添加 AWS 凭据。

1. 在 Jenkins 控制面板中，选择 “管理 Jenkins”、“凭据”，然后选择 “系统”。
2. 选择 “全局凭据”，然后选择 “添加凭据”。
3. 对于种类，请选择 AWS 证书。
4. 输入您的详细信息，包括您的访问密钥 ID 和私有访问密钥，然后选择 OK。

第 5 步。在Jenkins脚本中添加 CSS 支持

以下过程介绍如何在Jenkikns脚本中添加 CSS 支持。

1. 重启 Jenkins。
2. 在控制面板中，选择管理 Jenkins、节点、内置节点，然后选择脚本控制台。
3. 在文本框中，添加该行`System.setProperty("hudson.model.DirectoryBrowserSupport.CSP", "")`，然后选择 “运行”。

第 6 步。将 Amazon Inspector Scan 添加到你的版本中

您可以通过在项目中添加构建步骤或使用 Jenkins 声明式管道，将 Amazon Inspector 扫描添加到您的构建中。

Amazon Inspector 通过在项目中添加构建步骤来扫描您的版本

1. 在配置页面上，向下滚动到生成步骤，然后选择添加构建步骤。然后选择 Amazon Inspector 扫描。
2. 在两种 inspector-sbomgen 安装方法之间进行选择：自动或手动。

   a. （选项 1）选择 “自动” 以下载 inspector-sbomgen 的最新版本。如果选择此方法，请确保选择与执行插件的系统相匹配的 CPU 架构。

b. （选项 2）如果您要设置 Amazon Inspector SBOM 生成器二进制文件进行扫描，请选择 “手动”。如果您选择这种方法，请确保提供之前下载的 inspector-sbomgen 版本的完整路径。

有关更多信息，请参阅在 Amazon Inspector SBOM 生成器中安装 Amazon Inspector SBOM 生成器（Sbomgen）。

3. 完成以下操作以完成 Amazon Inspector 扫描构建步骤的配置：

a. 输入映像 ID。映像可以是本地映像、远程映像或归档映像。映像名称应遵循 Docker 命名约定。如果要分析导出的映像，请提供预期的 tar 文件的路径。请参阅下列示例映像 ID 路径：

i. 对于本地或远程容器：`NAME[:TAG|@DIGEST]`

ii. 对于 tar 文件：`/path/to/image.tar`

b. 选择用于发送扫描请求的 AWS 区域。

c. （可选）对于 Docker 凭证，请选择您的 Docker 用户名。仅当容器映像位于私有存储库中时才执行此操作。

d. （可选）您可以提供以下支持的 AWS 身份验证方法：

i. （可选）对于 IAM 角色，请提供角色 ARN (arn: aws: iam:: role/)。*AccountNumberRoleName*

ii. （可选）对于 AWS 证书，请选择要基于 IAM 用户进行身份验证的 ID。

iii. （可选）对于AWS 配置文件名称，请提供要使用配置文件名称进行身份验证的配置文件名称。

e. （可选）指定每种严重性的漏洞阈值。如果扫描期间的漏洞数超过了您指定的数量，则映像构建将失败。如果值全部为 `0`，则无论是否发现任何漏洞，构建都将成功。

4. 选择保存。

使用Jenkins声明式管道将 Amazon Inspector Scan 添加到你的版本中

你可以使用 Jenkins 声明式管道自动或手动将 Amazon Inspector Scan 添加到你的版本中。

自动下载 sbomGen 声明式管道

- 要将 Amazon Inspector Scan 添加到版本中，请使用以下示例语法。根据你首选的 Amazon Inspector SBOM 生成器下载操作系统架构，用 LinuxAMD64 或 LinuxArm64 替换 *SBOMGEN_SOUR* CE。将 IMA *GE_PATH* 替换为图片的路径（例如 *alpine: l* atest），将

IAM_ROLE #换为您在步骤 1 中配置的 IAM 角色的 ARN，如果您使用的是私有存储库，请Docker将 ID 替换为凭# ID。您可以选择启用漏洞阈值并为每个严重性指定值。

```
pipeline {
   agent any
   stages {
       stage('amazon-inspector-image-scanner') {
           steps {
               script {
               step([
               $class:
 'com.amazon.inspector.jenkins.amazoninspectorbuildstep.AmazonInspectorBuilder',
               sbomgenSource: 'SBOMGEN_SOURCE', // this can be linuxAmd64 or linuxArm64
               archivePath: 'IMAGE_PATH',
               awsRegion: 'REGION',
               iamRole: 'IAM ROLE',
               credentialId: 'Id', // provide empty string if image not in private
 repositories
               awsCredentialId: ''AWS ID;',
               awsProfileName: 'Profile Name',
               isThresholdEnabled: false,
               countCritical: 0,
               countHigh: 0,
               countLow: 10,
               countMedium: 5,
              ])
           }
         }
       }
   }
 }
```

手动下载 sbomGen 声明式管道

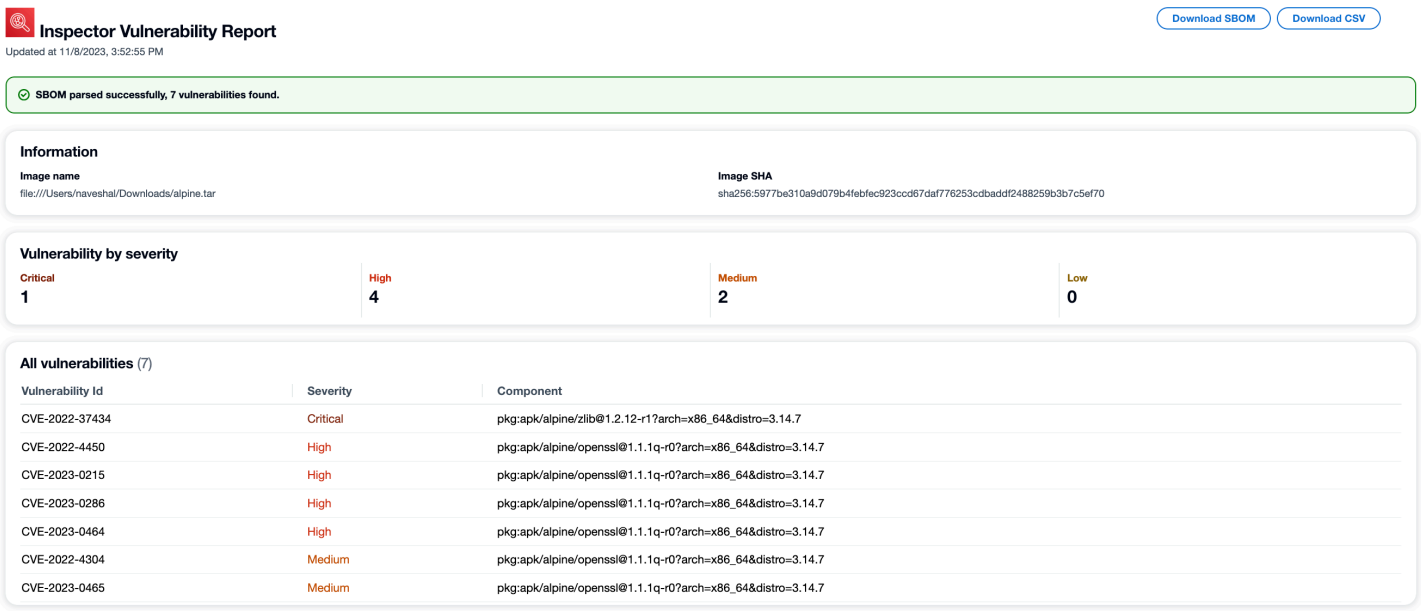
- 要将 Amazon Inspector Scan 添加到版本中，请使用以下示例语法。*# SBOMGEN_PATH #### ### 3 #### Amazon Inspector SBOM ######## IMAGE_ PATH ########## alp ine: latest### IAM_ROLE ####### 1 #### IAM ### ARN################ ID # #### ID#* Docker您可以选择启用漏洞阈值并为每个严重性指定值。

> **Note**
>
> 放Sbomgen在 Jenkins 目录中，然后在插件中提供 Jenkins 目录的路径（例如 /opt/folder/arm64/inspector-sbomgen）。

```
pipeline {
    agent any
    stages {
        stage('amazon-inspector-image-scanner') {
            steps {
                script {
                step([
                $class:
 'com.amazon.inspector.jenkins.amazoninspectorbuildstep.AmazonInspectorBuilder',
                sbomgenPath: 'SBOMGEN_PATH',
                archivePath: 'IMAGE_PATH',
                awsRegion: 'REGION',
                iamRole: 'IAM ROLE',
                awsCredentialId: ''AWS ID;',
                credentialId: 'Id;', // provide empty string if image not in private
 repositories
                awsProfileName: 'Profile Name',
                isThresholdEnabled: false,
                countCritical: 0,
                countHigh: 0,
                countLow: 10,
                countMedium: 5,
               ])
            }
          }
        }
    }
 }
```

第 7 步。查看您的 Amazon Inspector 漏洞报告

1. 完成项目的新构建。
2. 生成完成后，从结果中选择一种输出格式。如果您选择 HTML，则可以选择下载 JSON SBOM 或 CSV 版本的报告。以下显示了 HTML 报告的示例：



故障排除

以下是您在使用 Amazon Inspector Scan 插件时可能遇到的常见错误Jenkins。

加载凭证失败或 sts 异常错误

错误：

```
InstanceProfileCredentialsProvider(): Failed to load credentials or sts exception.
```

分辨率

aws_secret_access_key获取aws_access_key_id并使用您的 AWS 帐户。在 ~/.aws/credentials 中设置 aws_access_key_id 和 aws_secret_access_key。

检查员-sbomgen 路径错误

错误：

```
Exception:com.amazon.inspector.jenkins.amazoninspectorbuildstep.exception.Sbomge
There was an issue running inspector-sbomgen, is /opt/inspector/inspector-sbomgen the correct path?
```

解决方案：

完成以下步骤以解决问题。

1. 将正确的操作系统架构 Inspector-sbomgen 放在Jenkins目录中有关更多信息，请参阅 Amazon Inspector SBOM 生成器。
2. 使用以下命令向二进制文件授予可执行权限：`chmod +x inspector-sbomgen`。
3. 在插件中提供正确的Jenkins机器路径，例如`/opt/folder/arm64/inspector-sbomgen`。
4. 保存配置并执行Jenkins作业。

使用 Amazon Inspector TeamCity 插件

通过 Amazon Inspector TeamCity 插件，您可以将 Amazon Inspector 漏洞扫描添加到 TeamCity 管道中。该插件利用 Amazon Inspector SBOM 生成器二进制文件和 Amazon Inspector Scan API 在构建结束时生成详细的报告，这样您就可以在部署之前调查和修复风险。此外，还可以根据检测到的漏洞数量和严重性将扫描配置为使管道执行通过或失败。

Amazon Inspector 是一项漏洞管理服务 AWS ，它基于 CVE 扫描容器映像中是否存在操作系统和编程语言包漏洞。有关 Amazon Inspector CI/CD 集成的更多信息，请参阅将 Amazon Inspector 扫描集成到 CI/CD 管道中。

有关 Amazon Inspector 插件支持的程序包和容器映像格式的列表，请参阅支持的程序包和映像格式。

你可以在TeamCity市场上查看该插件的最新版本，网址为 https://plugins.jetbrains.com/plugin/23236-amazon-inspector-scanner。或者，可以按照本文档每个部分中的步骤设置 Amazon Inspector TeamCity 插件：

1. 设置一个 AWS 账户.
   - AWS 账户 使用允许访问 Amazon Inspector Scan API 的 IAM 角色进行配置。有关说明，请参阅设置 AWS 账户以使用 Amazon Inspector CI/CD 集成。
2. 安装 Amazon Inspector TeamCity 插件。
   a. 在控制面板中，前往管理 > 插件。
   b. 搜索 Amazon Inspector 扫描。
   c. 安装 插件。
3. 安装 Amazon Inspector SBOM 生成器。
   - 在 Teamcity 服务器目录中安装 Amazon Inspector SBOM 生成器二进制文件。有关说明，请参阅安装 Amazon Inspector SBOM 生成器（Sbomgen）。
4. 将 Amazon Inspector 扫描构建步骤添加到项目中。

a. 在配置页面上，向下滚动到 “构建步骤”，选择 “添加构建步骤”，然后选择 Amazon Inspector Scan。

b. 通过填写以下详细信息来配置 Amazon Inspector 扫描构建步骤：

- 添加步骤名称。
- 在两种 Amazon Inspector SBOM 生成器安装方法之间进行选择：自动或手动。
  - 根据您的系统和 CPU 架构，自动下载最新版本的 Amazon Inspector SBOM 生成器。
  - 手册要求您提供之前下载的 Amazon Inspector SBOM 生成器的完整路径。

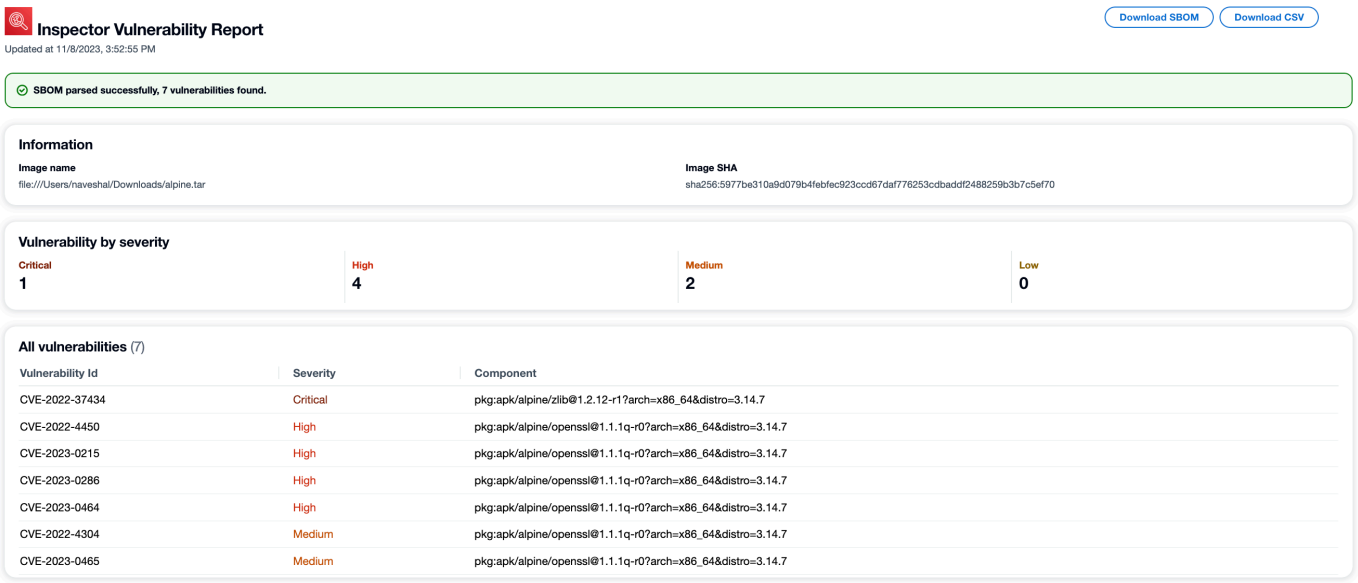
  有关更多信息，请参阅在亚马逊 Inspector SBOM 生成器中安装 Amazon Inspector SBOM 生成器 (Sbomgen)。
- 输入映像 ID。映像可以是本地映像、远程映像或归档映像。映像名称应遵循 Docker 命名约定。如果要分析导出的映像，请提供预期的 tar 文件的路径。请参阅下列示例映像 ID 路径：
  - 对于本地或远程容器：`NAME[:TAG|@DIGEST]`
  - 对于 tar 文件：`/path/to/image.tar`
- 对于 IAM 角色，输入您在步骤 1 中配置的角色的 ARN。
- 选择用于发送扫描请求的 AWS 区域。
- （可选）对于 Docker 身份验证，请输入您的 Docker 用户名和 Docker 密码。仅当容器映像位于私有存储库中时才执行此操作。
- （可选）对于AWS 身份验证，请输入您的 AWS 访问密钥 ID 和 AWS 密钥。只有在您想要根据 AWS 凭据进行身份验证时才执行此操作。
- （可选）指定每种严重性的漏洞阈值。如果扫描期间的漏洞数超过了您指定的数量，则映像构建将失败。如果值全部为 `0`，则无论发现多少漏洞，构建都将成功。

c. 选择保存。

5. 查看 Amazon Inspector 漏洞报告。

a. 完成项目的新构建。

b. 构建完成后，从结果中选择一种输出格式。如果选择 HTML，您可以选择下载 JSON SBOM 或 CSV 版本的报告。以下是一个 HTML 报告的示例：



Amazon Inspector CycloneDX 命名空间

Amazon Inspector 保留了 CycloneDX 命名空间和属性名称，供 Amazon Inspector SBOM 生成器和 Amazon Inspector Scan API 生成的 SBOM 使用。本页面记录了可以添加到使用 Amazon Inspector 工具创建的 CycloneDX SBOM 中组件的所有自定义键/值属性。有关 CycloneDX 属性分类的更多信息，请参阅官方文档。

`amazon:inspector:sbom_scanner` 命名空间分类

`amazon:inspector:sbom_scanner` 命名空间供 Amazon Inspector Scan API 使用。它具有以下属性：

属性	说明
`amazon:inspector:sbom_scanner:critical_vulnerabilities`	在 SBOM 中发现的严重性为“严重”的漏洞总数。
`amazon:inspector:sbom_scanner:high_vulnerabilities`	在 SBOM 中发现的严重性为“高”的漏洞总数。
`amazon:inspector:sbom_scanner:medium_vulnerabilities`	在 SBOM 中发现的严重性为“中”的漏洞总数。

属性	说明
amazon:inspector:sbom_scanner:low_vulnerabilities	在 SBOM 中发现的严重性为“低”的漏洞总数。
amazon:inspector:sbom_scanner:info	为给定组件提供扫描上下文，例如：“已扫描组件：未发现漏洞。”
amazon:inspector:sbom_scanner:warning	为未扫描给定组件的原因提供上下文，例如：“已跳过组件：未提供 purl。”
amazon:inspector:sbom_scanner:fixed_version: *component_bom_ref*	为给定漏洞提供指定组件已修复漏洞的版本。
amazon:inspector:sbom_scanner:exploit_available	表示是否存在针对给定漏洞的漏洞利用。
amazon:inspector:sbom_scanner:exploit_last_seen_in_public	表示针对给定漏洞的漏洞利用最后一次公开出现的时间。
amazon:inspector:sbom_scanner:cisa_kev_date_added	表示相应漏洞何时被添加到 CISA 已知被利用的漏洞目录中。
amazon:inspector:sbom_scanner:cisa_kev_date_due	表示根据 CISA 已知被利用的漏洞目录，漏洞修复程序的到期时间。
amazon:inspector:sbom_scanner:path	生成主题程序包信息的文件的路径。

amazon:inspector:sbom_generator 命名空间分类

amazon:inspector:sbom_generator 命名空间供 Amazon Inspector SBOM 生成器使用。它具有以下属性：

属性	说明
amazon:inspector:sbom_generator:os_hostname	正在清点的系统的主机名。
amazon:inspector:sbom_generator:kernel_name	正在清点的系统的内核名。
amazon:inspector:sbom_generator:kernel_version	正在清点的系统的内核版本。
amazon:inspector:sbom_generator:cpu_architecture	正在清点的系统的 CPU 架构，例如 x86_64。
amazon:inspector:sbom_generator:image_id	容器映像的配置文件的哈希值，也称为映像 ID。
amazon:inspector:sbom_generator:layer_diff_id	未压缩的容器映像层的哈希值。
amazon:inspector:sbom_generator:source_file_scanner	找到了包含程序包信息的文件的扫描器，例如：/var/lib/dpkg/status 。
amazon:inspector:sbom_generator:source_package_collector	从特定文件中提取了程序包名称和版本的收集器。
amazon:inspector:sbom_generator:source_path	从中提取了主题程序包信息的文件的路径。
amazon:inspector:sbom_generator:is_duplicate_package	表示主题程序包是由多个文件扫描器找到的。
amazon:inspector:sbom_generator:go_toolchain	表示用于生成 Go 可执行文件的 Go 编译器或工具链版本。
amazon:inspector:sbom_generator:expires_before	SSL 证书有效之前的日期。

属性	说明
amazon:inspector:sbom_generator:expires_after	SSL 证书在此日期之后无效。
amazon:inspector:sbom_generator:is_expired	一个布尔值，用于指示 SSL 证书是否已过期。

使用 Amazon Inspector 自动扫描资源

适用于 Amazon EC2 的 Amazon Inspector 无代理扫描目前为预览版。您对无代理 Amazon EC2 扫描功能的使用受 AWS 服务条款第 2 部分（“测试版和预览”）的约束。

Amazon Inspector 使用自己的专用扫描引擎。该引擎会监控资源中是否存在软件脆弱性或开放的网络路径，这些问题可能导致工作负载受损、资源被恶意使用或未经授权访问您的数据。当 Amazon Inspector 检测到脆弱性时，它会生成一个调查发现。调查发现包括与检测相关的详细信息，可帮助您修复脆弱性。你可以在 Amazon Inspector 控制台上和使用 Amazon Inspector API 查看调查发现。有关更多信息，请参阅管理 Amazon Inspector 中的调查发现。

激活后，Amazon Inspector 会自动发现所有符合条件的资源，并开始对这些资源进行持续扫描。Amazon Inspector 可以扫描软件脆弱性和意外网络暴露。Amazon Inspector 还会针对事件运行扫描，例如安装新应用程序或补丁。

首次激活 Amazon Inspector 时，您的账户会自动注册所有扫描类型。以下主题介绍了有关 Amazon Inspector 提供的扫描类型的具体细节。Amazon Inspector 根据受脆弱性影响的资源类型对扫描类型进行分类。以下主题介绍了 Amazon Inspector 扫描哪些资源、重新扫描这些资源的原因以及如何为每种资源类型配置扫描。

主题

- Amazon Inspector 扫描类型概述
- 激活扫描类型
- 使用 Amazon Inspector 扫描 Amazon EC2 实例
- 使用 Amazon Inspector 扫描 Amazon ECR 容器映像
- 使用 Amazon Inspector 进行扫描 AWS Lambda
- 停用扫描类型

首次激活 Amazon Inspector 时，您的账户会自动注册以下扫描类型：Amazon EC2 扫描、Amazon ECR 扫描、Lambda 标准扫描。Lambda 代码扫描是 Lambda 函数扫描的可选层，您可以随时激活该扫描。

Amazon Inspector 扫描类型概述

Amazon Inspector 提供了一系列不同的扫描类型，重点关注您 AWS 环境中的特定资源类型。

Amazon EC2 扫描

激活 Amazon EC2 扫描后，Amazon Inspector 会扫描您的 Amazon EC2 实例，查找操作系统程序包和编程语言包脆弱性并检查网络可达性。Amazon Inspector 会扫描您的 EC2 实例，看其是否存在常见漏洞和风险（CVE）以及网络泄露问题。Amazon Inspector 使用实例上安装的 SSM 代理或通过实例的 Amazon EBS 快照执行扫描。有关 Amazon EC2 扫描的更多信息，请参阅使用 Amazon Inspector 扫描 Amazon EC2 实例。

Amazon ECR 扫描

当您激活 Amazon ECR 扫描时，Amazon Inspector 会将您的私有注册表中的所有基本扫描容器存储库转换为具有持续扫描功能的增强型扫描。您也可以选择将此设置配置为仅在推送时扫描或通过包含规则扫描特定存储库。最初扫描过去 30 天内推送或最近 90 天内提取的所有图像。默认情况下，Amazon Inspector 会在 90 天内继续监控图像，此设置可以随时更改。有关 Amazon ECR 扫描的更多信息，请参阅使用 Amazon Inspector 扫描 Amazon ECR 容器映像。

Lambda 标准扫描

激活 Lambda 标准扫描后，Amazon Inspector 会发现您账户中的 Lambda 函数并立即开始扫描这些函数以查找脆弱性。Amazon Inspector 会在部署新的 Lambda 函数和层时对其进行扫描，并在它们更新或新的常见脆弱性和风险 (CVE) 发布时对其进行重新扫描。有关 Lambda 函数扫描的更多信息，请参阅使用 Amazon Inspector 进行扫描 AWS Lambda。

Lambda 标准扫描 + Lambda 代码扫描

该选项结合了 Lambda 标准扫描和 Lambda 代码扫描。激活 Lambda 代码扫描后，Amazon Inspector 会发现您账户中的 Lambda 函数和层，并扫描您的应用程序包依赖项中是否存在代码脆弱性。Lambda 代码扫描会扫描 Lambda 函数中的自定义应用程序代码，以查找代码脆弱性。必须同时激活这两种扫描类型。有关更多信息，请参阅 Amazon Inspector Lambda 代码扫描。

激活扫描类型

您可以随时激活新的 Amazon Inspector 扫描类型。激活扫描类型后，Amazon Inspector 将立即开始扫描符合该扫描类型的资源。有关可用扫描类型的概述，请参阅Amazon Inspector 扫描类型概述。以下内容介绍了首次激活每种扫描类型时会发生的情况：

- Amazon EC2 扫描 — 为账户激活 Amazon Inspector Amazon EC2 扫描后，Amazon Inspector 会扫描您账户中所有符合条件的实例，以查找程序包脆弱性和网络可达性问题。Amazon Inspector SSM 插件已安装在所有由 SSM 管理Windows的主机上。有关更多信息，请参阅扫描 Windows 实例。此外，Amazon Inspector 会在您的账户中创建以下 SSM 关联：
  - `InspectorDistributor-do-not-delete`
  - `InspectorInventoryCollection-do-not-delete`
  - `InspectorLinuxDistributor-do-not-delete`
  - `InvokeInspectorLinuxSsmPlugin-do-not-delete`
  - `InvokeInspectorSsmPlugin-do-not-delete`.
- Amazon ECR 扫描 — 为账户激活 Amazon ECR 容器映像扫描后，该账户中私有存储库的 Amazon ECR 扫描类型将从使用 Amazon ECR 进行基本扫描更改为使用 Amazon Inspector 进行增强扫描。然后，对过去 30 天内推送或过去 90 天内提取的所有符合条件的 Amazon ECR 容器镜像进行包裹漏洞扫描。此外，您的 Amazon ECR 重新扫描持续时间设置为 90 天，以确定图片推送和提取日期。
- Lambda 标准扫描 — 为账户激活 Lambda 标准扫描后，将对您账户中过去 90 天内调用或更新的所有 Lambda 函数进行程序包脆弱性扫描。此外，还会在您的账户中创建 CloudTrail服务关联频道。
- Lambda 标准扫描 + Lambda 代码扫描 — 这些 Lambda 函数扫描类型需要同时激活。为账户激活 Lambda 代码扫描后，将对您账户中过去 90 天内调用或更新的所有 Lambda 函数进行代码脆弱性扫描。

激活扫描

如果您是 AWS 组织中亚马逊 Inspector 的委托管理员，则可以使用由 Amazon Inspector inspect or2-enablement-with-cli on 开发的 shell 脚本自动为多个区域的多个账户启用各种 Amazon Inspect or 扫描类型。 GitHub否则，要通过控制台在多账户环境中完成此程序，请在以 Amazon Inspector 委托管理员身份登录后完成以下步骤。

Console

激活扫描

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的选择 AWS 区域 器，选择要激活新扫描类型的区域。
3. 在导航窗格中，选择账户管理。
4. 在账户管理页面上，选择要为其激活扫描类型的账户。

5. 选择激活，然后选择要激活的扫描类型。
6. （推荐）在要激活该扫描类型的每个 AWS 区域 步骤中重复这些步骤。

API

运行启用 API 操作。在请求中，提供您要激活扫描的账户 ID、幂等性令牌，以及一个或多个 EC2、ECR、LAMBDA 或 LAMBDA_CODE，以便 resourceTypes 激活相应的扫描类型。

使用 Amazon Inspector 扫描 Amazon EC2 实例

适用于 Amazon EC2 的 Amazon Inspector 无代理扫描目前为预览版。您对无代理 Amazon EC2 扫描功能的使用受 AWS 服务条款第 2 部分（“测试版和预览”）的约束。

Amazon Inspector EC2 扫描会从 EC2 实例提取元数据，然后将这些元数据与从安全公告中收集的规则进行比较，以得出调查发现。Amazon Inspector 会扫描实例中是否存在程序包漏洞和网络可访问性问题。有关针对这些问题得出的调查发现类型的信息，请参阅Amazon Inspector 中的调查发现类型。

Amazon Inspector 每 24 小时执行一次网络可访问性扫描，而程序包漏洞扫描的执行节奏各不相同，具体取决于与实例关联的扫描方法。

扫描方法

程序包漏洞扫描可以使用基于代理或无代理的扫描方法执行。这些扫描方法决定了 Amazon Inspector 如何以及何时从 EC2 实例收集软件清单，以进行程序包漏洞扫描。基于代理的方法依赖 SSM 代理来收集软件清单，而无代理方法使用 Amazon EBS 快照（而不是代理）。

Amazon Inspector 使用的扫描方法取决于您账户的扫描模式设置。有关更多信息，请参阅管理扫描模式。

要激活 Amazon EC2 扫描，请参阅激活扫描类型。

基于代理的扫描

在所有符合条件的实例上，我们使用 SSM 代理持续执行基于代理的扫描。对于基于代理的扫描，Amazon Inspector 使用 SSM 关联以及通过这些关联安装的插件从实例收集软件清单。除了对操作系统程序包进行程序包漏洞扫描外，Amazon Inspector 基于代理的扫描还可以通过Amazon EC2 Linux 实例的 Amazon Inspector 深度检查，检测基于 Linux 的实例中的应用程序编程语言包是否存在程序包漏洞。

以下过程说明了 Amazon Inspector 如何使用 SSM 收集清单和执行基于代理的扫描：

1. Amazon Inspector 会在您的账户中创建 SSM 关联，以便从实例中收集清单。对于某些实例类型（Windows 和 Linux），这些关联会在单个实例上安装插件以收集清单。
2. Amazon Inspector 使用 SSM 从实例中提取程序包清单。
3. Amazon Inspector 会评估提取的清单，并针对检测到的任何漏洞生成调查发现。

符合条件的实例

如果实例满足以下条件，Amazon Inspector 将使用基于代理的方法对其进行扫描：

- 该实例具有支持的操作系统。有关支持的操作系统列表，请参阅the section called “Amazon EC2 扫描支持的操作系统”的基于代理的扫描支持列。
- 未使用 Amazon Inspector EC2 排除标签将该实例排除在扫描范围之外。
- 该实例由 SSM 托管。有关验证和配置该代理的说明，请参阅配置 SSM 代理。

基于代理的扫描行为

使用基于代理的扫描方法时，在以下情况下，Amazon Inspector 会对 EC2 实例启动新的漏洞扫描：

- 启动新 EC2 实例时。
- 在现有 EC2 实例（Linux 和 Mac）上安装新软件时。
- Amazon Inspector 在其数据库中添加新的常见脆弱性和风险 (CVE) 项目，且该 CVE 与您的 EC2 实例（Linux 和 Mac）相关时。

初始扫描完成后，Amazon Inspector 会更新 EC2 实例的上次扫描时间字段。此后，当 Amazon Inspector 评估 SSM 清单时（默认为每 30 分钟一次），或者由于影响实例的新 CVE 被添加到 Amazon Inspector 数据库而需要重新扫描该实例时，上次扫描时间字段就会更新。

您可以通过账户管理页面的“实例”选项卡或通过 ListCoverage 命令查看上次扫描 EC2 实例是否存在漏洞的时间。

配置 SSM 代理

为了让 Amazon Inspector 检测到使用基于代理的扫描方法的 Amazon EC2 实例的软件漏洞，该实例必须是 Amazon EC2 Systems Manager（SSM）中的托管实例。SSM 托管实例已安装并运行了 SSM

代理，SSM 有权管理该实例。如果您已经在使用 SSM 来管理实例，那么无需执行其他步骤，即可开始基于代理的扫描。

使用某些亚马逊机器映像 (AMI) 创建的 EC2 实例默认安装了 SSM 代理。有关更多信息，请参阅 AWS Systems Manager 用户指南中的关于 SSM 代理。但是，即使已经安装了 SSM 代理，您也可能需要手动激活 SSM 代理，并授予 SSM 管理实例的权限。

以下步骤介绍了如何使用 IAM 实例配置文件将 Amazon EC2 实例配置为托管实例。这些步骤还提供了指向 AWS Systems Manager 用户指南中更多详细信息的链接。

附加实例配置文件时，建议使用 AmazonSSMManagedInstanceCore 策略。该策略拥有 Amazon Inspector EC2 扫描所需的所有权限。

> **Note**
>
> 您还可以使用 SSM 默认主机管理配置，自动通过 SSM 管理所有 EC2 实例，而无需使用 IAM 实例配置文件。有关更多信息，请参阅默认主机管理配置。

为 Amazon EC2 实例配置 SSM

1. 如果操作系统供应商未安装 SSM 代理，请先安装。有关更多信息，请参阅使用 SSM 代理。
2. AWS CLI 使用验证 SSM 代理是否正在运行。有关更多信息，请参阅检查 SSM 代理状态并启动代理。
3. 向 SSM 授予管理实例的权限。您可以通过创建 IAM 实例配置文件并将其附加到实例来授予相应权限。我们建议使用 AmazonSSMManagedInstanceCore 策略，因为该策略具有 Amazon Inspector 扫描所需的 SSM Distributor、SSM Inventory 和 SSM State Manager 权限。有关创建具有这些权限的实例配置文件并将其附加到实例的说明，请参阅为 Systems Manager 配置实例权限。
4. （可选）激活 SSM 代理的自动更新。有关更多信息，请参阅自动更新 SSM 代理。
5. （可选）将 Systems Manager 配置为使用 Amazon Virtual Private Cloud（Amazon VPC）端点。有关更多信息，请参阅创建 Amazon VPC 端点。

> **Important**
>
> Amazon Inspector 需要您的账户中具有 Systems Manager State Manager 关联才能收集软件应用程序清单。如果不存在相应关联，Amazon Inspector 会自动创建一个名为 `InspectorInventoryCollection-do-not-delete` 的关联。

Amazon Inspector 还需要资源数据同步，如果不存在相应同步，则会自动创建一个名为 `InspectorResourceDataSync-do-not-delete` 的同步。有关更多信息，请参阅 AWS Systems Manager 用户指南中的配置清单的资源数据同步。每个账户可在每个区域拥有一定数量的资源数据同步。有关更多信息，请参阅 SSM 端点和配额中资源数据同步的最大数量（ AWS 账户 每个区域）。如果您已达到此最大值，则需要删除一个资源数据同步，请参阅管理资源数据同步。

为扫描创建的 SSM 资源

Amazon Inspector 需要您的账户中有大量 SSM 资源才能运行 Amazon EC2 扫描。以下资源是在您首次激活 Amazon Inspector EC2 扫描时创建的：

> **Note**
>
> 如果在为您的账户激活 Amazon Inspector Amazon EC2 扫描时删除了这些 SSM 资源中的任何一个，Amazon Inspector 将在下一个扫描间隔尝试重新创建它们。

`InspectorInventoryCollection-do-not-delete`

这是一个 Systems Manager State Manager (SSM) 关联，Amazon Inspector 使用它从您的 Amazon EC2 实例收集软件应用程序清单。如果您的账户已经有用于从 `InstanceIds*` 中收集清单的 SSM 关联，则 Amazon Inspector 将使用该关联而不是自己创建。

`InspectorResourceDataSync-do-not-delete`

这是一个资源数据同步，Amazon Inspector 使用它将收集的清单数据从 Amazon EC2 实例发送到 Amazon Inspector 拥有的 Amazon S3 存储桶。有关更多信息，请参阅 AWS Systems Manager 用户指南中的配置清单的资源数据同步。

`InspectorDistributor-do-not-delete`

这是 Amazon Inspector 用于扫描 Windows 实例的 SSM 关联。此关联会在 Windows 实例上安装 Amazon Inspector SSM 插件。如果插件文件被无意中删除，则此关联将在下一个关联间隔重新安装它。

`InvokeInspectorSsmPlugin-do-not-delete`

这是 Amazon Inspector 用于扫描 Windows 实例的 SSM 关联。此关联使 Amazon Inspector 可以使用该插件启动扫描，您也可以使用它来设置扫描 Windows 实例的自定义间隔。有关更多信息，请参阅为 Windows 实例扫描设置自定义计划。

`InspectorLinuxDistributor-do-not-delete`

这是 Amazon Inspector 用于亚马逊 EC2 Linux 深度检查的 SSM 关联。此关联会在 Linux 实例上安装 Amazon Inspector SSM 插件。

`InvokeInspectorLinuxSsmPlugin-do-not-delete`

这是亚马逊 Inspector 用于亚马逊 EC2 Linux 深度检查的 SSM 关联。此关联使 Amazon Inspector 可以使用该插件启动扫描。

> **Note**
>
> 当你停用 Amazon Inspector Amazon EC2 扫描或深度检查时，所有 SSM 资源都将自动从相应的 Linux 主机上卸载。

无代理扫描

当您的账户处于混合扫描模式时（包括基于代理的扫描和无代理扫描），Amazon Inspector 会对符合条件的实例使用无代理扫描方法。对于无代理扫描，Amazon Inspector 使用 EBS 快照从您的实例收集软件清单。它会对使用无代理方法扫描的实例进行扫描，看其是否存在操作系统程序包和应用程序编程语言包漏洞。

> **Note**
>
> 扫描 Linux 实例是否存在应用程序编程语言包漏洞时，无代理方法会扫描所有可用路径，而基于代理的扫描仅扫描默认路径和您在Amazon EC2 Linux 实例的 Amazon Inspector 深度检查中指定的其他路径。这可能会导致同一个实例有不同的调查发现，具体取决于它是使用基于代理的方法还是使用无代理方法进行扫描。

以下过程说明了 Amazon Inspector 如何使用 EBS 快照收集清单和执行无代理扫描：

1. Amazon Inspector 会为附加到实例的所有卷创建一个 EBS 快照。当 Amazon Inspector 使用它时，快照存储在您的账户中，会使用 `InspectorScan` 标签键进行标记，并使用唯一的扫描 ID 作为标签值。
2. Amazon Inspector 使用 EBS 直接 API 从快照中检索数据，并评估它们是否存在漏洞。针对任何检测到的漏洞，都会生成调查发现。

3. Amazon Inspector 会删除它在您的账户中创建的 EBS 快照。

符合条件的实例

如果实例满足以下条件，Amazon Inspector 将使用无代理方法对其进行扫描：

- 该实例具有支持的操作系统。有关支持的操作系统列表，请参阅the section called “Amazon EC2 扫描支持的操作系统”的基于代理的扫描支持列。
- 未使用 Amazon Inspector EC2 排除标签将该实例排除在扫描范围之外。
- 该实例的状态为`Unmanaged EC2 instance`、`Stale inventory`或`No inventory`。
- 该实例由 EBS 支持，具有以下文件系统格式之一：
  - `ext3`
  - `ext4`
  - `xfs`

无代理扫描行为

当您的账户配置为混合扫描时，Amazon Inspector 会每 24 小时对符合条件的实例执行一次无代理扫描。Amazon Inspector 每小时都会检测和扫描新的符合条件的实例，其中包括没有 SSM 代理的新实例，或者状态已更改为 `SSM_UNMANAGED` 的现有实例。

每当 Amazon Inspector 在无代理扫描后扫描从实例提取的快照时，都会更新 Amazon EC2 实例的上次扫描时间字段。

您可以通过账户管理页面的“实例”选项卡或通过 ListCoverage 命令查看上次扫描 EC2 实例是否存在漏洞的时间。

管理扫描模式

EC2 扫描模式决定了 Amazon Inspector 在您的账户中执行 EC2 扫描时将使用哪些扫描方法。您可以在常规设置下的 EC2 扫描设置页面查看账户的扫描模式。独立账户或 Amazon Inspector 委派管理员可以更改扫描模式。当您将扫描模式设置为 Amazon Inspector 委派管理员时，系统会为您组织中的所有成员账户设置该扫描模式。Amazon Inspector 具有以下扫描模式：

基于代理的扫描 – 在此扫描模式下，Amazon Inspector 在扫描程序包漏洞时将仅使用基于代理的扫描方法。此扫描模式仅扫描您账户中的 SSM 托管实例，但其好处是可以提供持续扫描，以响应新的

CVE 或对实例的更改。基于代理的扫描还可以为符合条件的实例提供 Amazon Inspector 深度检查。这是新激活账户的默认扫描模式。

混合扫描 – 在此扫描模式下，Amazon Inspector 将结合使用基于代理和无代理的方法来扫描程序包漏洞。对于安装并配置了 SSM 代理的符合条件的 EC2 实例，Amazon Inspector 会使用基于代理的方法。对于不受 SSM 管理的符合条件的实例，Amazon Inspector 将对符合条件且由 EBS 支持的实例使用无代理方法。

更改扫描模式

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的选择 AWS 区域 器，选择要更改 EC2 扫描模式的区域。
3. 在侧面导航面板的常规设置下，选择 EC2 扫描设置。
4. 在扫描模式下，选择编辑。
5. 选择一个扫描模式，然后选择保存更改。

从 Amazon Inspector 扫描中排除实例

您可以标记某些实例，将其排除在 Amazon Inspector 扫描之外。从扫描中排除实例有助于防止出现无法操作的警报。您无需为排除的实例付费。

要将 EC2 实例排除在扫描范围之外，请使用以下键标记实例：

- `InspectorEc2Exclusion`

值是可选的。

有关添加标签的更多信息，请参阅标记 Amazon EC2 资源。

此外，您可以通过使用标签标记用于加密该卷的 AWS KMS 密钥，将加密的 EBS 卷排除在无代理扫描之外。`InspectorEc2Exclusion`有关更多信息，请参阅标记密钥。

支持的操作系统

Amazon Inspector 会扫描支持的 Mac、Windows 和 Linux EC2 实例，查找操作系统程序包中的脆弱性。对于 Linux 实例，Amazon Inspector 可以使用 Amazon EC2 Linux 实例的 Amazon Inspector 深度检查 生成应用程序编程语言包的调查发现。对于 Mac 和 Windows 实例，仅扫描操作系统程序包。

有关支持的操作系统的信息，包括无需 SSM 代理即可扫描哪些操作系统，请参阅Amazon EC2 扫描支持的操作系统。

Amazon EC2 Linux 实例的 Amazon Inspector 深度检查

Amazon Inspector 扩大了其亚马逊 EC2 扫描覆盖范围，将深度检查包括在内。通过深度检查，Amazon Inspector 可以检测基于 Linux 的 Amazon EC2 实例中应用程序编程语言包的软件包漏洞。

Amazon Inspector 会扫描编程语言包库的默认路径。除了默认路径外，您还可以配置自定义路径。有关更多信息，请参阅Amazon Inspector 深度检查的自定义路径。

Amazon Inspector 使用通过 Amazon Inspector SSM 插件收集的数据进行深度检查扫描。为了管理插件并对 Linux 进行深入检查，Amazon Inspector 会自动`InvokeInspectorLinuxSsmPlugin-do-not-delete`在您的账户中创建以下 SSM 关联。当 Amazon Inspector 激活深度检查时，就会发生这种情况。

Amazon Inspector 每 6 小时从实例收集更新的应用程序清单，以便进行深度检查。

有关 Amazon Inspector 支持深度检查的编程语言列表，请参阅支持的编程语言：Amazon EC2 深度检查。

> **Note**
>
> Windows 或 Mac 实例不支持深度检查。

激活或停用深度检查

> **Note**
>
> 对于在 2023 年 4 月 17 日之后激活 Amazon Inspector 的账户，深度检查将作为 Amazon EC2 扫描的一部分自动激活。

您可以在 Amazon Inspector 控制台的账户管理页面的 Amazon EC2 扫描列中查看账户是否激活了深度检查。如果深度检查未激活，则此列将显示已激活（深度检查已停用）。要以编程方式查看激活状态，请使用 GetEc2DeepInspectionConfiguration API。或者，对于多个账户，请使用 BatchGetMemberEc2DeepInspectionStatus API。

如果您在 2023 年 4 月 17 日之前激活了 Amazon Inspector，则可以通过控制台横幅或 UpdateEc2DeepInspectionConfiguration API 激活深度检查。如果您是 Amazon Inspector 中某组织的委派管理员，则可以使用 BatchUpdateMemberEc2DeepInspectionStatus API 为自己和成员账户激活深度检查。

您可以通过 UpdateEc2DeepInspectionConfiguration API 停用深度检查。组织中的成员账户无法停用深度检查。因此，成员账户必须由其委派管理员使用 BatchUpdateMemberEc2DeepInspectionStatus API 停用。

关于适用于 Linux 的 Amazon Inspector SSM 插件

Amazon Inspector 使用 Amazon Inspector SSM 插件对 Linux 实例进行深度检查。Amazon Inspector SSM 插件会自动安装在 Linux 实例的以下目录中：`/opt/aws/inspector/bin`。可执行文件的名称是 `inspectorssmplugin`。

> **Note**
>
> Amazon Inspector 使用 Systems Manager Distributor 在 Amazon EC2 实例中部署该插件。Systems Manager Distributor 支持 Systems Manager 指南内的支持的程序包平台和架构中列出的操作系统。您的 Amazon EC2 实例的操作系统是 Systems Manager Distributor 和 Amazon Inspector 支持的系统，Amazon Inspector 才能执行深度检查扫描。

Amazon Inspector 会创建以下文件目录来管理 Amazon Inspector SSM 插件收集的用于深度检查的数据：

- `/opt/aws/inspector/var/input`
- `/opt/aws/inspector/var/output`
  - 此目录中的 `packages.txt` 存储了深度检查发现的程序包的完整路径。如果 Amazon Inspector 在实例上多次检测到同一个程序包，则此文件会列出该程序包的每个位置。

Amazon Inspector 将该插件的日志存储在 `/var/log/amazon/inspector` 目录中。

卸载 Amazon Inspector SSM 插件

如果 `inspectorssmplugin` 文件被无意中删除，则 `InspectorLinuxDistributor-do-not-delete` SSM 关联将在下一个扫描间隔尝试重新安装它。

如果您停用 Amazon EC2 扫描，则该插件将自动从所有 Linux 主机上卸载。

Amazon Inspector 深度检查的自定义路径

您可以配置自定义路径，让 Amazon Inspector 在对您的 Linux Amazon EC2 实例进行深入检查时进行搜索。当您添加自定义路径时，Amazon Inspector 会扫描该目录及其中的所有子目录中的程序包。

所有账户都可以为其个人账户定义最多 5 个自定义路径。如果您是组织的委托管理员，则可以额外定义 5 条路径，这些路径将适用于整个组织。这样一来，组织中每个账户最多可扫描 10 个自定义路径。

Amazon Inspector 除了扫描所有账户的以下默认路径外，还会扫描所有自定义路径：

- `/usr/lib`
- `/usr/lib64`
- `/usr/local/lib`
- `/usr/local/lib64`

> **Note**
>
> 自定义路径必须是本地路径。Amazon Inspector 不会扫描映射的网络路径，例如网络文件系统 (NFS) 挂载或 Amazon S3 文件系统挂载。

自定义路径的格式

以下为自定义路径的格式示例：`/home/usr1/project01`

自定义路径不能超过 256 个字符。

每个实例的程序包上限为 5,000 个，程序包清单收集时间上限为 15 分钟。我们建议您尽量选择自定义路径，以规避这些限制。

在控制台中设置自定义路径

Console

以 Amazon Inspector 委托管理员的身份登录，然后按照以下步骤为组织添加自定义路径。

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的 AWS 区域 选择器，选择要激活 Lambda 标准扫描的区域。

3. 在侧面导航面板的常规设置下，选择 EC2 扫描设置。
4. 在自己账户的自定义路径下，选择编辑，为自己的个人账户添加路径。如果您是委托管理员，则可以在组织的自定义路径窗格中选择编辑，为组织内的所有账户添加自定义路径。
5. 在文本框中输入自定义路径。
6. 选择保存以保存自定义路径。Amazon Inspector将在下一次深度检查中包括这些路径。

API

运行 UpdateEc2DeepInspectionConfiguration 命令。对于 `packagePaths`，指定要扫描的路径数组。

支持的编程语言

对于 Linux 实例，除了操作系统包中的漏洞外，Amazon Inspector 深度检查还可以生成应用程序编程语言包的调查发现。对于 Mac 和 Windows 实例，仅扫描操作系统程序包。

有关支持的编程语言的信息，请参阅Amazon Inspector 深度检查支持的编程语言。

使用 Amazon Inspector 扫描 Windows EC2 实例

> **Note**
>
> 2022 年 8 月 31 日，Amazon Inspector 将其 Amazon EC2 扫描覆盖范围扩展至运行 Windows 的 EC2 实例。

Amazon Inspector 会自动发现所有支持的 Windows 实例，并将其纳入连续扫描，无需任何额外操作。有关支持的实例的信息，请参阅Amazon EC2 扫描支持的操作系统。

与扫描基于 Linux 的实例不同，Amazon Inspector 会定期运行 Windows 扫描。Windows 实例最初在发现时进行扫描，然后每 6 小时扫描一次。但是，默认的 6 小时扫描间隔是可调整的。有关更多信息，请参阅为 Windows 实例扫描设置自定义计划。下面概述了 Amazon Inspector 如何扫描 Windows 实例：

1. 激活 Amazon EC2 扫描后，Amazon Inspector 会为 Windows 资源创建新的 SSM 关联：`InspectorDistributor-do-not-delete`、`InspectorInventoryCollection-do-not-delete` 和 `InvokeInspectorSsmPlugin-do-not-delete`。

2. `InspectorDistributor-do-not-deleteSSM` 关联使用 SSM 文档和 `AWS-ConfigureAWSPackageAmazonInspector2-InspectorSsmPlugin` SSM 分销商包在您的实例上安装 Amazon Inspector SSM 插件。Windows请参阅关于适用于 Amazon Inspector SSM 插件 Windows了解更多信息。
3. `InvokeInspectorSsmPlugin-do-not-deleteSSM` 协会定期运行 Amazon Inspector SSM 插件，以收集实例数据并生成 Amazon Inspector 的调查结果。默认情况下，间隔为每 6 小时一次。但是，您可以使用 SSM 为关联设置 cron 表达式或 rate 表达式，自定义这一间隔。有关更多信息，请参阅 AWS Systems Manager 用户指南中的参考：适用于 Systems Manager 的 cron 和 rate 表达式。

> **Note**
>
> Amazon Inspector 将更新的开放脆弱性和评测语言 (OVAL) 定义文件暂存到 S3 存储桶 `inspector2-oval-prod-REGION`。此 S3 存储桶包含扫描中使用的 OVAL 定义，不应进行修改。更改此设置将使 Amazon Inspector 无法扫描新发布的 CVE。

Windows 实例的 Amazon Inspector 扫描要求

要扫描 Windows 实例，Amazon Inspector 要求实例满足以下条件：

- 该实例是 SSM 托管实例。有关设置扫描实例的说明，请参阅配置 SSM 代理。
- 实例操作系统是支持的 Windows 操作系统之一。有关支持的操作系统类型的完整列表，请参阅Amazon EC2 扫描支持的操作系统。
- 该实例安装了 Amazon Inspector SSM 插件。发现后，Amazon Inspector 会自动为托管实例安装 Amazon Inspector SSM 插件。有关该插件的详细信息，请参阅下一个主题。

> **Note**
>
> 如果主机在 Amazon VPC 中运行，但没有出站互联网访问权限，则 Windows 扫描要求主机能够访问区域 Amazon S3 端点。要了解如何配置 Amazon S3 Amazon VPC 端点，请参阅 Amazon Virtual Private Cloud 用户指南中的创建网关端点。如果您的 Amazon VPC 终端节点策略限制对外部 S3 存储桶的访问，则必须明确允许访问由 Amazon Inspector 维护的存储桶 AWS 区域 ，该存储桶存储用于评估您的实例的 OVAL 定义。此存储桶使用以下格式：`inspector2-oval-prod-REGION`。

关于适用于 Amazon Inspector SSM 插件 Windows

Amazon Inspector 需要使用 Amazon Inspector SSM 插件才能扫描您的Windows实例。Amazon Inspector SSM 插件会自动安装在您的Windows实例上`C:\Program Files\Amazon\Inspector`，并命名为`InspectorSsmPlugin.exe`可执行的二进制文件。

创建以下文件位置是为了存储 Amazon Inspector SSM 插件收集的数据：

- `C:\ProgramData\Amazon\Inspector\Input`
- `C:\ProgramData\Amazon\Inspector\Output`
- `C:\ProgramData\Amazon\Inspector\Logs`

> **Note**
>
> 默认情况下，Amazon Inspector SSM 插件的运行优先级低于正常水平。

卸载 Amazon Inspector SSM 插件

如果 `InspectorSsmPlugin.exe` 文件被无意中删除，则 `InspectorDistributor-do-not-delete` SSM 关联将在下一个 Windows 扫描间隔重新安装插件。如果你想卸载 Amazon Inspector SSM 插件，你可以使用`AmazonInspector2-ConfigureInspectorSsmPlugin`文档上的 “卸载” 操作。

此外，如果您停用亚马逊 EC2 扫描，Amazon Inspector SSM 插件将自动从所有Windows主机上卸载。

> **Note**
>
> 如果你在停用 Amazon Inspector 之前卸载 SSM 代理，Amazon Inspector SSM 插件将保留在Windows主机上，但将不再向 Amazon Inspector SSM 插件发送数据。有关更多信息，请参阅停用 Amazon Inspector。

为 Windows 实例扫描设置自定义计划

您可以使用 SSM 为 `InvokeInspectorSsmPlugin-do-not-delete` 关联设置 cron 表达式或 rate 表达式，从而自定义 Windows Amazon EC2 实例扫描之间的间隔时间。有关更多信息，请参阅 AWS

Systems Manager 用户指南中的参考：适用于 Systems Manager 的 cron 和 rate 表达式，或使用以下说明。

从以下代码示例中选择一个，使用 rate 表达式或 cron 表达式将 Windows 实例的扫描节奏从默认的 6 小时更改为 12 小时。

以下示例要求您使用名AssociationId为的关联`InvokeInspectorSsmPlugin-do-not-delete`。你可以AssociationId通过运行以下 AWS CLI 命令来检索你的：

```
$ aws ssm list-associations --association-filter-list
 "key=AssociationName,value=InvokeInspectorSsmPlugin-do-not-delete" --region us-east-1
```

> **Note**
>
> AssociationId是区域性的，因此您需要先为每个区域检索一个唯一的 ID AWS 区域。然后，您可以运行上述命令，在要为 Windows 实例设置自定义扫描计划的每个区域更改扫描节奏。

Example rate expression

```
$ aws ssm update-association \
--association-id "YourAssociationId" \
--association-name "InvokeInspectorSsmPlugin-do-not-delete" \
--schedule-expression "rate(12 hours)"
```

Example cron expression

```
$ aws ssm update-association \
--association-id "YourAssociationId" \
--association-name "InvokeInspectorSsmPlugin-do-not-delete" \
--schedule-expression "cron(0 0/12 * * ? *)"
```

使用 Amazon Inspector 扫描 Amazon ECR 容器映像

Amazon Inspector 会扫描存储在 Amazon ECR 中的容器映像以查找软件脆弱性，生成程序包脆弱性调查发现。有关针对这些问题得出的调查发现类型的信息，请参阅Amazon Inspector 中的调查发现类型。

为 Amazon ECR 激活 Amazon Inspector 扫描后，会将 Amazon Inspector 设置为私有注册表的首选扫描服务。这会将默认的基本扫描（由 Amazon ECR 免费提供）替换为增强扫描（由 Amazon Inspector 提供和计费）。

Amazon Inspector 提供的增强扫描可在注册表级别对操作系统和编程语言包进行脆弱性扫描。您可以在 Amazon ECR 控制台中，查看针对映像每一层，使用增强扫描在映像级别发现的调查发现。此外，您还可以在其他不适用于基本扫描结果的服务（包括和 Amazon）中查看 AWS Security Hub 和处理这些发现 EventBridge。你可以在 Amazon Inspector 控制台上查看通过扫描发现的结果，网址为 https://console.aws.amazon.com/inspector/v2/home。有关如何处理调查发现的信息，请参阅管理 Amazon Inspector 中的调查发现。

有关激活 Amazon ECR 扫描的说明，请参阅激活扫描类型。

Amazon ECR 扫描的扫描行为

当您首次激活 ECR 扫描并且您的存储库配置为持续扫描时，Amazon Inspector 会检测到您在 30 天内推送或在过去 90 天内提取的所有符合条件的图像。然后，Amazon Inspector 会扫描检测到的图像并将其扫描状态设置为`active`。只要图像是在过去 90 天内（默认）或在您配置的 ECR 重新扫描时间内推送或拉取的，Amazon Inspector 就会继续监控这些图像。有关更多信息，请参阅配置 ECR 重新扫描持续时间。

为了持续扫描，Amazon Inspector 会在以下情况下启动对容器映像的新漏洞扫描：

- 每当推送新的容器映像时。
- 每当 Amazon Inspector 在其数据库中添加新的常见脆弱性和风险 (CVE) 项目，并且 CVE 与该容器映像相关时（仅限持续扫描）。

如果您将存储库配置为推送扫描，则只有在推送图像时才会对其进行扫描。

您可以通过账户管理页面的容器映像选项卡或使用 ListCoverage API 查看上次检查容器映像是否存在漏洞的时间。发生以下事件时，Amazon Inspector 会更新 Amazon ECR 映像的上次扫描时间字段：

- Amazon Inspector 完成对容器映像的初始扫描时。
- 由于 Amazon Inspector 数据库中添加了影响该容器映像的新的常见脆弱性和风险 (CVE) 项目，因此 Amazon Inspector 重新扫描容器映像时。

支持的操作系统和媒体类型

有关支持的操作系统的信息，请参阅Amazon ECR 扫描支持的操作系统。

Amazon Inspector 对 Amazon ECR 存储库的扫描涵盖以下支持的媒体类型：

- "application/vnd.docker.distribution.manifest.v1+json"
- "application/vnd.docker.distribution.manifest.v1+prettyjws"
- "application/vnd.oci.image.manifest.v1+json"
- "application/vnd.docker.distribution.manifest.v2+json"

> **Note**
>
> 不支持暂存映像和 DockerV2ListMediaType 映像。

为 Amazon ECR 存储库配置增强扫描

激活 Amazon Inspector 对 Amazon ECR 容器映像的扫描时，会更改私有注册表的扫描配置设置。注册表的扫描类型会从基本扫描更改为 Amazon Inspector 提供的增强扫描。有关更多信息，请参阅 Amazon ECR 用户指南中的映像扫描。

您可以在 ECR 中管理存储库级别的增强扫描设置。您可以为存储库选择持续扫描或推送时扫描。持续扫描包括推送时扫描和自动重新扫描。推送时扫描仅在最初推送映像时扫描。对于这两个选项，都可以通过包含筛选条件来调整扫描范围。默认情况下，首次激活增强扫描功能时，设置将更改为持续扫描所有存储库。

配置增强扫描设置

1. 打开 Amazon ECR 控制台：https://console.aws.amazon.com/ecr/。
2. 在页面右上角的 AWS 区域 选择器中，选择包含您正在扫描的存储库的区域。
3. 在导航窗格中，选择私有注册表，然后选择扫描。
4. 在扫描类型下，确保增强扫描已选中。如果尚未选中，请选择增强扫描。

   默认情况下，持续扫描所有存储库选项处于选中状态，这将为所有存储库开启完整的 Amazon Inspector 扫描覆盖范围。
5. 取消选择持续扫描所有存储库，可筛选哪些存储库需要持续扫描或在推送时扫描。

有关配置增强扫描的更多信息，请参阅 Amazon ECR 用户指南中的使用增强扫描。

配置 ECR 重新扫描持续时间

ECR 重新扫描持续时间设置决定了 Amazon Inspector 持续监控存储库中容器映像的时间长度。您可以为图像推送日期和图像提取日期配置重新扫描持续时间。新帐户（包括添加到组织中的新帐户）的默认扫描持续时间为 90 天。

图片推送日期持续时间

图片推送日期的持续时间决定了 Amazon Inspector 在最新拉取日期之后将图像推送到存储库后持续监控多长时间。以下选项可用作重新扫描持续时间：

- 14 天
- 30 天
- 60 天
- 90 天（默认）
- 180 天
- 生命周期

图片提取日期持续时间

图片提取日期持续时间决定了在最近一次提取日期之后 Amazon Inspector 持续监控图像的时间。以下选项可用作重新扫描持续时间：

- 14 天
- 30 天
- 60 天
- 90 天（默认）
- 180 天

只要在配置的推送和拉取日期内推送或拉出图像，Amazon Inspector 就会继续监控和重新扫描图像。如果未在配置的推送和拉取日期内推送或拉取映像，Amazon Inspector 将停止对其进行监控。

> **Note**
>
> 当 Amazon Inspector 停止监控图像时，它会将图像扫描状态代码设置为`inactive`，原因代码设置为`expired`。然后，它会安排关闭所有相关的图像搜索结果。

将重新扫描持续时间设置为最适合您的环境。例如，如果您经常构建图像，请选择较短的扫描持续时间。同样，如果您长时间使用图像，请选择更长的扫描持续时间。

当您配置委托管理员账户的重新扫描持续时间时，Amazon Inspector 会将该设置应用于组织中的所有成员账户。

配置 ECR 重新扫描持续时间

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 从导航窗格中选择 “常规设置”，然后选择 “ECR 扫描设置”。
3. 在 ECR 扫描设置中，在 ECR 重新扫描持续时间下，选择要设置的图像推送日期持续时间和图像提取日期持续时间。
4. 选择保存。您的新设置将立即生效。

> **Note**
>
> 如果您延长推送日期持续时间，Amazon Inspector 会将更改应用于配置为持续扫描的存储库中所有主动扫描的图像。但是，即使您在新的持续时间内推送了非活动图像，它们仍处于非活动状态。

使用 Amazon Inspector 进行扫描 AWS Lambda

Amazon Inspector 对 AWS Lambda 函数的支持为 Lambda 函数和层提供了持续、自动的安全漏洞评估。Amazon Inspector 提供两种类型的 Lambda 扫描。这些扫描类型可查找不同类型的脆弱性。

Amazon Inspector Lambda 标准扫描

这是默认的 Lambda 扫描类型。Lambda 标准扫描会扫描 Lambda 函数及其层中的应用程序依赖项，以查找程序包脆弱性。有关更多信息，请参阅Lambda 标准扫描。

Amazon Inspector Lambda 代码扫描

这种扫描类型会扫描函数及其层中的自定义应用程序代码，以查找代码脆弱性。您可以单独激活 Lambda 标准扫描，也可以同时激活 Lambda 标准扫描和 Lambda 代码扫描。有关更多信息，请参阅Amazon Inspector Lambda 代码扫描。

激活 Lambda 扫描后，Amazon Inspector 会在您的账户中创建以下 AWS CloudTrail 与服务相关的渠道：

- `cloudtrail:CreateServiceLinkedChannel`
- `cloudtrail:DeleteServiceLinkedChannel`

Amazon Inspector 管理这些渠道，并使用它们来监控您的扫描 CloudTrail 事件。有关服务关联频道的更多信息，请参阅使用 CLI AWS 查看服务相关频道。 CloudTrail

> **Note**
>
> Amazon Inspector 创建的服务相关渠道允许您像有 CloudTrail 跟踪一样查看账户中的 CloudTrail 事件，但是，我们建议您创建自己的渠道 CloudTrail 来管理账户的事件。

有关激活 Lambda 函数的说明，请参阅激活扫描类型。

Lambda 函数扫描的扫描行为

激活后，Amazon Inspector 会扫描您账户中过去 90 天内调用或更新的所有 Lambda 函数。在以下情况下，Amazon Inspector 会对 Lambda 函数启动脆弱性扫描：

- Amazon Inspector 发现现有的 Lambda 函数时。
- 将新的 Lambda 函数部署到 Lambda 服务时。
- 部署现有 Lambda 函数或其层的应用程序代码或依赖项更新时。
- Amazon Inspector 在其数据库中添加新的常见脆弱性和风险 (CVE) 项目，且该 CVE 与您的函数相关时。

Amazon Inspector 会在每个 Lambda 函数的整个生命周期内对其进行监控，直到该函数被删除或被排除在扫描范围之外。

您可以通过账户管理页面的 Lambda 函数选项卡或使用 ListCoverage API 查看上次检查 Lambda 函数是否存在漏洞的时间。发生以下事件时，Amazon Inspector 会更新 Lambda 函数的上次扫描时间字段：

- Amazon Inspector 完成对 Lambda 函数的初始扫描时。
- 更新 Lambda 函数时。
- 由于影响 Lambda 函数的新 CVE 项目添加到 Amazon Inspector 数据库中，Amazon Inspector 重新扫描该函数时。

支持的运行时系统和符合条件的函数

对于 Lambda 标准扫描和 Lambda 代码扫描，Amazon Inspector 支持不同的运行时系统。有关每种扫描类型支持的运行时系统的列表，请参阅支持的运行时系统：Amazon Inspector Lambda 标准扫描和支持的运行时系统：Amazon Inspector Lambda 代码扫描。

除了具有受支持的运行时系统之外，Lambda 函数还需要满足以下条件才有资格进行 Amazon Inspector 扫描：

- 过去 90 天内调用或更新过该函数。
- 该函数被标记为 `$LATEST`。
- 该函数未按标签从扫描中排除。

> **Note**
>
> 过去 90 天内未调用或修改的 Lambda 函数将自动排除在扫描范围之外。如果 Lambda 函数再次被调用或对函数代码进行了更改，则 Amazon Inspector 将恢复对自动排除的函数的扫描。

Amazon Inspector Lambda 标准扫描

Amazon Inspector Lambda 标准扫描可识别您添加到 Lambda 函数代码和层的应用程序包依赖项中的软件漏洞。例如，如果 Lambda 函数使用的 `python-jwt` 程序包版本存在已知脆弱性，则 Lambda 标准扫描将生成针对该函数的调查发现。

如果 Amazon Inspector 在 Lambda 函数应用程序包依赖项中检测到脆弱性，则 Amazon Inspector 会生成详细的程序包脆弱性类型的调查发现。

有关激活扫描类型的说明，请参阅激活扫描类型。

> **Note**
>
> Lambda 标准扫描不会扫描 Lambda 运行时环境中默认安装的 AWS SDK 依赖项。Amazon Inspector 仅扫描使用函数代码上传的依赖项或从层继承的依赖项。

> **Note**
>
> 停用 Amazon Inspector Lambda 标准扫描也将同时停用 Amazon Inspector Lambda 代码扫描。

从 Lambda 标准扫描中排除函数

您可以标记某些函数，将其排除在 Amazon Inspector Lambda 标准扫描之外。从扫描中排除函数有助于防止出现无法操作的警报。

要从 Lambda 标准扫描中排除 Lambda 函数，请使用以下键值对标记函数：

- 键：`InspectorExclusion`
- 值：`LambdaStandardScanning`

从 Lambda 标准扫描中排除函数

1. 在 https://console.aws.amazon.com/lambda/ 上打开 Lambda 控制台。
2. 选择函数。
3. 在函数表中，选择要从 Amazon Inspector Lambda 标准扫描中排除的函数的名称。
4. 选择配置，然后从菜单中选择标签。
5. 选择管理标签，然后选择添加新标签。
6. 在键字段中输入 `InspectorExclusion`，然后在值字段中输入 `LambdaStandardScanning`。
7. 选择保存以添加标签，并将函数排除在 Amazon Inspector Lambda 标准扫描之外。

有关在 Lambda 中添加标签的更多信息，请参阅在 Lambda 函数上使用标签。

Amazon Inspector Lambda 代码扫描

> **Important**
>
> 代码扫描从 Lambda 函数中捕获代码片段，以突出显示检测到的脆弱性。这些片段可能以纯文本形式显示硬编码的凭证或其他敏感材料。

Amazon Inspector Lambda 代码扫描会根据安全最佳实践扫描 Lambda 函数中的自定义应用程序代码，以查找代码漏洞。 AWS Lambda 代码扫描可检测注入缺陷、数据泄露、弱加密或代码中缺少加密。有关可用区域的信息，请参阅特定于区域的功能可用性。

Lambda 标准扫描这项功能可评估函数中使用的应用程序包依赖项，以查找常见脆弱性和风险 (CVE)。您可以同时激活 Lambda 代码扫描和 Lambda 标准扫描。

Amazon Inspector 会使用自动推理和机器学习来评估 Lambda 函数应用程序代码，分析应用程序代码的总体安全合规性。它基于与 Amazon 合作开发的内部检测器来识别违反政策的行为和漏洞 CodeGuru。有关可能检测的列表，请参阅探CodeGuru 测器库。

如果 Amazon Inspector 在 Lambda 函数应用程序代码中检测到脆弱性，Amazon Inspector 会生成详细的代码脆弱性类型的调查发现。此调查发现类型包括问题在代码中的确切位置、展示问题的代码片段以及建议的补救措施。建议的补救措施包括 plug-and-play 可用于替换易受攻击的代码行的代码块。除了针对相应调查发现的一般代码补救指南外，还提供了这些建议的代码修复。

> **Important**
>
> 代码补救建议由自动推理和生成式人工智能服务提供支持，因此可能无法按预期工作。您应对自己采纳的代码补救建议负责。在采纳代码补救建议之前，请务必仔细审视这些建议。您可能需要对代码补救建议进行编辑，以确保代码符合您的预期。请参阅负责任的 AI 策略。

在代码脆弱性调查发现中加密代码

该服务存储在使用 Lambda 代码扫描发现代码漏洞时检测到的代码片段。 CodeGuru 默认情况下，由 CodeGuru 控制的AWS 自有密钥用于加密您的代码，但是，您可以通过 Amazon Inspector API 使用自己的客户托管密钥进行加密。有关更多信息，请参阅对调查发现中的代码进行静态加密

可以同时激活 Lambda 代码扫描和 Lambda 标准扫描。有关激活扫描类型的说明，请参阅激活扫描类型。

从 Lambda 代码扫描中排除函数

您可以标记某些函数，将其排除在 Amazon Inspector Lambda 代码扫描之外。从扫描中排除函数有助于防止出现无法操作的警报。

要从 Amazon Inspector 中排除 Lambda 函数，Lambda 代码扫描可使用以下键值对标记函数：

- 键：`InspectorCodeExclusion`

- 值：`LambdaCodeScanning`

从 Lambda 代码扫描中排除函数

1. 通过以下网址登录 Lambda 控制台：https://console.aws.amazon.com/lambda/。
2. 选择函数。
3. 在函数表中，选择要从 Amazon Inspector Lambda 代码扫描中排除的函数的名称。
4. 选择配置，然后从菜单中选择标签。
5. 选择管理标签，然后选择添加新标签。
6. 在键字段中输入 `InspectorCodeExclusion`，然后在值字段中输入 `LambdaCodeScanning`。
7. 选择保存以添加标签，并将函数排除在 Amazon Inspector Lambda 代码扫描之外。

有关在 Lambda 中添加标签的更多信息，请参阅在 Lambda 函数上使用标签。

停用扫描类型

您可以随时停用新的 Amazon Inspector 扫描类型。停用某扫描类型后，您将无法访问该扫描类型生成的现有调查发现。如果重新激活扫描类型，系统会扫描符合条件的资源，Amazon Inspector 将生成新的调查发现。要记录调查发现数据，可以在停用之前导出调查发现。有关更多信息，请参阅从 Amazon Inspector 导出调查发现报告。

停用扫描类型后，该 AWS 帐户可能会发生某些更改，具体取决于要停用的扫描类型。以下是停用这些扫描类型后将发生的变化：

- Amazon EC2 扫描 — 为账户停用 Amazon Inspector Amazon EC2 扫描时，Amazon Inspector 使用的以下 SSM 关联将被删除：
  - `InspectorDistributor-do-not-delete`
  - `InspectorInventoryCollection-do-not-delete`
  - `InspectorLinuxDistributor-do-not-delete`
  - `InvokeInspectorLinuxSsmPlugin-do-not-delete`
  - `InvokeInspectorSsmPlugin-do-not-delete`。 此外，通过此关联安装的 Amazon Inspector SSM 插件已从您的所有Windows主机上移除。有关更多信息，请参阅扫描 Windows 实例。
- Amazon ECR 扫描 — 为账户停用 Amazon ECR 容器映像扫描后，该账户的 Amazon ECR 扫描类型将从使用 Amazon Inspector 进行增强扫描更改为使用 Amazon ECR 进行基本扫描。

- Lambda 标准扫描 — 为账户停用 Lambda 标准扫描时，如果代码扫描也处于活动状态，则将同时停用 Lambda 代码扫描。此外，启用扫描时创建的 CloudTrail 服务关联频道将被删除。

停用扫描

停用某个账户的所有扫描类型会在 AWS 区域停用该账户的 Amazon Inspector。有关更多信息，请参阅停用 Amazon Inspector。

要在多账户环境中完成此步骤，请在以 Amazon Inspector 委托管理员身份登录后完成以下步骤。

Console

停用扫描

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的 AWS 区域 选择器，选择要停用扫描的区域。
3. 在导航窗格中，选择账户管理。
4. 选择账户选项卡以显示账户的扫描状态。
5. 选中要停用扫描的每个账户对应的复选框。
6. 选择操作，然后从停用选项中选择要停用的扫描类型。
7. （推荐）在要停 AWS 区域 用该扫描类型的每个扫描类型中重复这些步骤。

API

运行禁用 API 操作。在请求中，提供您要停用扫描的账户 ID，对于 `resourceTypes`，提供一个或多个 `EC2`、`ECR`、`LAMBDA` 或 `LAMBDA_CODE`，以停用扫描。

互联网安全中心 (CIS) 扫描 EC2 实例

当你为账户启用 Amazon Inspector EC2 扫描时，你就允许亚马逊 Inspector 执行或安排 CIS 扫描。Amazon Inspector CIS 会对您的 Amazon EC2 实例的操作系统进行基准测试，以查看其配置是否符合互联网安全中心制定的最佳实践建议。CIS 安全基准测试计划为安全配置系统提供了行业标准配置基准和最佳实践。有关更多信息，请参阅什么是 CIS 基准？

Amazon Inspector 根据您在扫描配置中定义的实例标签和扫描计划，对目标 Amazon EC2 实例执行 CIS 扫描。对于每个目标实例，Amazon Inspector 都会对该实例执行一系列检查。每项检查都会评估您的系统配置是否符合特定的 CIS 基准建议。每张支票都有CIS支票ID和标题，这与该平台的CIS基准测试建议直接相关。扫描完成后，您可以查看结果，并查看您的实例对该系统的哪些检查通过、失败或跳过。

Amazon Inspector CIS 扫描的 EC2 实例要求

要对您的实例运行 CIS 扫描，Amazon Inspector 要求该实例满足以下标准：

- 实例操作系统是 CIS 扫描支持的操作系统之一。有关支持的操作系统类型的完整列表，请参阅支持的操作系统：CIS 扫描。
- 该实例是 Amazon EC2 Systems Manager (SSM) 托管实例。有关更多信息，请参阅使用 SSM 代理。
- 该实例安装了 Amazon Inspector SSM 插件。Amazon Inspector 会自动为 SSM 托管实例安装此插件。
- 该实例的实例配置文件授予 SSM 管理该实例的权限，以及 Amazon Inspector 对该实例运行 CIS 扫描的权限。要授予这些权限，请将 AmazonInspector2 FullAccess、 AmazonSSM ManagedInstanceCore 和 AmazonInspector2 个ManagedCispolicy策略附加到 IAM 角色，并将该角色作为实例配置文件附加到您的实例。有关创建和附加实例配置文件的说明，请参阅 A mazon EC2 用户指南中的使用 IAM 角色。

> **Note**
>
> 在实例上运行 CIS 扫描时，不再需要启用 Amazon Inspector 深度检查。如果您禁用深度检查，Amazon Inspector 仍会继续安装 SSM 代理，但不会再调用该插件来运行深度检查。这意味着您的帐户中将存在以下关联:`InspectorLinuxDistributor-do-not-delete`.

正在运行 CIS 扫描

您可以按需运行一次 CIS 扫描，也可以按计划定期扫描。要运行扫描，请先创建扫描配置。

创建扫描配置时，您可以指定用于定位实例的标签键值对。如果您是某个组织的 Amazon Inspector 委托管理员，则可以在扫描配置中指定多个账户，然后 Amazon Inspector 将在每个账户中查找带有指定标签的实例。您可以为扫描选择 CIS 基准级别。对于每个基准测试，CIS 都支持 1 级和 2 级配置文件，该配置文件旨在为不同环境可能需要的不同安全级别提供基准。

- 级别 1-推荐可在任何系统上配置的基本基本安全设置。实施这些设置应该会导致很少或根本不会中断服务。这些建议的目标是减少系统入口点的数量，从而降低整体网络安全风险。
- 第 2 级-为高安全性环境推荐更高级的安全设置。实施这些设置需要进行规划和协调，以最大限度地降低业务影响的风险。这些建议的目标是帮助您实现监管合规。

等级 2 延长 1 级。当您选择 2 级时，Amazon Inspector 会检查第 1 级和第 2 级推荐的所有配置。

定义扫描参数后，您可以选择将其作为一次性扫描（在完成配置后运行）还是定期扫描。定期扫描可以每天、每周或每月运行，时间由您选择。

> **Tip**
>
> 我们建议在扫描运行期间选择不太可能影响系统的日期和时间。

创建 CIS 扫描配置

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的 AWS 区域 选择器，选择要运行 CIS 扫描 AWS 区域 的位置。
3. 在导航面板的按需扫描下，选择 CIS 扫描。
4. 选择 “创建新扫描”。
    a. 输入扫描配置名称。
    b. 对于目标资源，输入要扫描的实例上标签的密钥和相应值。您总共可以指定 25 个要包含在扫描中的标签，对于每个密钥，您最多可以指定五个不同的值。
    c. 选择 CIS 基准水平。您可以为基本安全配置选择级别 1，为高级安全配置选择级别 2。
5. 对于 Target 帐户，请指定要在扫描中包括哪些帐户。组织中的独立账户或成员可以选择 “自我” 为其账户创建扫描配置。Amazon Inspector 授权管理员可以选择 “所有账户” 来定位组织内的所有账

户，或者选择 “指定账户” 并指定要定位的成员账户子集。授权的管理员可以输入SELF而不是账户 ID 来为自己的账户创建扫描配置。有关更多信息，请参阅 在 AWS 组织中管理 Amazon Inspector CIS 扫描的注意事项。

6. 为扫描选择时间表。在 “一次性扫描”（创建完扫描配置后立即运行）或 “重复扫描”（将在您选择的计划时间运行，直到将其删除）之间进行选择。
7. 选择 “创建” 以完成扫描配置的创建。

查看和编辑 CIS 扫描配置

您可以随时查看或编辑先前计划的扫描。

查看或编辑 CIS 扫描配置

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的 AWS 区域 选择器，选择创建 CIS 扫描配置 AWS 区域 的位置。
3. 在导航面板的按需扫描下，选择 CIS 扫描。
4. 选择 “计划” 以查看预设扫描配置。
5. 从 “扫描配置名称” 列中选择一项，打开该扫描配置的详细信息。
6. （可选）选择 “编辑” 以更改此扫描的参数。

查看 CIS 扫描结果

每次运行扫描配置时，Amazon Inspector 都会创建一个扫描任务，并在唯一的扫描 ID 下收集扫描结果。

扫描结果将在扫描完成后的 90 天内公布。您可以查看按支票或目标资源汇总的扫描结果。

通过支票汇总的扫描结果

扫描结果按扫描期间执行的每项检查进行分组。对于每项检查，您都会收到一份报告，说明有多少资源通过、失败或被跳过。

按资源汇总的扫描结果

扫描结果按扫描配置所针对的每个资源进行分组。对于每种资源，您都会收到一份报告，说明该资源的哪些检查通过、失败或跳过。

查看扫描结果

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的选择 AWS 区域 器，选择要查看扫描结果 AWS 区域 的位置。
3. 在导航面板的按需扫描下，选择 CIS 扫描。
4. 从 “扫描 ID” 列中选择要查看结果的扫描 ID。
5. 选择如何查看扫描结果：
   - 选择检查选项卡可查看按检查汇总的扫描结果。
     - 对于列出的检查，请在资源状态列中选择通过、已跳过或失败的数字，以打开按该状态和该检查筛选的资源视图。
   - 选择 “已扫描的资源” 选项卡，查看按资源汇总的扫描结果。
     - 选择一个资源以打开详细信息面板，其中列出了该资源通过、失败或跳过的检查。
6. （可选）使用任一视图中的筛选栏来优化结果。

您可以使用控制台或 API 下载 CIS 扫描的结果。

下载扫描结果

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的选择 AWS 区域 器，选择要查看扫描结果 AWS 区域 的位置。
3. 在导航面板的按需扫描下，选择 CIS 扫描。
4. 从 “扫描 ID” 列中选择要查看结果的扫描 ID。
5. 选择下载。如果您是授权管理员，则可以选择下载特定成员账户的结果。

在 AWS 组织中管理 Amazon Inspector CIS 扫描的注意事项

在组织内运行 CIS 扫描时，成员账户和 Amazon Inspector 委派的管理员会以不同的方式与 CIS 扫描配置和扫描结果进行交互。

当授权的管理员为所有帐户或成员帐户 ID 列表创建 CIS 扫描配置时，该组织将拥有该扫描配置。无论当前委派的管理员是哪个账户，都可以管理组织拥有的扫描配置，即使这些配置是由其他账户创建的。组织拥有的 CIS 扫描配置将有一个 ARN，该ARN将组织ID列为所有者，其模式如下：。`arn:aws:inspector2:Region:111122223333:owner/OrganizationId/cis-configuration/scanId`账户 ID 将是 Organizations 管理账户的 ID。

> **Important**
>
> 您无法向组织拥有的 CIS 扫描配置添加标签。

当授权的管理员创建扫描配置并指定SELF为目标帐户时，其帐户将拥有该扫描配置。即使他们离开了自己的组织，他们仍然可以管理该扫描配置。

> **Note**
>
> 授权的管理员无法更改目标扫描配置的目标SELF。

由成员账户、独立账户或以目标SELF为目标的委托管理员创建的扫描配置归创建这些配置的账户所有。这些 CIS 扫描配置有一个 ARN，该ARN将该账户列为所有者，其模式如下：。`arn:aws:inspector2:`*`Region`*`:`*`111122223333`*`:owner/`*`111122223333`*`/cis-configuration/`*`scanId`*账户 ID 将是创建扫描的账户。

组织中的成员帐户可以为自己的账户创建扫描配置。授权的管理员可以查看成员创建的扫描配置，但无法编辑或删除它们。如果成员账户离开组织，则委派的管理员将无法再看到该账户创建的扫描配置。

授权管理员可以查看组织中任何账户的扫描结果，包括成员安排的扫描结果。成员帐户可以查看其账户中资源的任何 CIS 扫描结果，包括由授权管理员安排的资源。

Amazon Inspector 拥有用于亚马逊 Inspector CIS 扫描的 Amazon S3 存储桶

Amazon Inspector 暂存 CIS 扫描所需的更新开放漏洞和评估语言 (OVAL) 定义文件。下表列出了 Amazon Inspector 拥有的所有带有 OVAL 定义的 Amazon S3 存储桶，CIS 扫描根据支持使用这些存储桶。 AWS 区域如有必要，应在 VPC 中将存储桶列入许可名单。

> **Note**
>
> 以下每个 Amazon Inspector 拥有的 Amazon S3 存储桶的详细信息都不会发生变化。但是，该列表可能会更新以反映新的支持 AWS 区域。您不能将这些存储桶用于其他 Amazon S3 操作或在您自己的 Amazon S3 存储桶中。

CIS 桶	AWS 区域
cis-datasets-prod-arn-5908f6f	欧洲（斯德哥尔摩）
cis-datasets-prod-bah-8f88801	中东（巴林）
cis-datasets-prod-bjs-0f40506	中国（北京）
cis-datasets-prod-bom-435a167	亚太地区（孟买）
cis-datasets-prod-cdg-f3a9c58	欧洲地区（巴黎）
cis-datasets-prod-cgk-09eb12f	亚太地区（雅加达）
cis-datasets-prod-cmh-63030b9	美国东部（俄亥俄州）
cis-datasets-prod-cpt-02c5c6f	非洲（开普敦）
cis-datasets-prod-dub-984936f	欧洲地区（爱尔兰）
cis-datasets-prod-fra-6eb96eb	欧洲地区（法兰克福）
cis-datasets-prod-gru-de69f99	South America（São Paulo）
cis-datasets-prod-hkg-8e30800	亚太地区（香港）
cis-datasets-prod-iad-8438411	美国东部（弗吉尼亚州北部）
cis-datasets-prod-icn-f4eff1c	亚太地区（首尔）
cis-datasets-prod-kix-5743b21	亚太地区（大阪）
cis-datasets-prod-lhr-8b1fbd0	欧洲地区（伦敦）
cis-datasets-prod-mxp-7b1bbce	欧洲地区（米兰）
cis-datasets-prod-nrt-464f684	亚太地区（东京)
cis-datasets-prod-osu-5bead6f	AWS GovCloud （美国东部）
cis-datasets-prod-pdt-adadf9c	AWS GovCloud （美国西部）

CIS 桶	AWS 区域
cis-datasets-prod-pdx-acfb052	US West（Oregon）
cis-datasets-prod-sfo-1515ba8	美国西部（北加利福尼亚）
cis-datasets-prod-sin-309725b	亚太地区（新加坡）
cis-datasets-prod-syd-f349107	亚太地区（悉尼）
cis-datasets-prod-yul-5e0c95e	加拿大（中部）
cis-datasets-prod-zhy-5a8eacb	中国（宁夏）
cis-datasets-prod-zrh-67e0e3d	欧洲（苏黎世）

评测 Amazon Inspector 对 AWS 环境的覆盖率

为了帮助您评测和解释 Amazon Inspector 对您AWS环境的覆盖率，Amazon Inspector 控制台上的账户管理页面提供了有关 Amazon Inspector 扫描您的账户和资源的状态的统计数据和详细信息。通过此页面，您可以查看资源的汇总统计数据和其他数据。您还可以深入分析 Amazon Inspector 对各个资源的覆盖率，并深入查看特定资源的调查发现。如果您是组织的 Amazon Inspector 委托管理员，则此数据将包括组织所有账户的统计数据和详细信息。

评测 Amazon Inspector 对 AWS 环境的覆盖率

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 在导航窗格中，选择账户管理。
3. 在账户管理页面上，选择五种不同的覆盖率视图选项卡中的一种：

   - 账户，用于账户级别的覆盖率。
   - 实例，用于 Amazon Elastic Compute Cloud (Amazon EC2) 实例的覆盖率。
   - 存储库，用于 Amazon Elastic Container Registry (Amazon ECR) 存储库的覆盖率。
   - 映像，用于 Amazon ECR 容器映像的覆盖率。
   - Lambda，用于 Lambda 函数的覆盖率。

本节中的主题介绍了每个选项卡提供的信息，包括单个资源可能具有的扫描状态。

主题

评测账户级别的覆盖率

如果您的账户不是组织的一员，或者不是组织的 Amazon Inspector 委托管理员账户，则账户选项卡会提供有关您的账户和账户资源扫描状态的信息。在此选项卡上，您可以激活或停用对您账户中所有或仅特定类型资源的扫描。有关更多信息，请参阅使用 Amazon Inspector 自动扫描资源。

如果您的账户是组织的 Amazon Inspector 委托管理员账户，则账户选项卡会提供组织中账户的自动激活设置并列出组织中的所有账户。对于每个账户，该列表都会显示账户是否已激活 Amazon Inspector，如果已激活，还会显示为该账户激活的资源扫描类型。作为委托管理员，您可以使用此选项卡更改组织的自动激活设置。您还可以为个人成员账户激活或停用特定类型的资源扫描。有关更多信息，请参阅为成员账户激活 Amazon Inspector 扫描。

评测 Amazon EC2 实例的覆盖率

实例选项卡显示环境中AWS的 Amazon EC2 实例。列表按以下选项卡分组：

- 全部 – 显示环境中的所有实例。状态列显示实例的当前扫描状态。
- 扫描 – 显示 Amazon Inspector 在环境中主动监控和扫描的所有实例。
- 未扫描 – 显示 Amazon Inspector 未在环境中主动监控和扫描的所有实例。原因列说明了为什么 Amazon Inspector 没有监控和扫描实例。

  EC2 实例可能由于多种原因出现在未扫描选项卡上。Amazon Inspector 使用 AWS Systems Manager (SSM) 和 SSM 代理来自动监控和扫描 EC2 实例是否存在漏洞。如果实例未运行 SSM 代理，没有支持 Systems Manager 的 AWS Identity and Access Management (IAM) 角色，或者未运行支持的操作系统或架构，则 Amazon Inspector 无法监控和扫描相应实例。有关更多信息，请参阅扫描 Amazon EC2 实例。

在每个选项卡上，账户列指定拥有实例的 AWS 账户。

EC2 实例标签 – 此列显示与实例关联的标签，可用于确定您的实例是否已按标签排除在扫描之外。

操作系统 – 此列显示操作系统类型，可以是 `WINDOWS`、`MAC`、`LINUX` 或 `UNKNOWN`。

使用已监控 – 此列显示 Amazon Inspector 对此实例使用的是基于代理还是无代理的扫描方法。

上次扫描时间 – 此列显示 Amazon Inspector 上次检查资源是否存在漏洞的时间。Amazon Inspector 执行扫描的频率取决于它用来扫描实例的扫描方法。

要查看有关 EC2 实例的更多详细信息，请选择 EC2 实例列中的链接。然后，Amazon Inspector 会显示有关该实例的详细信息以及该实例的当前调查发现。要查看调查发现的详细信息，请选择标题列中的链接。如需了解这些详细信息，请参阅Amazon Inspector 调查发现详细信息。

正在扫描 Amazon EC2 实例的状态值

对于 Amazon Elastic Compute Cloud (Amazon EC2) 实例，可能的状态值包括：

- 主动监控 – Amazon Inspector 正在持续监控和扫描实例。
- EC2 实例已停止 – 由于实例处于停止状态，Amazon Inspector 暂停了对实例的扫描。所有现有调查发现都将持续到实例终止。如果实例重新启动，Amazon Inspector 将自动恢复对实例的扫描。
- 内部错误 – Amazon Inspector 尝试扫描实例时发生内部错误。Amazon Inspector 将自动处理错误并尽快恢复扫描。
- 无清单 – Amazon Inspector 找不到用于扫描实例的软件应用程序清单。实例的 Amazon Inspector 关联可能已被删除或者无法运行。

  要修复此问题，请使用 AWS Systems Manager 来确保 `InspectorInventoryCollection-do-not-delete` 关联存在且其关联状态为成功。此外，使用 AWS Systems Manager Fleet Manager 验证实例的软件应用程序清单。
- 待禁用 – Amazon Inspector 已停止扫描该实例。正在禁用该实例，等待清理任务完成。
- 等待初始扫描 – Amazon Inspector 已将实例纳入队列，等待初始扫描。
- 资源已终止 – 实例已终止。Amazon Inspector 当前正在清理该实例的现有调查发现和覆盖率数据。
- 清单过期 – Amazon Inspector 无法收集过去 7 天内为该实例捕获的更新后的软件应用程序清单。

  要修复此问题，请使用 AWS Systems Manager 来确保该实例存在所需的 Amazon Inspector 关联并且关联正在运行。此外，使用 AWS Systems Manager Fleet Manager 验证实例的软件应用程序清单。
- 非托管型 EC2 实例 – Amazon Inspector 未监控或扫描实例。实例不由 AWS Systems Manager 托管。

  要修复此问题，您可以使用 AWS Systems Manager Automation 提供的 AWSSupport-TroubleshootManagedInstance runbook。在您配置 AWS Systems Manager 来托管实例后，Amazon Inspector 将自动开始持续监控和扫描该实例。
- 不支持的操作系统 – Amazon Inspector 未监控或扫描实例。实例使用了 Amazon Inspector 不支持的操作系统或架构。有关 Amazon Inspector 支持的操作系统的列表，请参阅Amazon EC2 扫描支持的操作系统。
- 主动监视且存在部分错误 – 此状态表示 EC2 扫描处于活动状态，但存在与 Amazon EC2 Linux 实例的 Amazon Inspector 深度检查 相关的错误。可能的深度检查错误有：
  - 已@@ 超过深度检查包裹收集限制 — 该实例已超过 Amazon Inspector 深度检查的 5000 个包裹限制。要恢复对此实例的深入检查，您可以尝试调整与该账户关联的自定义路径。
  - 已超过深度检查每日 ssm 库存限制 — SSM 代理无法向 Amazon Inspector 发送库存，因为该实例每天收集的库存数据的 SSM 配额已经达到。有关更多信息，请参阅 Amazon EC2 Systems Manager 端点和配额。

- 已超过深度检查收集时间限制 — Amazon Inspector 未能提取包裹库存，因为包裹收集时间超过了 15 分钟的最大阈值。
- 深度检查没有清单 – Amazon Inspector SSM 插件尚未能够收集此实例的程序包清单。这通常是待处理扫描的结果，但是，如果此状态在 6 小时后仍然存在，请使用 Amazon EC2 Systems Manager 确保该实例存在所需的 Amazon Inspector 关联并且关联正在运行。

有关为 EC2 实例配置扫描设置的详细信息，请参阅扫描 Amazon EC2 实例。

评测 Amazon ECR 存储库的覆盖率

存储库选项卡显示AWS环境中的 Amazon ECR 存储库。列表按以下选项卡分组：

- 全部 – 显示环境中的所有存储库。状态列显示存储库的当前扫描状态。
- 已激活 – 显示根据 Amazon Inspector 配置要在环境中监控和扫描的所有存储库。状态列显示存储库的当前扫描状态。
- 已激活 – 显示 Amazon Inspector 未在环境中监控和扫描的所有存储库。原因列说明了为什么 Amazon Inspector 没有监控和扫描存储库。

在每个选项卡上，账户列指定拥有存储库的 AWS 账户。

要查看有关存储库的其他详细信息，请选择存储库的名称。然后，Amazon Inspector 会显示存储库中的容器映像的列表以及每个映像的详细信息。详细信息包括映像标签、映像摘要和扫描状态。其中还包括关键调查发现统计数据，例如映像的关键调查发现数量。要深入了解和查看调查发现统计数据的支持数据，请选择映像的映像标签。

正在扫描 Amazon ECR 存储库的状态值

对于 Amazon Elastic Container Registry (Amazon ECR) 存储库，可能的状态值为：

- 已激活（持续）— 对于存储库，Amazon Inspector 会持续监控该存储库中的图像。存储库的增强扫描设置为持续扫描。Amazon Inspector 最初会在推送新图像时对其进行扫描，如果发布了与该图像相关的新 CVE，则会重新扫描图像。在您配置的 ECR 扫描持续时间内，Amazon Inspector 将继续监控此存储库中的图像。
- 已激活（推送时）— 当推送新映像时，Amazon Inspector 会自动扫描存储库中的单个容器映像。已激活存储库的增强扫描，并将其设置为推送时扫描。
- 访问被拒绝 – Amazon Inspector 无法访问存储库或存储库中的任何容器映像。

要修复此问题，请确保存储库的 AWS Identity and Access Management (IAM) 策略允许 Amazon Inspector 访问存储库。

- 已停用（手动）– Amazon Inspector 未监控或扫描存储库中的任何容器映像。存储库的 Amazon ECR 扫描设置为基本手动扫描。

  要开始使用 Amazon Inspector 扫描存储库中的映像，请将存储库的扫描设置更改为增强扫描，然后选择是持续扫描映像还是仅在推送新映像时扫描映像。
- 已激活（推送时）— 当推送新映像时，Amazon Inspector 会自动扫描存储库中的单个容器映像。存储库的增强扫描设置为推送时扫描。
- 内部错误-Amazon Inspector 尝试扫描存储库时出现内部错误。Amazon Inspector 将自动处理错误并尽快恢复扫描。

有关配置存储库扫描设置的详细信息扫描 Amazon ECR 容器映像。

评测 Amazon ECR 容器映像的覆盖率

映像选项卡显示AWS环境中的 Amazon ECR 容器映像。列表按以下选项卡分组：

- 全部 – 显示环境中的所有容器映像。状态列显示映像的当前扫描状态。
- 正在扫描 – 显示根据 Amazon Inspector 配置要在环境中监控和扫描的所有容器映像。状态列显示映像的当前扫描状态。
- 未扫描 – 显示 Amazon Inspector 未在环境中监控和扫描的所有容器映像。原因列说明了为什么 Amazon Inspector 没有监控和扫描映像。

  容器映像可能会由于多种原因出现在未激活选项卡上。映像可能存储在未激活 Amazon Inspector 扫描的存储库中，或者 Amazon ECR 筛选规则阻止扫描该存储库。或者在您为 ECR 重新扫描持续时间配置的天数内未推送或拉取映像。有关更多信息，请参阅配置 ECR 重新扫描持续时间。

在每个选项卡上，存储库名称列指定存储容器映像的存储库的名称。账户列指定拥有该存储库的 AWS 账户。上次扫描列显示 Amazon Inspector 上次检查资源是否存在漏洞的时间。这可能包括在更新调查发现元数据时、更新资源的应用程序清单时，或者针对新 CVE 重新扫描时进行检查。有关更多信息，请参阅Amazon ECR 扫描的扫描行为。

要查看有关容器映像的其他详细信息，请选择 ECR 容器映像列中的链接。然后，Amazon Inspector 会显示有关该映像的详细信息以及该映像的当前调查发现。要查看调查发现的详细信息，请选择标题列中的链接。如需了解这些详细信息，请参阅Amazon Inspector 调查发现详细信息。

Amazon ECR 容器镜像的扫描状态值

对于 Amazon 弹性容器注册表容器镜像，可能的状态值为：

- 主动监控（持续）— Amazon Inspector 会持续监控，每当发布新的相关 CVE 时，都会对其进行图像和新的扫描。每当推送或拉取图像时，都会刷新图像的 Amazon ECR 重新扫描持续时间。存储映像的存储库启用了增强扫描，存储库的增强扫描设置为持续扫描。
- 已激活（推送时）— 每次推送新图像时，Amazon Inspector 都会自动扫描图像。存储映像的存储库启用了增强扫描，存储库的增强扫描设置为推送时扫描。
- 内部错误-Amazon Inspector 尝试扫描容器图像时出现内部错误。Amazon Inspector 将自动处理错误并尽快恢复扫描。
- 等待初始扫描 — Amazon Inspector 已将图像排队等候初始扫描。
- 扫描资格已过期（持续）— Amazon Inspector 已暂停扫描图片。在您为自动重新扫描存储库中的映像指定的持续时间内，映像尚未更新。您可以推送或拉动图像以恢复扫描。
- 扫描资格已过期（推送中）— Amazon Inspector 已暂停对图片的扫描。在您为自动重新扫描存储库中的映像指定的持续时间内，映像尚未更新。您可以推送图像以恢复扫描。
- 手动扫描频率（手动）– Amazon Inspector 不会扫描 Amazon ECR 容器映像。存储映像的存储库的 Amazon ECR 扫描设置为基本手动扫描。要开始使用 Amazon Inspector 自动扫描映像，请将存储库设置为增强扫描，然后选择持续扫描映像或者仅在推送新映像时扫描。
- 不支持的操作系统 — Amazon Inspector 没有监控或扫描图像。该映像基于 Amazon Inspector 不支持的操作系统，或者它使用了 Amazon Inspector 不支持的媒体类型。

  有关 Amazon Inspector 支持的操作系统的列表，请参阅Amazon ECR 扫描支持的操作系统。有关 Amazon Inspector 支持的媒体类型的列表，请参阅支持的媒体类型。

有关为存储库和映像配置扫描设置的详细信息，请参阅扫描 Amazon ECR 容器映像。

评测 AWS Lambda 函数覆盖率

Lambda 选项卡显示AWS环境中的 Lambda 函数。本页有两个表，一个显示 Lambda 标准扫描的函数覆盖率详细信息，另一个显示 Lambda 代码扫描的函数覆盖率详细信息。您可以根据以下选项卡对函数进行分组：

- 全部 – 显示环境中的所有 Lambda 函数。状态列显示 Lambda 函数的当前扫描状态。
- 扫描 – 显示根据 Amazon Inspector 配置要扫描的 Lambda 函数。状态列显示每个 Lambda 函数的当前扫描状态。

- 未扫描 – 显示根据 Amazon Inspector 配置未扫描的 Lambda 函数。原因列说明了为什么 Amazon Inspector 没有监控和扫描函数。

  Lambda 函数可能由于多种原因出现在未扫描选项卡上。Lambda 函数可能属于尚未添加到 Amazon Inspector 的账户，或者筛选规则阻止扫描此函数。有关更多信息，请参阅扫描 AWS Lambda 功能。

在每个选项卡上，函数名称列指定 Lambda 函数的名称。账户列指定拥有该函数的 AWS 账户。运行时系统指定函数的运行时系统。状态列显示每个 Lambda 函数的当前扫描状态。资源标签显示已应用于函数的标签。上次扫描列显示 Amazon Inspector 上次检查资源是否存在漏洞的时间。这可能包括在更新调查发现元数据时、更新资源的应用程序清单时，或者针对新 CVE 重新扫描时进行检查。有关更多信息，请参阅Lambda 函数扫描的扫描行为。

正在扫描AWS Lambda函数的状态值

对于 Lambda 函数，可能的状态值包括：

- 主动监控 – Amazon Inspector 正在持续监控和扫描 Lambda 函数。持续扫描包括在将新函数推送到存储库时对其进行初始扫描，以及在函数更新或发布新的常见漏洞和风险 (CVE) 时自动重新扫描函数。
- 按标签排除 – Amazon Inspector 未扫描此函数，因为按标签它已被排除在扫描范围之外。
- 扫描资格已过期 – Amazon Inspector 未监控此函数，因为自上次调用或更新该函数已过去 90 天或更长时间。
- 内部错误 – Amazon Inspector 尝试扫描函数时发生内部错误。Amazon Inspector 将自动处理错误并尽快恢复扫描。
- 等待初始扫描 – Amazon Inspector 已将函数纳入队列，等待初始扫描。
- 不支持 – Lambda 函数的运行时系统不受支持。

在 Amazon Inspector 中使用组织管理多个账户

您可以使用 Amazon Inspector 管理通过 AWS Organizations 关联的多个账户。为了管理多个 Amazon Inspector 账户，Organizations 管理账户将组织内的一个账户指定为亚马逊检查员的委托管理员账户。该委托管理员将为组织管理 Amazon Inspector，并将获得代表您的组织执行任务的特殊权限。这些任务包括激活或停用对成员帐户的扫描、查看整个组织的聚合查找数据，以及创建和管理禁止规则。

> **Note**
>
> 要以编程方式为多个账户启用 Amazon InspectorAWS 区域，你可以使用 Amazon Inspector 开发的 shell 脚本。有关使用此脚本的更多信息，请参阅网站上的 insp ector2-enablement-with-cli。 GitHub

主题

了解 Amazon Inspector 管理员和成员账户之间的关系

在多账户环境中使用 Amazon Inspector 时，Amazon Inspector 委托管理员账户可以访问某些元数据。这些元数据包括 Amazon EC2 和 Amazon ECR 的配置数据以及成员账户的安全调查发现结果。管理员账户还可以创建应用于成员账户的调查发现抑制规则。有关更多信息，请参阅使用抑制规则抑制 Amazon Inspector 调查发现。

委派管理员操作

通常，当授权管理员将设置应用于其账户时，这些设置将应用于组织中的所有其他账户。委派管理员还可以查看和检索自有账户和任何关联成员的信息。Amazon Inspector 委派管理员账户可以执行以下操作：

- 查看和管理关联账户的 Amazon Inspector 状态，包括激活和停用 Amazon Inspector。
- 为组织内的所有成员账户激活或停用扫描类型。
- 查看整个组织的汇总调查发现数据，以及组织内所有成员账户的调查发现详情。
- 创建和管理应用于组织内所有账户的调查发现的抑制规则。
- 为组织的所有成员激活 Amazon ECR 增强扫描。

- 查看整个组织的资源覆盖率。
- 为组织内所有成员账户定义自动重新扫描 ECR 容器映像的持续时间。委托管理员的扫描持续时间设置会覆盖成员账户先前的所有设置。组织中的所有账户共享授权管理员的 Amazon ECR 自动重新扫描时长。您不能为个人账户设置不同的重新扫描持续时间。
- 为 Amazon Inspector 的 Amazon EC2 深度检查指定五个自定义路径，这些路径将在组织中的所有账户中使用。除此之外，委托管理员还可为个人账户设置五个自定义路径。有关配置深度检查自定义路径的更多信息，请参阅Amazon Inspector 深度检查的自定义路径。
- 激活和停用对成员账户的 Amazon Inspector 深度检查。
- 为组织内的任何成员账户导出 SBOM。
- 为组织中的所有成员账户设置 Amazon EC2 扫描模式。有关更多信息，请参阅管理扫描模式。
- 创建和管理组织中所有帐户的 CIS 扫描配置，但成员帐户创建的任何扫描配置除外。

> **Note**
>
> 如果成员账户离开组织，则委派的管理员将无法再看到该账户安排的扫描配置。

- 查看组织中所有帐户的 CIS 扫描结果。

成员账户操作

成员账户可以在 Amazon Inspector 中查看和检索有关其账户的信息，而其账户的设置则由授权管理员管理。组织内的成员账户可以在 Amazon Inspector 中执行以下操作：

- 为自己的账户激活 Amazon Inspector。
- 查看自己账户的资源覆盖率。
- 查看自己账户的调查发现详细信息。
- 查看自己账户的 ECR 容器映像自动重新扫描持续时间设置。
- 为 Amazon Inspector 的 EC2 深度检查指定五个自定义路径，这些路径将用于他们的个人账户。除了委派管理员为组织指定的任何自定义路径外，还会扫描这些路径。有关配置深度检查路径的更多信息，请参阅Amazon Inspector 深度检查的自定义路径。
- 查看您的委托管理员为 Amazon Inspector 深度检查设置的自定义路径。
- 为与其账户关联的任何资源导出 SBOM。
- 查看其账户的扫描模式。
- 为其账户创建和管理 CIS 扫描配置。

- 查看其账户中资源的任何 CIS 扫描结果，包括由授权管理员安排的扫描。

> **Note**
>
> 激活后，只有委托管理员账户才能停用 Amazon Inspector。

指定 Amazon Inspector 委托管理员

委托管理员的重要注意事项

请注意以下因素，它们定义了委托管理员在 Amazon Inspector 中如何操作：

一名委托管理员最多可以管理 5,000 名成员。

每个 Amazon Inspector 委托管理员有 5,000 个成员账户的配额。但是，您的组织可能有超过 5,000 个账户。如果您的成员账户超过 5,000 个，您将通过 Amazon Person CloudWatch al Health Dashboard 收到通知，并向委托管理员账户发送一封电子邮件。

委托管理员是区域性的。

与 AWS Organizations 不同，Amazon Inspector 是一项区域服务。这意味着您必须指定委托管理员，添加成员账户，并在要在其中使用 Amazon Inspector 的每个AWS 区域账户中激活扫描类型。

一个组织只能有一名委托管理员。

您只能为一个组织指定一名 Amazon Inspector 委托管理员。如果您已将一个账户指定为一个区域的委托管理员，则该账户必须是您在所有其他区域的委托管理员。

更改委托管理员不会停用成员账户的 Amazon Inspector。

如果您移除授权的管理员，则这些账户中的 Amazon Inspector 不会被停用，扫描设置也不会受到影响。

您的 AWS 组织必须激活所有功能。

这是的默认设置AWS Organizations。如果尚未激活，请参阅激活组织中的所有功能。

指定委托管理员所需的权限

您必须拥有激活 Amazon Inspector 和指定 Amazon Inspector 委托管理员的权限。

将以下语句添加到 IAM policy 的末尾以授予这些权限。

```
{
    "Sid": "PermissionsForInspectorAdmin",
    "Effect": "Allow",
    "Action": [
        "inspector2:EnableDelegatedAdminAccount",
        "organizations:EnableAWSServiceAccess",
        "organizations:RegisterDelegatedAdministrator",
        "organizations:ListDelegatedAdministrators",
        "organizations:ListAWSServiceAccessForOrganization",
        "organizations:DescribeOrganizationalUnit",
        "organizations:DescribeAccount",
        "organizations:DescribeOrganization"
    ],
    "Resource": "*"
}
```

为您的 AWS 组织指定委托管理员

以下步骤为您演示了如何为您 AWS 组织指定委托管理员。指定完成后，将为组织管理账户和所选委托管理员账户激活 Amazon Inspector。

> **Note**
>
> 只有组织管理账户才能指定委托管理员。

首次激活 Amazon Inspector 会`AWSServiceRoleForAmazonInspector`为账户创建服务相关角色 (SLR)。有关 Amazon Inspector 如何使用服务相关角色的更多信息，请参阅对 Amazon Inspector 使用服务相关角色。有关服务相关角色的一般信息，请参阅 IAM 用户指南中的使用服务相关角色。

指定 Amazon Inspector 委托管理员

Console

在控制台中指定委托管理员

1. 使用 AWS Organizations 管理账户登录 AWS Management Console。
2. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home，然后使用右上角的AWS 区域选择器指定您要在其中指定管理员的区域。

3. 在授权管理员窗格中，输入您要指定为组织的 Amazon Inspector 委托管理员的十二位数账户 ID。AWS 账户然后选择 “委托管理”。
4. （推荐）在每个 AWS 区域中重复执行上述步骤。

API

使用 API 指定委托管理员

- 使用 Organizations 管理账户AWS 账户的凭据运行 EnableDelegatedAdminAccountAPI 操作。您也可以通过运行AWS Command Line Interface以下 CLI 命令来执行此操作：`aws inspector2 enable-delegated-admin-account --delegated-admin-account-id 11111111111`。

> **Note**
>
> 请务必指定您想要成为 Amazon Inspector 委托管理员的账户的账户 ID。

指定委托管理员后，您只能使用AWS Organizations管理账号来更改或移除委托管理员账号。

为成员账户激活 Amazon Inspector 扫描

作为组织的委托管理员，您可以为与 AWS Organizations 管理账户关联的所有成员激活 Amazon EC2 扫描、Amazon ECR 扫描或同时激活两者。为成员账户激活扫描后，该账户将与委托管理员关联，Amazon Inspector 会自动激活，对所选类型的扫描会立即开始。有关可以扫描哪些资源以及如何配置扫描的信息，请参阅使用 Amazon Inspector 自动扫描资源。

Amazon Inspector 提供了多种用来为成员账户管理和激活扫描的选项，包括允许成员账户激活 Amazon Inspector。使用以下选项之一开始对成员账户的扫描。

要为所有成员账户自动激活扫描

1. 登录委托管理员账户。
2. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。然后使用右上角的AWS 区域选择器指定要激活扫描所有成员帐户的区域。
3. 在导航窗格中的设置下，选择账户管理。账户表显示了与 AWS Organizations 管理账户关联的所有成员账户。

4. 选中表顶部的复选框以选择此页面上的所有账户。然后选择激活，并从菜单中选择偏好的扫描类型选项。

> **Note**
>
> 仅选择页面上当前可见的账户。如果您有多页帐户，则必须在每个页面上重复此过程。要更改页面上显示的账户数量，请选择齿轮图标。

5. 打开 “为新成员账户自动激活 Inspec tor” 设置，然后选择扫描类型以激活添加到组织中的任何新成员。
6. （推荐）在要扫描成员账户的每个区域重复这些步骤。

为新成员账户自动激活 Inspector 设置可为组织的所有未来成员激活 Amazon Inspector。这样，您的 Amazon Inspector 委托管理员就可以管理添加到组织中的所有新成员。当成员账户数量达到 5,000 的配额时，此设置将自动关闭。如果有账户被删除，成员总数减少到少于 5,000，则该设置将自动重新激活。

有选择地激活成员账户

1. 登录委托管理员账户。
2. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home，然后使用右上角的AWS 区域选择器指定要激活扫描某些成员账户的区域。
3. 在导航窗格中的设置下，选择账户管理。账户表显示了与 AWS Organizations 管理账户关联的所有成员账户。
4. 在账户管理页面上，选中要激活扫描的每个成员账户对应的复选框。
5. 选择激活。
6. 从激活菜单中选择要为所选账户激活的扫描类型。可以选择以下扫描选项：
   - 全部扫描-激活所有扫描类型。
   - EC2 扫描 — 激活对亚马逊 EC2 实例的扫描。
   - ECR 容器扫描 — 激活 ECR 容器映像的扫描。
   - AWS Lambda标准扫描 — 激活 Lambda 函数的扫描。
7. （推荐）在要激活对特定成员的扫描的每个区域重复这些步骤。

如果您的AWS Organizations管理账户已为 Amazon Inspector 委派了管理员，则您可以以成员身份激活自己的账户并查看自己账户的扫描详情。

以成员账户身份激活扫描

1. 登录自己的账户。
2. 通过 https://console.aws.amazon.com/inspector/v2/home 打开 Amazon Inspector 控制台，然后使用右上角的AWS 区域选择器指定要激活扫描的区域。
3. 在导航窗格中的设置下，选择账户管理。
4. 在账户管理页面上，选中您的账户对应的复选框。
5. 从激活菜单中选择要激活的扫描类型。可以选择以下扫描选项：
   - 全部扫描-激活所有扫描类型。
   - EC2 扫描 — 激活对亚马逊 EC2 实例的扫描。
   - ECR 容器扫描 — 激活 ECR 容器映像的扫描。
   - AWS Lambda标准扫描 — 激活 Lambda 函数的扫描。
6. （推荐）在要激活扫描的每个区域重复这些步骤。

在 Amazon Inspector 中取消成员账户的关联

以下过程展示了如何取消成员账户的关联。已取消关联的成员账户作为独立的 Amazon Inspector 账户保留在您的AWS Organizations组织中。Amazon Inspector 委派的管理员不再有权激活和管理这些账户的 Amazon Inspector。您可以稍后再次将已取消关联的账户添加为成员。

> **Note**
>
> 取消关联账户不会停用 Amazon Inspector 对该账户的扫描。

Console

使用控制台取消成员账户的关联

1. 登录委托管理员账户。
2. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home，然后使用右上角的AWS 区域选择器指定要取消关联一个或多个成员账户的区域。

3. 在导航窗格中的设置下，选择账户管理。
4. 在账户管理页面上，选中要取消关联的每个账户对应的复选框。
5. 从操作菜单中选择取消账户的关联。
6. （推荐）在要取消关联账户的每个区域重复这些步骤。

API

使用 API 取消成员账户的关联

运行 DisassociateMember API 操作。在请求中，提供您要取消关联的账户 ID。

移除 Amazon Inspector 委托管理员

如果您必须分配新的 Amazon Inspector 委托管理员，则可以移除现有委托管理员作为AWS Organizations管理账户。

当您移除委托管理员时，它不会停用该账户或任何组织成员账户中的 Amazon Inspector。组织内的帐户将转换为独立帐户，并保留其在由授权管理员管理之前的扫描设置。

移除委托管理员

1. 使用 AWS Organizations 管理账户登录 AWS Management Console。
2. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home，然后使用右上角的AWS 区域选择器指定要移除委托管理员的区域。
3. 在导航窗格中的设置下，选择账户管理。
4. 在委托管理员部分中，选择移除，然后确认操作。
5. 在您注册此委托管理员的每个区域中重复这些步骤。

在添加新的 Amazon Inspector 委托管理员时，必须手动将组织成员关联到新的管理员账户。使用以下步骤将组织成员关联到新的管理员帐户。

将成员与新的委托管理员关联

1. 使用委托管理员账户登录 AWS Management Console。
2. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home，然后使用右上角的AWS 区域选择器指定要将成员与新委托管理员关联的区域。

3. 在导航窗格中的设置下，选择账户管理。
4. 使用顶部的复选框选择组织中列出的所有账户。
5. 从操作菜单中选择添加成员。
6. 在要将成员与新授权管理员关联的每个区域重复这些步骤。

监控 Amazon Inspector 的使用量和成本

您可以使用 Amazon Inspector 控制台和 API 操作来预测在环境中使用 Amazon Inspector 的每月成本。如果您是多账户环境的 Amazon Inspector 管理员，则可以查看整个环境的总成本以及每个成员账户的成本指标。

使用“使用量”控制台

您可以通过控制台评测 Amazon Inspector 的使用量和预计成本。

访问“使用量”统计数据

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的 AWS 区域选择器选择要监控成本的区域。
3. 在导航窗格中，选择使用量。

在按账户选项卡中，您将看到账户使用量下列出的 30 天期间的预计总费用。在预计成本列下的表格中，选择一个值，以查看该账户按扫描类型划分的使用量明细。在此详细信息窗格中，您还可以看到该账户的哪些扫描类型激活了免费试用。

如果您是某组织的委托管理员，您将在表格中看到组织内每个账户对应的行。如果您组织中的某个账户已取消关联，则控制台会将其预计费用显示为 -。

在按扫描类型选项卡中，您可以看到当前 30 天内按扫描类型划分的使用量明细。这些信息用于计算按账户选项卡中的预计成本。

如果您是某组织的委托管理员，您可以看到组织内每个账户的使用量。

在此选项卡中，您可以展开以下任一窗格以获取使用量统计数据：

Amazon EC2 扫描

Amazon Inspector 使用控制台跟踪以下基于代理的扫描和无代理扫描的指标：

- 实例（平均值）— Amazon Inspector 使用覆盖时间来计算 EC2 实例扫描的平均资源数量。平均值等于总覆盖时间除以 720 小时（30 天内的小时数）。
- 覆盖时间 — 对于 Amazon EC2 扫描，这是过去 30 天内 Amazon Inspector 为账户中的每个 EC2 实例提供有效覆盖的总小时数。对于 EC2 实例，覆盖时间是指从 Amazon Inspector 发现实例到

实例被终止或停止，或者按标签排除在扫描之外的时间。（当您重启已停止的实例或删除排除标签时，Amazon Inspector 将恢复覆盖，并且该实例的覆盖时间将继续累积）。

CIS 实例扫描-对账户中的实例执行的 CIS 扫描总数。

Amazon ECR 扫描

初始扫描 — 过去 30 天内首次扫描账户中映像的总次数。

重新扫描 — 过去 30 天内重新扫描账户中映像的总次数。重新扫描是指对 Amazon Inspector 之前扫描过的 ECR 映像进行的所有扫描。如果您已将 ECR 存储库配置为持续扫描，则当 Amazon Inspector 向其数据库添加新的常见漏洞和风险 (CVE) 时，会自动进行重新扫描。

Lambda 扫描

Amazon Inspector 使用控制台会跟踪 Lambda 标准扫描和 Lambda 代码扫描的以下指标：

- Lambda 函数数量 (Avg) — Amazon Inspector 使用服务时间来计算 Lambda 函数扫描的平均函数数。平均值等于总覆盖时间除以 720 小时（30 天内的小时数）。
- 覆盖时间 — 对于 Lambda 函数扫描，这是过去 30 天内 Amazon Inspector 为账户中的每个 Lambda 函数提供有效覆盖的总小时数。对于 AWS Lambda 函数，覆盖时间是指从 Amazon Inspector 发现函数到函数被删除或从扫描中排除的时间。如果被排除的函数再次被纳入，则该函数的覆盖时间将继续累积。

了解 Amazon Inspector 如何计算使用成本

Amazon Inspector 提供的成本是估算值，不是实际成本，因此它们可能与您的 AWS Billing 控制台中的成本不同。

请注意以下有关 Amazon Inspector 如何在使用量页面上计算成本的信息：

- 使用成本仅反映当前区域的情况。每种扫描类型的价格因 AWS 区域而异，要查看每个区域的确切价格，请参阅 Amazon Inspector 的定价
- 所有使用量预测均按美元四舍五入。
- 预计费用不包含折扣。
- 预计成本代表每种扫描类型 30 天使用期内的总成本。如果账户的使用天数少于 30 天，Amazon Inspector 会预测 30 天后的费用，并假设当前覆盖的所有资源仍将在 30 天的剩余时间内继续保持覆盖。
- 每种扫描类型的成本根据以下方式计算：
  - EC2 扫描：成本反映过去 30 天内 Amazon Inspector 覆盖的 EC2 实例的平均数量。

- ECR 容器扫描：成本反映过去 30 天内初始映像扫描次数 + 映像重新扫描次数的总和。
- Lambda 标准扫描：成本反映过去 30 天内 Amazon Inspector 覆盖的 Lambda 函数的平均数量。
- Lambda 代码扫描：成本反映过去 30 天内 Amazon Inspector 覆盖的 Lambda 函数的平均数量。

关于 Amazon Inspector 免费试用

激活 Amazon Inspector 扫描类型后，您将自动注册该扫描类型的 15 天免费试用。每种扫描类型都有独立的免费试用，包括：EC2 扫描、ECR 扫描、Lambda 标准扫描和 Lambda 代码扫描。

> **Note**
>
> 免费试用版不适用于 CIS 扫描。

如果您在免费试用期间停用某一扫描类型，则该扫描类型的免费试用将暂停。如果您重新激活该服务，免费试用将恢复，您将获得此次免费试用的剩余天数。

Amazon Inspector 安全性

云安全 AWS 是重中之重。作为 AWS 客户，您可以受益于专为满足大多数安全敏感型组织的要求而构建的数据中心和网络架构。

安全是双方共同承担 AWS 的责任。责任共担模式将其描述为云的安全性和云中的安全性：

- 云安全 — AWS 负责保护在云中运行 AWS 服务的基础架构 AWS Cloud。 AWS 还为您提供可以安全使用的服务。作为AWS 合规计划合规计划合规计划合的一部分，第三方审计师定期测试和验证我们安全的有效性。要了解适用于 Amazon Inspector 的合规计划，请参阅合规计划范围内的 AWSAWS 服务按合分的范围内服务。
- 云端安全-您的责任由您使用的 AWS 服务决定。您还需要对其他因素负责，包括您的数据的敏感性、您公司的要求以及适用的法律法规。

该文档帮助您了解如何在使用 Amazon Inspector 时应用责任共担模式。以下主题说明如何配置 Amazon Inspector 以实现您的安全性和合规性目标。您还将学习如何使用其他 AWS 服务来帮助您监控和保护您的 Amazon Inspector 资源。

主题

Amazon Inspector 中的数据保护

AWS 分担责任模型分担责任模型适用于 Amazon Inspector 中的数据保护。如本模型所述 AWS ，负责保护运行所有内容的全球基础架构 AWS Cloud。您负责维护对托管在此基础设施上的内容的控制。您还负责您所使用的 AWS 服务 的安全配置和管理任务。有关数据隐私的更多信息，请参阅数据隐私常见问题解答。有关欧洲数据保护的信息，请参阅AWS 安全性博客上的 AWS 责任共担模式和 GDPR 博客文章。

出于数据保护目的，我们建议您保护 AWS 账户 凭证并使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 设置个人用户。这样，每个用户只获得履行其工作职责所需的权限。我们还建议您通过以下方式保护数据：

- 对每个账户使用多重身份验证 (MFA)。
- 使用 SSL/TLS 与资源通信。 AWS 我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 使用设置 API 和用户活动日志 AWS CloudTrail。
- 使用 AWS 加密解决方案以及其中的所有默认安全控件 AWS 服务。
- 使用高级托管安全服务（例如 Amazon Macie），它有助于发现和保护存储在 Amazon S3 中的敏感数据。
- 如果您在 AWS 通过命令行界面或 API 进行访问时需要经过 FIPS 140-2 验证的加密模块，请使用 FIPS 端点。有关可用的 FIPS 端点的更多信息，请参阅《美国联邦信息处理标准(FIPS) 第 140-2 版》。

我们强烈建议您切勿将机密信息或敏感信息（如您客户的电子邮件地址）放入标签或自由格式文本字段（如名称字段）。这包括当你 AWS 服务 使用控制台、API 或软件开发工具包与 Amazon In AWS spector 或其他人合作时。 AWS CLI在用于名称的标签或自由格式文本字段中输入的任何数据都可能会用于计费或诊断日志。如果您向外部服务器提供网址，强烈建议您不要在网址中包含凭证信息来验证对该服务器的请求。

主题

静态加密

默认情况下，Amazon Inspector 使用 AWS 加密解决方案安全地存储您的静态数据。Amazon Inspector 使用密钥管理服务 () 中 AWS 拥有的加密密钥对数据进行加密，例如使用 Sy AWS stems Manager 收集的资源清单、从 AWS Amazon ECR 映像中解析的资源清单以及生成的安全调查结果。AWS KMS您无法查看、管理或使用 AWS 自有密钥，也无法审核其使用情况。但是，无需采取任何措施或更改任何计划即可保护用于加密数据的密钥。有关更多信息，请参阅 AWS 自有密钥。

如果您禁用 Amazon Inspector，它将永久删除它为您存储或维护的所有资源，例如收集的清单和安全调查发现。

对调查发现中的代码进行静态加密

要扫描 Amazon Inspector Lambda 代码，Amazon Inspector CodeGuru 会合作扫描您的代码中是否存在漏洞。检测到漏洞时，会 CodeGuru 提取包含该漏洞的代码片段并存储该代码，直到 Amazon Inspector 请求访问为止。默认情况下， CodeGuru 使用 AWS 自有密钥对提取的代码进行加密，但是，您可以将 Amazon Inspector 配置为使用您自己的客户托管 AWS KMS 密钥进行加密。

以下工作流程说明了 Amazon Inspector 如何使用您配置的密钥来加密代码：

1. 您可以使用 Amazon Inspector UpdateEncryptionKeyAPI 向亚马逊 Inspector 提供 AWS KMS 密钥。
2. Amazon Inspector 会将有关您的 AWS KMS 密钥的信息转发给。 CodeGuru CodeGuru 存储信息以备将来使用。
3. CodeGuru 为您在 Amazon Ins AWS KMS pector 中配置的密钥申请授权。
4. CodeGuru 根据您的密钥创建加密的数据 AWS KMS 密钥并将其存储。此数据密钥用于加密您存储的代码数据 CodeGuru。
5. 每当 Amazon Inspector 通过代码扫描请求数据时，都会 CodeGuru 使用授权来解密加密的数据密钥，然后使用该密钥解密数据，以便可以检索数据。

禁用 Lambda 代码扫描后，授权将 CodeGuru 停用并删除关联的数据密钥。

使用客户托管密钥进行代码加密的权限

要使用加密，您需要制定允许访问 AWS KMS 操作的策略，以及授予 Amazon Inspector 和通过条件键使用这些操作的 CodeGuru 权限的声明。

如果要设置、更新或重置账户的加密密钥，则需要使用 Amazon Inspector 管理员策略，例如 AWS 托管策略：AmazonInspector2FullAccess。您还需要向只读用户授予以下权限，这些用户需要从调查发现中检索代码片段或与所选用于加密的密钥相关的数据。

对于 KMS，该策略必须允许执行以下操作：

- `kms:CreateGrant`
- `kms:Decrypt`
- `kms:DescribeKey`
- `kms:GenerateDataKeyWithoutPlainText`

- kms:Encrypt
- kms:RetireGrant

在确认您的策略中拥有正确的 AWS KMS 权限后，您必须附上一份声明，允许 Amazon Inspector 和 CodeGuru 使用您的密钥进行加密。附上以下策略语句：

> **Note**
>
> 将区域替换为您启用了 Amazon Inspector Lambda 代码扫描的 AWS 区域。

```
{
            "Sid": "allow CodeGuru Security to request a grant for a AWS KMS key",
          "Effect": "Allow",
          "Action": "kms:CreateGrant",
          "Resource": "*",
          "Condition": {
           "ForAllValues:StringEquals": {
            "kms:GrantOperations": [
             "GenerateDataKey",
             "GenerateDataKeyWithoutPlaintext",
             "Encrypt",
             "Decrypt",
             "RetireGrant",
             "DescribeKey"
            ]
           },
           "StringEquals": {
            "kms:ViaService": [
             "codeguru-security.Region.amazonaws.com"
            ]
           }
          }
         },
         {
          "Sid": "allow Amazon Inspector and CodeGuru Security to use your AWS KMS key",
          "Effect": "Allow",
          "Action": [
           "kms:Encrypt",
           "kms:Decrypt",
```

```
          "kms:RetireGrant",
          "kms:DescribeKey",
          "kms:GenerateDataKeyWithoutPlaintext"
         ],
         "Resource": "*",
         "Condition": {
          "StringEquals": {
           "kms:ViaService": [
            "inspector2.Region.amazonaws.com",
            "codeguru-security.Region.amazonaws.com"
           ]
          }
         }
        }
```

> **Note**
>
> 添加语句时，请确保语法有效。策略使用 JSON 格式。这意味着需要在语句之前或之后添加一个逗号，具体取决于在策略中添加语句的位置。如果将语句添加在最后，请在前一语句的右大括号后面添加一个逗号。如果将语句添加为第一个语句，或添加在两个现有语句之间，请在语句的右大括号后面添加一个逗号。

使用客户托管密钥配置加密

要使用客户托管密钥为您的账户配置加密，您必须是具有 使用客户托管密钥进行代码加密的权限 中列出的权限的 Amazon Inspector 管理员。此外，您还需要一个与您的发现位于同一 AWS 区域的 AWS KMS 密钥，或者一个多区域密钥。您可以使用账户中现有的对称密钥，也可以使用 AWS 管理控制台或 API 创建对称客户托管密钥。 AWS KMS 有关更多信息，请参阅 AWS KMS 用户指南中的创建对称加密 AWS KMS 密钥。

使用 Amazon Inspector API 配置加密

要设置加密密钥，请在以亚马逊 Inspector 管理员身份登录时UpdateEncryptionKey运行 Amazon Inspector API。在 API 请求中，使用`kmsKeyId`字段指定要使用的 AWS KMS 密钥的 ARN。对于`scanType`，请输入`CODE`，对于`resourceType`，请输入`AWS_LAMBDA_FUNCTION`。

你可以使用 UpdateEncryptionKeyAPI 来查看 Amazon Inspector 正在使用哪个 AWS KMS 密钥进行加密。

> **Note**
>
> 如果您在未设置客户托管密钥`GetEncryptionKey`的情况下尝试使用，则操作会返回`ResourceNotFoundException`错误，这意味着正在使用 AWS 自有密钥进行加密。

如果您删除密钥或更改其政策以拒绝访问 Amazon Inspector，否则 CodeGuru 您将无法访问您的代码漏洞发现，并且您的账户的 Lambda 代码扫描将失败。

您可以使用恢复使用 AWS 自有密钥`ResetEncryptionKey`对作为 Amazon Inspector 调查结果一部分提取的代码进行加密。

传输中加密

AWS 对 AWS 内部系统和其他 AWS 服务之间传输的所有数据进行加密。

为了收集清单，Systems Manager 从客户拥有的 EC2 实例收集遥测数据，然后通过受传输层安全 (TLS) 保护的通道将这些数据发送回以 AWS 进行评估。要了解 SSM 如何加密传输中数据，请参阅 Systems Manager 中的数据保护。

同样，发送到 Security Hub 的 Amazon ECR 和 Lamb AWS da 函数扫描结果也使用受 TLS 保护的通道进行加密。

适用于 Amazon Inspector 的 Identity and Access Management

AWS Identity and Access Management (IAM) AWS 服务 可帮助管理员安全地控制对 AWS 资源的访问权限。IAM 管理员控制谁可以通过身份验证（登录）和授权（具有权限）使用 Amazon Inspector 资源。您可以使用 IAM AWS 服务 ，无需支付额外费用。

主题

- 受众
- 使用身份进行身份验证
- 使用策略管理访问
- Amazon Inspector 如何与 IAM 配合使用
- Amazon Inspector 基于身份的策略示例
- AWS Amazon Inspector 的托管政策

- 对 Amazon Inspector 使用服务相关角色
- Amazon Inspector 身份和访问问题排查

受众

您的使用方式 AWS Identity and Access Management (IAM) 会有所不同，具体取决于您在 Amazon Inspector 中所做的工作。

服务用户 – 如果您使用 Amazon Inspector 服务来完成任务，则您的管理员会为您提供所需的凭证和权限。随着您使用更多 Amazon Inspector 功能来完成工作，您可能需要额外权限。了解如何管理访问权限有助于您向管理员请求适合的权限。如果您无法访问 Amazon Inspector 中的功能，请参阅Amazon Inspector 身份和访问问题排查。

服务管理员 – 如果您在公司负责管理 Amazon Inspector 资源，您可能对 Amazon Inspector 具有完全访问权限。您有责任确定您的服务用户应访问哪些 Amazon Inspector 功能和资源。然后，您必须向 IAM 管理员提交请求以更改服务用户的权限。请查看该页面上的信息以了解 IAM 的基本概念。要了解有关您的公司如何将 IAM 与 Amazon Inspector 搭配使用的更多信息，请参阅Amazon Inspector 如何与 IAM 配合使用。

IAM 管理员 – 如果您是 IAM 管理员，您可能需要了解如何编写策略以管理对 Amazon Inspector 的访问的详细信息。要查看您可在 IAM 中使用的 Amazon Inspector 基于身份的策略示例，请参阅Amazon Inspector 基于身份的策略示例。

使用身份进行身份验证

身份验证是您 AWS 使用身份凭证登录的方式。您必须以 IAM 用户身份或通过担 AWS 账户根用户任 IAM 角色进行身份验证（登录 AWS）。

您可以使用通过身份源提供的凭据以 AWS 联合身份登录。 AWS IAM Identity Center （IAM Identity Center）用户、贵公司的单点登录身份验证以及您的 Google 或 Facebook 凭据就是联合身份的示例。当您以联合身份登录时，管理员以前使用 IAM 角色设置了身份联合验证。当你使用联合访问 AWS 时，你就是在间接扮演一个角色。

根据您的用户类型，您可以登录 AWS Management Console 或 AWS 访问门户。有关登录的更多信息 AWS，请参阅《AWS 登录 用户指南》中的如何登录到您 AWS 账户的。

如果您 AWS 以编程方式访问，则会 AWS 提供软件开发套件 (SDK) 和命令行接口 (CLI)，以便使用您的凭据对请求进行加密签名。如果您不使用 AWS 工具，则必须自己签署请求。有关使用推荐的方法自行签署请求的更多信息，请参阅 IAM 用户指南中的签署 AWS API 请求。

无论使用何种身份验证方法，您可能需要提供其他安全信息。例如， AWS 建议您使用多重身份验证 (MFA) 来提高账户的安全性。要了解更多信息，请参阅《AWS IAM Identity Center 用户指南》中的多重身份验证 和《IAM 用户指南》中的在 AWS中使用多重身份验证（MFA）。

AWS 账户 root 用户

创建时 AWS 账户，首先要有一个登录身份，该身份可以完全访问账户中的所有资源 AWS 服务 和资源。此身份被称为 AWS 账户 root 用户，使用您创建账户时使用的电子邮件地址和密码登录即可访问该身份。强烈建议不要使用根用户执行日常任务。保护好根用户凭证，并使用这些凭证来执行仅根用户可以执行的任务。有关要求以根用户身份登录的任务的完整列表，请参阅《IAM 用户指南》中的需要根用户凭证的任务。

联合身份

作为最佳实践，要求人类用户（包括需要管理员访问权限的用户）使用与身份提供商的联合身份验证 AWS 服务 通过临时证书进行访问。

联合身份是指您的企业用户目录、Web 身份提供商、Identity C enter 目录中的用户，或者任何使用 AWS 服务 通过身份源提供的凭据进行访问的用户。 AWS Directory Service当联合身份访问时 AWS 账户，他们将扮演角色，角色提供临时证书。

要集中管理访问权限，我们建议您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中创建用户和群组，也可以连接并同步到您自己的身份源中的一组用户和群组，以便在您的所有 AWS 账户 和应用程序中使用。有关 IAM Identity Center 的信息，请参阅《AWS IAM Identity Center 用户指南》中的什么是 IAM Identity Center？

IAM 用户和群组

I AM 用户是您 AWS 账户 内部对个人或应用程序具有特定权限的身份。在可能的情况下，建议使用临时凭证，而不是创建具有长期凭证（如密码和访问密钥）的 IAM 用户。但是，如果有一些特定的使用场景需要长期凭证以及 IAM 用户，我们建议轮换访问密钥。有关更多信息，请参阅《IAM 用户指南》中的对于需要长期凭证的使用场景定期轮换访问密钥。

IAM 组是一个用于指定一组 IAM 用户的身份。您不能使用群组的身份登录。您可以使用组来一次性为多个用户指定权限。如果有大量用户，使用组可以更轻松地管理用户权限。例如，您可能具有一个名为 IAMAdmins 的组，并为该组授予权限以管理 IAM 资源。

用户与角色不同。用户唯一地与某个人员或应用程序关联，而角色旨在让需要它的任何人担任。用户具有永久的长期凭证，而角色提供临时凭证。要了解更多信息，请参阅《IAM 用户指南》中的何时创建 IAM 用户（而不是角色）。

IAM 角色

I AM 角色是您内部具有特定权限 AWS 账户 的身份。它类似于 IAM 用户，但与特定人员不关联。您可以 AWS Management Console 通过切换角色在中临时担任 IAM 角色。您可以通过调用 AWS CLI 或 AWS API 操作或使用自定义 URL 来代入角色。有关使用角色的方法的更多信息，请参阅《IAM 用户指南》中的使用 IAM 角色。

具有临时凭证的 IAM 角色在以下情况下很有用：

- 联合用户访问——要向联合身份分配权限，请创建角色并为角色定义权限。当联合身份进行身份验证时，该身份将与角色相关联并被授予由此角色定义的权限。有关联合身份验证的角色的信息，请参阅《IAM 用户指南》中的为第三方身份提供商创建角色。如果您使用 IAM Identity Center，则需要配置权限集。为控制身份在进行身份验证后可以访问的内容，IAM Identity Center 将权限集与 IAM 中的角色相关联。有关权限集的信息，请参阅《AWS IAM Identity Center 用户指南》中的权限集。
- 临时 IAM 用户权限——IAM 用户或角色可代入 IAM 角色，以暂时获得针对特定任务的不同权限。
- 跨账户访问——您可以使用 IAM 角色以允许不同账户中的某个人（可信主体）访问账户中的资源。角色是授予跨账户存取权限的主要方式。但是，对于某些资源 AWS 服务，您可以将策略直接附加到资源（而不是使用角色作为代理）。要了解用于跨账户访问的角色和基于资源的策略之间的差别，请参阅《IAM 用户指南》中的 IAM 角色与基于资源的策略有何不同。
- 跨服务访问 — 有些 AWS 服务 使用其他 AWS 服务服务中的功能。例如，当您在某个服务中进行调用时，该服务通常会在 Amazon EC2 中运行应用程序或在 Amazon S3 中存储对象。服务可能会使用发出调用的主体的权限、使用服务角色或使用服务相关角色来执行此操作。
  - 转发访问会话 (FAS) — 当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，此操作然后在不同服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务 向下游服务发出请求的请求。 AWS 服务只有当服务收到需要与其他 AWS 服务 或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两个操作的权限。有关发出 FAS 请求时的策略详情，请参阅转发访问会话。
  - 服务角色 - 服务角色是服务代表您在您的账户中执行操作而分派的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。
  - 服务相关角色-服务相关角色是一种链接到的服务角色。 AWS 服务服务可以担任代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。
- 在 A@@ mazon EC2 上运行的应用程序 — 您可以使用 IAM 角色管理在 EC2 实例上运行并发出 AWS CLI 或 AWS API 请求的应用程序的临时证书。这优先于在 EC2 实例中存储访问密钥。要向 EC2 实例分配 AWS 角色并使其可供其所有应用程序使用，您需要创建附加到该实例的实例配置文

件。实例配置文件包含角色，并使 EC2 实例上运行的程序能够获得临时凭证。有关更多信息，请参阅《IAM 用户指南》中的使用 IAM 角色为 Amazon EC2 实例上运行的应用程序授予权限。

要了解是使用 IAM 角色还是 IAM 用户，请参阅《IAM 用户指南》中的何时创建 IAM 角色（而不是用户）。

使用策略管理访问

您可以 AWS 通过创建策略并将其附加到 AWS 身份或资源来控制中的访问权限。策略是其中的一个对象 AWS ，当与身份或资源关联时，它会定义其权限。 AWS 在委托人（用户、root 用户或角色会话）发出请求时评估这些策略。策略中的权限确定是允许还是拒绝请求。大多数策略都以 JSON 文档的 AWS 形式存储在中。有关 JSON 策略文档的结构和内容的更多信息，请参阅《IAM 用户指南》中的 JSON 策略概述。

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

默认情况下，用户和角色没有权限。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM policy。然后，管理员可以向角色添加 IAM policy，并且用户可以代入角色。

IAM policy 定义操作的权限，无关乎您使用哪种方法执行操作。例如，假设有一个允许 `iam:GetRole` 操作的策略。拥有该策略的用户可以从 AWS Management Console AWS CLI、或 AWS API 获取角色信息。

基于身份的策略

基于身份的策略是可附加到身份（如 IAM 用户、用户群组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的 创建 IAM policy。

基于身份的策略可以进一步归类为内联策略或托管式策略。内联策略直接嵌入单个用户、群组或角色中。托管策略是独立的策略，您可以将其附加到中的多个用户、群组和角色 AWS 账户。托管策略包括 AWS 托管策略和客户托管策略。要了解如何在托管式策略和内联策略之间进行选择，请参阅《IAM 用户指南》中的在托管式策略与内联策略之间进行选择。

基于资源的策略

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资

源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中指定主体。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

基于资源的策略是位于该服务中的内联策略。您不能在基于资源的策略中使用 IAM 中的 AWS 托管策略。

访问控制列表 (ACL)

访问控制列表(ACL) 控制哪些主体（账户成员、用户或角色）有权访问资源。ACL 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

Amazon S3 和 Amazon VPC 就是支持 ACL 的服务示例。 AWS WAF要了解有关 ACL 的更多信息，请参阅《Amazon Simple Storage Service 开发人员指南》中的访问控制列表 (ACL) 概述。

其他策略类型

AWS 支持其他不太常见的策略类型。这些策略类型可以设置更常用的策略类型授予的最大权限。

- 权限边界——权限边界是一个高级特征，用于设置基于身份的策略可以为 IAM 实体（IAM 用户或角色）授予的最大权限。您可以为实体设置权限边界。这些结果权限是实体基于身份的策略及其权限边界的交集。在 `Principal` 中指定用户或角色的基于资源的策略不受权限边界限制。任一项策略中的显式拒绝将覆盖允许。有关权限边界的更多信息，请参阅《IAM 用户指南》中的 IAM 实体的权限边界。
- 服务控制策略 (SCP)-SCP 是 JSON 策略，用于指定组织或组织单位 (OU) 的最大权限。 AWS Organizations AWS Organizations 是一项用于对您的企业拥有的多 AWS 账户 项进行分组和集中管理的服务。如果您在组织内启用了特征，则可对任意或全部账户应用服务控制策略 (SCP)。SCP 限制成员账户中的实体（包括每个 AWS 账户根用户实体）的权限。有关组织和 SCP 的更多信息，请参阅《AWS Organizations 用户指南》中的 SCP 的工作原理。
- 会话策略——会话策略是当以编程方式为角色或联合用户创建临时会话时作为参数传递的高级策略。结果会话的权限是用户或角色的基于身份的策略和会话策略的交集。权限也可以来自基于资源的策略。任一项策略中的显式拒绝将覆盖允许。有关更多信息，请参阅《IAM 用户指南》中的会话策略。

多个策略类型

当多个类型的策略应用于一个请求时，生成的权限更加复杂和难以理解。要了解在涉及多种策略类型时如何 AWS 确定是否允许请求，请参阅 IAM 用户指南中的策略评估逻辑。

Amazon Inspector 如何与 IAM 配合使用

在使用 IAM 管理对 Amazon Inspector 的访问权限之前，您应该了解哪些 IAM 功能可用于 Amazon Inspector。

可与 Amazon Inspector 结合使用的 IAM 功能

IAM 功能	Amazon Inspector 支持
基于身份的策略	是
基于资源的策略	否
策略操作	是
策略资源	支持
策略条件键（特定于服务）	支持
ACL	否
ABAC（策略中的标签）	部分
临时凭证	是
主体权限	支持
服务角色	否
服务相关角色	支持

要全面了解 Amazon Inspector 和其他 AWS 服务 功能如何使用大多数 IAM 功能 AWS 服务 ，请在 IAM 用户指南中查看与 IAM 配合使用的方法。

Amazon Inspector 基于身份的策略

支持基于身份的策略	是

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的创建 IAM 策略。

通过使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作和资源以及允许或拒绝操作的条件。您无法在基于身份的策略中指定主体，因为它适用于其附加的用户或角色。要了解可在 JSON 策略中使用的所有元素，请参阅《IAM 用户指南》中的 IAM JSON 策略元素引用。

Amazon Inspector 基于身份的策略示例

要查看 Amazon Inspector 基于身份的策略的示例，请参阅Amazon Inspector 基于身份的策略示例。

Amazon Inspector 基于资源的策略

支持基于资源的策略	否

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中指定主体。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

要启用跨账户存取，您可以将整个账户或其他账户中的 IAM 实体指定为基于资源的策略中的主体。将跨账户主体添加到基于资源的策略只是建立信任关系工作的一半而已。当委托人和资源处于不同位置时 AWS 账户，可信账户中的 IAM 管理员还必须向委托人实体（用户或角色）授予访问资源的权限。他们通过将基于身份的策略附加到实体以授予权限。但是，如果基于资源的策略向同一个账户中的主体授予访问权限，则不需要额外的基于身份的策略。有关更多信息，请参阅《IAM 用户指南》中的 IAM 角色与基于资源的策略有何不同。

Amazon Inspector 的策略操作

支持策略操作	支持

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

JSON 策略的 `Action` 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 AWS API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行关联操作的权限。

有关 Amazon Inspector 操作的列表，请参阅《服务授权参考》中的 Amazon Inspector 定义的操作。

Amazon Inspector 中的策略操作在操作前使用以下前缀：

```
inspector2
```

要在单个语句中指定多项操作，请使用逗号将它们隔开。

```
"Action": [
      "inspector2:action1",
      "inspector2:action2"
         ]
```

要查看 Amazon Inspector 基于身份的策略的示例，请参阅Amazon Inspector 基于身份的策略示例。

Amazon Inspector 的策略资源

支持策略资源	支持

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

`Resource` JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 `Resource` 或 `NotResource` 元素。作为最佳实操，请使用其 Amazon 资源名称 (ARN) 指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符 (*) 指示语句应用于所有资源。

```
"Resource": "*"
```

要查看 Amazon Inspector 的资源类型及其 ARN 的列表，请参阅《服务授权参考》中的 Amazon Inspector 定义的资源。要了解您可以在哪些操作中指定每个资源的 ARN，请参阅 Amazon Inspector 定义的操作。

要查看 Amazon Inspector 基于身份的策略的示例，请参阅Amazon Inspector 基于身份的策略示例。

Amazon Inspector 的策略条件键

支持特定于服务的策略条件键	支持

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 Condition 元素（或 Condition 块）中，您可以指定语句生效的条件。Condition 元素是可选的。您可以创建使用条件运算符（例如，等于或小于）的条件表达式，以使策略中的条件与请求中的值相匹配。

如果在一个语句中指定多个 Condition 元素，或在单个 Condition 元素中指定多个密钥，则 AWS 使用逻辑 AND 运算评估它们。如果您为单个条件键指定多个值，则使用逻辑OR运算来 AWS 评估条件。在授予语句的权限之前必须满足所有的条件。

您也可以在指定条件时使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅《IAM 用户指南》中的 IAM 策略元素：变量和标签。

AWS 支持全局条件密钥和特定于服务的条件密钥。要查看所有 AWS 全局条件键，请参阅 IAM 用户指南中的AWS 全局条件上下文密钥。

要查看 Amazon Inspector 条件键的列表，请参阅《服务授权参考》中的 Amazon Inspector 的条件键。要了解您可以对哪些操作和资源使用条件键，请参阅 Amazon Inspector 定义的操作。

要查看 Amazon Inspector 基于身份的策略的示例，请参阅Amazon Inspector 基于身份的策略示例。

Amazon Inspector 中的 ACL

支持 ACL	否

访问控制列表(ACL) 控制哪些主体（账户成员、用户或角色）有权访问资源。ACL 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

ABAC 与 Amazon Inspector

支持 ABAC（策略中的标签）	部分

基于属性的访问权限控制 (ABAC) 是一种授权策略，该策略基于属性来定义权限。在中 AWS，这些属性称为标签。您可以将标签附加到 IAM 实体（用户或角色）和许多 AWS 资源。标记实体和资源是 ABAC 的第一步。然后设计 ABAC 策略，以在主体的标签与他们尝试访问的资源标签匹配时允许操作。

ABAC 在快速增长的环境中非常有用，并在策略管理变得繁琐的情况下可以提供帮助。

要基于标签控制访问，需要使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 条件密钥在策略的条件元素中提供标签信息。

如果某个服务对于每种资源类型都支持所有这三个条件密钥，则对于该服务，该值为是。如果某个服务仅对于部分资源类型支持所有这三个条件密钥，则该值为部分。

有关 ABAC 的更多信息，请参阅《IAM 用户指南》中的什么是 ABAC？。要查看设置 ABAC 步骤的教程，请参阅《IAM 用户指南》中的使用基于属性的访问权限控制（ABAC）。

将临时凭证用于 Amazon Inspector

支持临时凭证	支持

当你使用临时证书登录时，有些 AWS 服务 不起作用。有关更多信息，包括哪些 AWS 服务 适用于临时证书，请参阅 IAM 用户指南中的AWS 服务 与 IAM 配合使用的信息。

如果您使用除用户名和密码之外的任何方法登录，则 AWS Management Console 使用的是临时证书。例如，当您 AWS 使用公司的单点登录 (SSO) 链接进行访问时，该过程会自动创建临时证书。当您以用户身份登录控制台，然后切换角色时，还会自动创建临时凭证。有关切换角色的更多信息，请参阅《IAM 用户指南》中的切换到角色（控制台）。

您可以使用 AWS CLI 或 AWS API 手动创建临时证书。然后，您可以使用这些临时证书进行访问 AWS。 AWS 建议您动态生成临时证书，而不是使用长期访问密钥。有关更多信息，请参阅 IAM 中的临时安全凭证。

Amazon Inspector 的跨服务主体权限

支持转发访问会话 (FAS)	支持

当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，此操作然后在不同服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务 向下游服务发出请求的请求。 AWS 服务只有当服务收到需要与其他 AWS 服务 或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两个操作的权限。有关发出 FAS 请求时的策略详情，请参阅转发访问会话。

Amazon Inspector 的服务角色

支持服务角色	否

服务角色是由一项服务代入、代表您执行操作的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。

> **Warning**
>
> 更改服务角色的权限可能会破坏 Amazon Inspector 的功能。仅当 Amazon Inspector 提供相关指导时才编辑服务角色。

Amazon Inspector 的服务相关角色

支持服务相关角色	支持

服务相关角色是一种与服务相关联的 AWS 服务服务角色。服务可以担任代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

有关创建或管理服务相关角色的详细信息，请参阅使用 IAM 的AWS 服务。在表中查找服务相关角色列中包含 Yes 的服务。选择是链接以查看该服务的服务相关角色文档。

Amazon Inspector 基于身份的策略示例

默认情况下，用户和角色没有创建或修改 Amazon Inspector 资源的权限。他们也无法使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或 AWS API 执行任务。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM policy。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

要了解如何使用这些示例 JSON 策略文档创建基于 IAM 身份的策略，请参阅《IAM 用户指南》中的创建 IAM policy。

有关 Amazon Inspector 定义的操作和资源类型的详细信息，包括每种资源类型的 ARN 的格式，请参阅《服务授权参考》中的 Amazon Inspector 的操作、资源和条件键。

主题

策略最佳实践

基于身份的策略确定某个人是否可以创建、访问或删除您账户中的 Amazon Inspector 资源。这些操作可能会使 AWS 账户产生成本。创建或编辑基于身份的策略时，请遵循以下准则和建议：

- 开始使用 AWS 托管策略并转向最低权限权限 — 要开始向用户和工作负载授予权限，请使用为许多常见用例授予权限的AWS 托管策略。它们在你的版本中可用 AWS 账户。我们建议您通过定义针对您的用例的 AWS 客户托管策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的AWS 托管式策略或工作职能的AWS 托管式策略。
- 应用最低权限 – 在使用 IAM policy 设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 IAM 中的策略和权限。
- 使用 IAM policy 中的条件进一步限制访问权限 – 您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果服务操作是通过特定的方式使用的，则也可以使用条件来授予对服务操作的访问权限 AWS 服务，例如 AWS CloudFormation。有关更多信息，请参阅《IAM 用户指南》中的 IAM JSON 策略元素：条件。

- 使用 IAM Access Analyzer 验证 IAM policy，确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，确保策略符合 IAM policy 语言（JSON）和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，有助于制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的 IAM Access Analyzer 策略验证。
- 需要多重身份验证 (MFA)-如果 AWS 账户您的场景需要 IAM 用户或根用户，请启用 MFA 以提高安全性。要在调用 API 操作时需要 MFA，请将 MFA 条件添加到策略中。有关更多信息，请参阅《IAM 用户指南》https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_mfa_configure-api-require.html中的配置受 MFA 保护的 API 访问。

有关 IAM 中的最佳实践的更多信息，请参阅《IAM 用户指南》中的 IAM 中的安全最佳实践。

使用 Amazon Inspector 控制台

要访问 Amazon Inspector 控制台，您必须具有一组最低的权限。这些权限必须允许您列出和查看有关您的 AWS 账户中的 Amazon Inspector 资源的详细信息。如果创建比必需的最低权限更为严格的基于身份的策略，对于附加了该策略的实体（用户或角色），控制台将无法按预期正常运行。

对于仅调用 AWS CLI 或 AWS API 的用户，您无需为其设置最低控制台权限。相反，只允许访问与其尝试执行的 API 操作相匹配的操作。

为确保用户和角色仍然可以使用 Amazon Inspector 控制台，还需要将亚马逊检查器*ConsoleAccess*或*ReadOnly* AWS 托管策略附加到这些实体。有关更多信息，请参阅《IAM 用户指南》中的为用户添加权限。

允许用户查看他们自己的权限

该示例说明了如何创建策略，以允许 IAM 用户查看附加到其用户身份的内联和托管式策略。此策略包括在控制台上或使用 AWS CLI 或 AWS API 以编程方式完成此操作的权限。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
```

```
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

允许以只读方式访问所有 Amazon Inspector 资源

以下示例展示了一个策略，该策略允许以只读方式访问所有 Amazon Inspector 资源。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "inspector2:Describe*",
                "inspector2:Get*",
                "inspector2:BatchGet*",
                "inspector2:List*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "organizations:ListDelegatedAdministrators",
```

```
                "organizations:ListAWSServiceAccessForOrganization",
                "organizations:DescribeOrganizationalUnit",
                "organizations:DescribeAccount",
                "organizations:DescribeOrganization"
            ],
            "Resource": "*"
        }
    ]
}
```

允许完全访问所有 Amazon Inspector 资源

以下示例展示了一个策略，该策略允许完全访问所有 Amazon Inspector 资源。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "inspector2:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "iam:AWSServiceName": "inspector2.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "organizations:EnableAWSServiceAccess",
                "organizations:RegisterDelegatedAdministrator",
                "organizations:ListDelegatedAdministrators",
                "organizations:ListAWSServiceAccessForOrganization",
                "organizations:DescribeOrganizationalUnit",
                "organizations:DescribeAccount",
                "organizations:DescribeOrganization"
            ],
```

```
            "Resource": "*"
        }
    ]
}
```

AWS Amazon Inspector 的托管政策

AWS 托管策略是由创建和管理的独立策略 AWS。 AWS 托管策略旨在为许多常见用例提供权限，以便您可以开始为用户、组和角色分配权限。

请记住， AWS 托管策略可能不会为您的特定用例授予最低权限权限，因为它们可供所有 AWS 客户使用。我们建议通过定义特定于使用场景的客户管理型策略来进一步减少权限。

您无法更改 AWS 托管策略中定义的权限。如果 AWS 更新 AWS 托管策略中定义的权限，则更新会影响该策略所关联的所有委托人身份（用户、组和角色）。 AWS 最有可能在启动新的 API 或现有服务可以使用新 AWS 服务 的 API 操作时更新 AWS 托管策略。

有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管式策略。

AWS 托管策略：AmazonInspector2FullAccess

您可以将 `AmazonInspector2FullAccess` 策略附加到 IAM 身份。

此策略授予允许完全访问 Amazon Inspector 的权限。

权限详细信息

该策略包含以下权限。

- `inspector2` – 允许完全访问 Amazon Inspector 功能。

- `iam` – 允许 Amazon Inspector 创建服务相关角色，即 `AmazonInspector2AgentlessServiceRole`。这是必要的，这样 Amazon Inspector 才能执行一些操作，如检索有关 Amazon EC2 实例、Amazon ECR 存储库和容器映像的信息、分析 VPC 网络以及描述与贵组织关联的账户。有关更多信息，请参阅对 Amazon Inspector 使用服务相关角色。
- `organizations` — 允许管理员将 Amazon Inspector 用于 AWS Organizations中的组织。在中激活 Amazon Inspector 的可信访问权限后 AWS Organizations，委派管理员账户的成员可以管理其组织中的设置并查看调查结果。
- `codeguru-security`— 允许管理员使用 Amazon Inspector 检索信息代码片段并更改 CodeGuru 安全部门存储的代码的加密设置。有关更多信息，请参阅对调查发现中的代码进行静态加密。

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": "inspector2:*",
   "Resource": "*"
  },
  {
   "Effect": "Allow",
   "Action": [
    "codeguru-security:BatchGetFindings",
    "codeguru-security:GetAccountConfiguration",
    "codeguru-security:UpdateAccountConfiguration"
   ],
   "Resource": "*"
  },
  {
   "Effect": "Allow",
   "Action": "iam:CreateServiceLinkedRole",
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "iam:AWSServiceName": "inspector2.amazonaws.com"
    }
   }
  },
  {
   "Effect": "Allow",
   "Action": [
```

```
    "organizations:EnableAWSServiceAccess",
    "organizations:RegisterDelegatedAdministrator",
    "organizations:ListDelegatedAdministrators",
    "organizations:ListAWSServiceAccessForOrganization",
    "organizations:DescribeOrganizationalUnit",
    "organizations:DescribeAccount",
    "organizations:DescribeOrganization"
   ],
   "Resource": "*"
  }
 ]
}
```

AWS 托管策略：AmazonInspector2ReadOnlyAccess

您可以将 `AmazonInspector2ReadOnlyAccess` 策略附加到 IAM 身份。

此策略授予允许对 Amazon Inspector 进行只读访问的权限。

权限详细信息

该策略包含以下权限。

- `inspector2` – 允许以只读方式访问 Amazon Inspector 功能。
- `organizations`— 允许查看有关组织的 Amazon Inspector 覆盖范围 AWS Organizations 的详细信息。
- `codeguru-security`— 允许从 “ CodeGuru 安全” 中检索代码片段。还允许查看存储在 Sec CodeGuru urity 中的代码的加密设置。

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "organizations:ListDelegatedAdministrators",
    "organizations:ListAWSServiceAccessForOrganization",
```

```
    "organizations:DescribeOrganizationalUnit",
    "organizations:DescribeAccount",
    "organizations:DescribeOrganization",
    "inspector2:BatchGet*",
    "inspector2:List*",
    "inspector2:Describe*",
    "inspector2:Get*",
    "inspector2:Search*",
    "codeguru-security:BatchGetFindings",
    "codeguru-security:GetAccountConfiguration"
   ],
   "Resource": "*"
  }
 ]
}
```

AWS 托管策略：AmazonInspector2ManagedCisPolicy

可以将 AmazonInspector2ManagedCisPolicy 策略附加到您的 IAM 实体。此策略应附加到一个角色，该角色授予您的 Amazon EC2 实例运行该实例的 CIS 扫描的权限。您可以使用 IAM 角色管理在 EC2 实例上运行并发出 AWS CLI 或 AWS API 请求的应用程序的临时证书。这优先于在 EC2 实例中存储访问密钥。要向 EC2 实例分配 AWS 角色并使其可供其所有应用程序使用，您需要创建附加到该实例的实例配置文件。实例配置文件包含角色，并使 EC2 实例上运行的程序能够获得临时凭证。有关更多信息，请参阅《IAM 用户指南》中的使用 IAM 角色为 Amazon EC2 实例上运行的应用程序授予权限。

权限详细信息

该策略包含以下权限。

- inspector2— 允许访问用于运行 CIS 扫描的操作。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
            "inspector2:StartCisSession",
            "inspector2:StopCisSession",
            "inspector2:SendCisSessionTelemetry",
```

```
            "inspector2:SendCisSessionHealth"
            ],
            "Resource": "*",
        }
    ]
}
```

AWS 托管策略：AmazonInspector2ServiceRolePolicy

无法将 AmazonInspector2ServiceRolePolicy 策略附加到 IAM 实体。将此策略附加到允许 Amazon Inspector 代表您执行操作的服务相关角色。有关更多信息，请参阅对 Amazon Inspector 使用服务相关角色。

AWS 托管策略：AmazonInspector2AgentlessServiceRolePolicy

无法将 AmazonInspector2AgentlessServiceRolePolicy 策略附加到 IAM 实体。将此策略附加到允许 Amazon Inspector 代表您执行操作的服务相关角色。有关更多信息，请参阅对 Amazon Inspector 使用服务相关角色。

Amazon Inspector 更新 AWS 了托管政策

查看自该服务开始跟踪这些更改以来对 Amazon Inspector AWS 托管政策的更新的详细信息。要获得有关此页面更改的自动提醒，请订阅 Amazon Inspector 文档历史记录页面上的 RSS 源。

更改	描述	日期
AmazonInspector2 ManagedCisPolicy — 新政策	Amazon Inspector 添加了一个新的托管策略，您可以将其用作实例配置文件的一部分，以允许对实例进行 CIS 扫描。	2024 年 1 月 23 日
AmazonInspector2 ServiceRolePolicy — 对现有政策的更新	Amazon Inspector 添加了新的权限，允许亚马逊 Inspector 在目标实例上启动 CIS 扫描。	2024 年 1 月 23 日
AmazonInspector2 AgentlessServiceRolePolicy — 新政策	Amazon Inspector 添加了一项新的服务相关角色策略，以允	2023 年 11 月 27 日

更改	描述	日期
	许对 EC2 实例进行无代理扫描。	
AmazonInspector2ReadOnlyAccess — 对现有政策的更新	Amazon Inspector 增加了新的权限，允许只读用户检索程序包脆弱性调查发现的脆弱性情报详细信息。	2023 年 9 月 22 日
AmazonInspector2 ServiceRolePolicy — 对现有政策的更新	Amazon Inspector 增加了新的权限，允许 Amazon Inspector 扫描属于 Elastic Load Balancing 目标组的 Amazon EC2 实例的网络配置。	2023 年 8 月 31 日
AmazonInspector2ReadOnlyAccess — 对现有政策的更新	Amazon Inspector 增加了新的权限，允许只读用户为其资源导出软件材料清单 (SBOM)。	2023 年 6 月 29 日
AmazonInspector2ReadOnlyAccess — 对现有政策的更新	Amazon Inspector 增加了新的权限，允许只读用户检索其账户的 Lambda 代码扫描结果的加密设置详情。	2023 年 6 月 13 日
AmazonInspector2 FullAccess — 对现有政策的更新	Amazon Inspector 增加了新的权限，允许用户配置客户托管的 KMS 密钥，来加密 Lambda 代码扫描结果中的代码。	2023 年 6 月 13 日
AmazonInspector2ReadOnlyAccess — 对现有政策的更新	Amazon Inspector 增加了新的权限，允许只读用户检索其账户的 Lambda 代码扫描状态和结果的详细信息。	2023 年 5 月 2 日

更改	描述	日期
AmazonInspector2 ServiceRolePolicy — 对现有政策的更新	Amazon Inspector 添加了新的权限，允许亚马逊检查员在您激活 Lambda 扫描时在您的账户中创建 AWS CloudTrail 与服务相关的渠道。这样，Amazon Inspector 就可以监控您账户中的 CloudTrail 事件。	2023 年 4 月 30 日
AmazonInspector2 FullAccess — 对现有政策的更新	Amazon Inspector 增加了新的权限，允许用户检索 Lambda 代码扫描脆弱性调查发现的详细信息。	2023 年 4 月 21 日
AmazonInspector2 ServiceRolePolicy — 对现有政策的更新	Amazon Inspector 增加了新的权限，允许亚马逊 Inspector 向亚马逊 EC2 Systems Manager 发送有关客户为亚马逊 EC2 深度检查定义的自定义路径的信息。	2023 年 4 月 17 日
AmazonInspector2 ServiceRolePolicy — 对现有政策的更新	Amazon Inspector 添加了新的权限，允许亚马逊检查员在您激活 Lambda 扫描时在您的账户中创建 AWS CloudTrail 与服务相关的渠道。这样，Amazon Inspector 就可以监控您账户中的 CloudTrail 事件。	2023 年 4 月 30 日

更改	描述	日期
AmazonInspector2 ServiceRolePolicy — 对现有政策的更新	Amazon Inspector 增加了新的权限，允许亚马逊 Inspector 请求扫描 AWS Lambda 函数中的开发者代码，并从亚马逊 CodeGuru 安全部门接收扫描数据。此外，Amazon Inspector 还增加了审查 IAM policy 的权限。Amazon Inspector 使用这些信息扫描 Lambda 函数中是否存在代码脆弱性。	2023 年 2 月 28 日
AmazonInspector2 ServiceRolePolicy — 对现有政策的更新	Amazon Inspector 添加了一条新语句，允许 Amazon Inspector 检索 CloudWatch 有关上次调用 AWS Lambda 函数的时间的信息。Amazon Inspector 使用这些信息将扫描重点放在您环境中过去 90 天内处于活动状态的 Lambda 函数上。	2023 年 2 月 20 日
AmazonInspector2 ServiceRolePolicy — 对现有政策的更新	Amazon Inspector 添加了一个新声明，允许亚马逊 Inspector 检索有关 AWS Lambda 函数的信息，包括与每个函数关联的每个层版本。Amazon Inspector 使用这些信息扫描 Lambda 函数中是否存在安全脆弱性。	2022 年 11 月 28 日

更改	描述	日期
AmazonInspector2 ServiceRolePolicy — 对现有政策的更新	Amazon Inspector 增加了一项新操作，允许 Amazon Inspector 描述 SSM 关联的执行情况。此外，Amazon Inspector 还增加了额外的资源范围，允许 Amazon Inspector 使用 `AmazonInspector2` 拥有的 SSM 文档创建、更新、删除和启动 SSM 关联。	2022 年 8 月 31 日
AmazonInspector2 对现有政策的ServiceRolePolicy更新	Amazon Inspector 更新了政策的资源范围，允许亚马逊 Inspector 收集其他 AWS 分区中的软件库存。	2022 年 8 月 12 日
AmazonInspector2 ServiceRolePolicy — 对现有政策的更新	Amazon Inspector 重组了操作的资源范围，允许 Amazon Inspector 创建、删除和更新 SSM 关联。	2022 年 8 月 10 日
AmazonInspector2 ReadOnlyAccess — 新政策	Amazon Inspector 增加了一项新策略，允许以只读方式访问 Amazon Inspector 功能。	2022 年 1 月 21 日
AmazonInspector2 FullAccess — 新政策	Amazon Inspector 增加了一项新策略，允许完全访问 Amazon Inspector 功能。	2021 年 11 月 29 日
AmazonInspector2 ServiceRolePolicy — 新政策	Amazon Inspector 增加了一项新策略，允许 Amazon Inspector 代表您在其他服务中执行操作。	2021 年 11 月 29 日
Amazon Inspector 开始跟踪更改	Amazon Inspector 开始跟踪其 AWS 托管政策的变更。	2021 年 11 月 29 日

对 Amazon Inspector 使用服务相关角色

Amazon Inspector 使用名AWSServiceRoleForAmazonInspector2为的 AWS Identity and Access Management (IAM) 服务相关角色。此服务相关角色是与 Amazon Inspector 直接相关的 IAM 角色。它由 Amazon Inspector 预定义，它包括亚马逊检查员 AWS 服务 代表您致电他人所需的所有权限。

服务相关角色可让您更轻松地设置 Amazon Inspector，因为您不必手动添加必要的权限。Amazon Inspector 定义其服务相关角色的权限，除非另外定义，否则只有 Amazon Inspector 可以代入该角色。定义的权限包括信任策略和权限策略，而且权限策略不能附加到任何其它 IAM 实体。

必须配置权限，允许 IAM 实体（如组或角色）创建、编辑或删除服务相关角色。有关更多信息，请参阅《IAM 用户指南》中的服务相关角色权限。只有在首先删除服务相关角色的相关资源后，才能删除该角色。这将保护您的 Amazon Inspector 资源，因为您不会无意中删除对资源的访问权限。

有关支持服务相关角色的其他服务的信息，请参阅与 IAM 配合使用的AWS 服务，并查找服务相关角色列中显示为是的服务。选择带有链接的是可以查看该服务的服务相关角色文档。

Amazon Inspector 的服务相关角色权限

Amazon Inspector 使用名为 AWSServiceRoleForAmazonInspector2 的服务相关角色。该服务相关角色信任 inspector2.amazonaws.com 服务担任该角色。

该角色的权限策略名为 AmazonInspector2ServiceRolePolicy，允许 Amazon Inspector 执行以下任务：

- 使用 Amazon Elastic Compute Cloud (Amazon EC2) 操作检索有关实例和网络路径的信息。
- 使用 AWS Systems Manager 操作从您的 Amazon EC2 实例中检索库存，并从自定义路径中检索有关第三方包裹的信息。
- 使用 AWS Systems Manager SendCommand操作调用目标实例的 CIS 扫描。
- 使用 Amazon Elastic Container Registry 操作检索有关您的容器映像的信息。
- 使用 AWS Lambda 操作来检索有关您的 Lambda 函数的信息。
- 使用 AWS Organizations 操作来描述关联的账户。
- 使用 CloudWatch 操作来检索有关上次调用 Lambda 函数的时间的信息。
- 使用“选择 IAM”操作检索可能在您的 Lambda 代码中造成安全脆弱性的 IAM policy 的相关信息。
- 使用 CodeGuru 安全操作对 Lambda 函数中的代码执行扫描。Amazon Inspector 使用以下 CodeGuru 安全操作：
  - codeguru-security: CreateScan — 授予创建安全扫描的权限。 CodeGuru
  - codeguru-security：GetScan — 授予检索 CodeGuru 安全扫描元数据的权限。

- codeguru-security：ListFindings — 授予检索安全部门生成的发现结果的权限。 CodeGuru
- codeguru-security：DeleteScansByCategory — 授予安全部门删除由 Amazon Insp CodeGuru ector 启动的扫描的权限。
- codeguru-security：BatchGetFindings — 授予检索 Security 生成的一批特定发现的权限。 CodeGuru
- 使用“选择 Elastic Load Balancing”操作对属于 Elastic Load Balancing 目标组的 EC2 实例执行网络扫描。

该角色使用以下权限策略进行配置：

```
    {
"Version": "2012-10-17",
"Statement": [
 {
  "Sid": "TirosPolicy",
  "Effect": "Allow",
  "Action": [
   "directconnect:DescribeConnections",
   "directconnect:DescribeDirectConnectGatewayAssociations",
   "directconnect:DescribeDirectConnectGatewayAttachments",
   "directconnect:DescribeDirectConnectGateways",
   "directconnect:DescribeVirtualGateways",
   "directconnect:DescribeVirtualInterfaces",
   "ec2:DescribeAvailabilityZones",
   "ec2:DescribeCustomerGateways",
   "ec2:DescribeInstances",
   "ec2:DescribeInternetGateways",
   "ec2:DescribeManagedPrefixLists",
   "ec2:DescribeNatGateways",
   "ec2:DescribeNetworkAcls",
   "ec2:DescribeNetworkInterfaces",
   "ec2:DescribePrefixLists",
   "ec2:DescribeRegions",
   "ec2:DescribeRouteTables",
   "ec2:DescribeSecurityGroups",
   "ec2:DescribeSubnets",
   "ec2:DescribeTransitGatewayAttachments",
   "ec2:DescribeTransitGatewayConnects",
   "ec2:DescribeTransitGatewayPeeringAttachments",
   "ec2:DescribeTransitGatewayRouteTables",
```

```
    "ec2:DescribeTransitGatewayVpcAttachments",
    "ec2:DescribeTransitGateways",
    "ec2:DescribeVpcEndpointServiceConfigurations",
    "ec2:DescribeVpcEndpoints",
    "ec2:DescribeVpcPeeringConnections",
    "ec2:DescribeVpcs",
    "ec2:DescribeVpnConnections",
    "ec2:DescribeVpnGateways",
    "ec2:GetManagedPrefixListEntries",
    "ec2:GetTransitGatewayRouteTablePropagations",
    "ec2:SearchTransitGatewayRoutes",
    "elasticloadbalancing:DescribeListeners",
    "elasticloadbalancing:DescribeLoadBalancerAttributes",
    "elasticloadbalancing:DescribeLoadBalancers",
    "elasticloadbalancing:DescribeRules",
    "elasticloadbalancing:DescribeTags",
    "elasticloadbalancing:DescribeTargetGroups",
    "elasticloadbalancing:DescribeTargetGroupAttributes",
    "elasticloadbalancing:DescribeTargetHealth",
    "network-firewall:DescribeFirewall",
    "network-firewall:DescribeFirewallPolicy",
    "network-firewall:DescribeResourcePolicy",
    "network-firewall:DescribeRuleGroup",
    "network-firewall:ListFirewallPolicies",
    "network-firewall:ListFirewalls",
    "network-firewall:ListRuleGroups",
    "tiros:CreateQuery",
    "tiros:GetQueryAnswer"
   ],
   "Resource": [
    "*"
   ]
  },
  {
   "Sid": "PackageVulnerabilityScanning",
   "Effect": "Allow",
   "Action": [
    "ecr:BatchGetImage",
    "ecr:BatchGetRepositoryScanningConfiguration",
    "ecr:DescribeImages",
    "ecr:DescribeRegistry",
    "ecr:DescribeRepositories",
    "ecr:GetAuthorizationToken",
    "ecr:GetDownloadUrlForLayer",
```

```
    "ecr:GetRegistryScanningConfiguration",
    "ecr:ListImages",
    "ecr:PutRegistryScanningConfiguration",
    "organizations:DescribeAccount",
    "organizations:DescribeOrganization",
    "organizations:ListAccounts",
    "ssm:DescribeAssociation",
    "ssm:DescribeAssociationExecutions",
    "ssm:DescribeInstanceInformation",
    "ssm:ListAssociations",
    "ssm:ListResourceDataSync"
   ],
   "Resource": "*"
  },
  {
   "Sid": "LambdaPackageVulnerabilityScanning",
   "Effect": "Allow",
   "Action": [
    "lambda:ListFunctions",
    "lambda:GetFunction",
    "lambda:GetLayerVersion",
    "cloudwatch:GetMetricData"
   ],
   "Resource": "*"
  },
  {
   "Sid": "GatherInventory",
   "Effect": "Allow",
   "Action": [
    "ssm:CreateAssociation",
    "ssm:StartAssociationsOnce",
    "ssm:DeleteAssociation",
    "ssm:UpdateAssociation"
   ],
   "Resource": [
    "arn:aws:ec2:*:*:instance/*",
    "arn:aws:ssm:*:*:document/AmazonInspector2-*",
    "arn:aws:ssm:*:*:document/AWS-GatherSoftwareInventory",
    "arn:aws:ssm:*:*:managed-instance/*",
    "arn:aws:ssm:*:*:association/*"
   ]
  },
  {
   "Sid": "DataSyncCleanup",
```

```
    "Effect": "Allow",
    "Action": [
     "ssm:CreateResourceDataSync",
     "ssm:DeleteResourceDataSync"
    ],
    "Resource": [
     "arn:aws:ssm:*:*:resource-data-sync/InspectorResourceDataSync-do-not-delete"
    ]
   },
   {
    "Sid": "ManagedRules",
    "Effect": "Allow",
    "Action": [
     "events:PutRule",
     "events:DeleteRule",
     "events:DescribeRule",
     "events:ListTargetsByRule",
     "events:PutTargets",
     "events:RemoveTargets"
    ],
    "Resource": [
     "arn:aws:events:*:*:rule/DO-NOT-DELETE-AmazonInspector*ManagedRule"
    ]
   },
   {
    "Sid": "LambdaCodeVulnerabilityScanning",
    "Effect": "Allow",
    "Action": [
     "codeguru-security:CreateScan",
     "codeguru-security:GetAccountConfiguration",
     "codeguru-security:GetFindings",
     "codeguru-security:GetScan",
     "codeguru-security:ListFindings",
     "codeguru-security:BatchGetFindings",
     "codeguru-security:DeleteScansByCategory"
    ],
    "Resource": [
     "*"
    ]
   },
   {
    "Sid": "CodeGuruCodeVulnerabilityScanning",
    "Effect": "Allow",
    "Action": [
```

```
    "iam:GetRole",
    "iam:GetRolePolicy",
    "iam:GetPolicy",
    "iam:GetPolicyVersion",
    "iam:ListAttachedRolePolicies",
    "iam:ListPolicies",
    "iam:ListPolicyVersions",
    "iam:ListRolePolicies",
    "lambda:ListVersionsByFunction"
   ],
   "Resource": [
    "*"
   ],
   "Condition": {
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": [
      "codeguru-security.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "Ec2DeepInspection",
   "Effect": "Allow",
   "Action": [
    "ssm:PutParameter",
    "ssm:GetParameters",
    "ssm:DeleteParameter"
   ],
   "Resource": [
    "arn:aws:ssm:*:*:parameter/inspector-aws/service/inspector-linux-application-paths"
   ],
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
   }
  },
  {
   "Sid": "AllowManagementOfServiceLinkedChannel",
   "Effect": "Allow",
   "Action": [
    "cloudtrail:CreateServiceLinkedChannel",
    "cloudtrail:DeleteServiceLinkedChannel"
```

```
    ],
    "Resource": [
     "arn:aws:cloudtrail:*:*:channel/aws-service-channel/inspector2/*"
    ],
    "Condition": {
     "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
     }
    }
   },
   {
    "Sid": "AllowListServiceLinkedChannels",
    "Effect": "Allow",
    "Action": [
     "cloudtrail:ListServiceLinkedChannels"
    ],
    "Resource": [
     "*"
    ],
    "Condition": {
     "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
     }
    }
   },
   {
    "Sid": "AllowToRunInvokeCisSpecificDocuments",
    "Effect": "Allow",
    "Action": [
     "ssm:SendCommand",
     "ssm:GetCommandInvocation"
    ],
    "Resource": [
     "arn:aws:ssm:*:*:document/AmazonInspector2-InvokeInspectorSsmPluginCIS"
    ]
   },
   {
    "Sid": "AllowToRunCisCommandsToSpecificResources",
    "Effect": "Allow",
    "Action": [
     "ssm:SendCommand"
    ],
    "Resource": [
     "arn:aws:ec2:*:*:instance/*"
```

```
    ],
    "Condition": {
     "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
     }
    }
   },
   {
    "Sid": "AllowToPutCloudwatchMetricData",
    "Effect": "Allow",
    "Action": [
     "cloudwatch:PutMetricData"
    ],
    "Resource": [
     "*"
    ],
    "Condition": {
     "StringEquals": {
      "cloudwatch:namespace": "AWS/Inspector2"
     }
    }
   }
  ]
}
```

为 Amazon Inspector 创建服务相关角色

您无需手动创建服务相关角色。当你在 AWS Management Console、或 AWS API 中激活 Amazon Inspector 时，Amazon Inspector 会为你创建与服务相关的角色。 AWS CLI

为 Amazon Inspector 编辑服务相关角色

Amazon Inspector 不允许编辑 AWSServiceRoleForAmazonInspector2 服务相关角色。在创建服务相关角色后，您无法更改角色的名称，因为可能有多个实体会引用该角色。但是可以使用 IAM 编辑角色说明。有关更多信息，请参阅《IAM 用户指南》中的编辑服务相关角色。

删除适用于 Amazon Inspector 的服务相关角色

如果您不再使用 Amazon Inspector，我们建议您删除 AWSServiceRoleForAmazonInspector2 服务相关角色。在删除角色之前，您必须在每个激活该角色 AWS 区域 的地方停用 Amazon Inspector。停用 Amazon Inspector 时，它不会为您删除该角色。因此，如果您再次激活 Amazon Inspector，它可

以使用现有角色。这样，您就可以避免出现未监控或维护的未使用实体。但是，您必须先清除服务相关角色的资源，然后才能手动删除它。

如果删除此服务相关角色然后需要再次创建它，则可以使用相同的流程在您的账户中重新创建此角色。激活 Amazon Inspector 时，Amazon Inspector 会为您重新创建服务相关角色。

> **Note**
>
> 如果在您试图删除资源时，Amazon Inspector 服务正在使用该角色，则删除操作可能会失败。如果发生这种情况，请等待几分钟，然后再次尝试操作。

您可以使用 IAM 控制台 AWS CLI、或 AWS API 删除`AWSServiceRoleForAmazonInspector2`服务相关角色。有关更多信息，请参阅《IAM 用户指南》中的删除服务相关角色。

Amazon Inspector 无代理扫描的服务相关角色权限

Amazon Inspector 无代理扫描使用名为 `AWSServiceRoleForAmazonInspector2Agentless` 的服务相关角色。这个 SLR 允许 Amazon Inspector 在您的账户中创建 Amazon EBS 卷快照，然后访问该快照中的数据。该服务相关角色信任 `agentless.inspector2.amazonaws.com` 服务担任该角色。

> **Important**
>
> 此服务相关角色中的语句会阻止 Amazon Inspector 对您使用 `InspectorEc2Exclusion` 标签从扫描中排除的任何 EC2 实例执行无代理扫描。此外，当用于加密卷的 KMS 密钥带有 `InspectorEc2Exclusion` 标签时，这些语句会阻止 Amazon Inspector 访问相应卷中的加密数据。有关更多信息，请参阅从 Amazon Inspector 扫描中排除实例。

该角色的权限策略名为 `AmazonInspector2AgentlessServiceRolePolicy`，允许 Amazon Inspector 执行以下任务：

- 使用 Amazon Elastic Compute Cloud（Amazon EC2）操作检索有关 EC2 实例、卷和快照的信息。
  - 使用 Amazon EC2 标记操作，用 `InspectorScan` 标签键为扫描的快照添加标签。
  - 使用 Amazon EC2 快照操作创建快照，用 `InspectorScan` 标签键为其添加标签，然后删除 Amazon EBS 卷带有 `InspectorScan` 标签键的快照。
- 使用 Amazon EBS 操作，从带有 `InspectorScan` 标签键的快照中检索信息。

- 使用选择 AWS KMS 解密操作来解密使用客户托管密钥加密的 AWS KMS 快照。当用于加密快照的 KMS 密钥带有 `InspectorEc2Exclusion` 标签时，Amazon Inspector 不会解密相应快照。

该角色使用以下权限策略进行配置：

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "InstanceIdentification",
   "Effect": "Allow",
   "Action": [
    "ec2:DescribeInstances",
    "ec2:DescribeVolumes",
    "ec2:DescribeSnapshots"
   ],
   "Resource": "*"
  },
  {
   "Sid": "GetSnapshotData",
   "Effect": "Allow",
   "Action": [
    "ebs:ListSnapshotBlocks",
    "ebs:GetSnapshotBlock"
   ],
   "Resource": "arn:aws:ec2:*:*:snapshot/*",
   "Condition": {
    "StringLike": {
     "aws:ResourceTag/InspectorScan": "*"
    }
   }
  },
  {
   "Sid": "CreateSnapshotsAnyInstanceOrVolume",
   "Effect": "Allow",
   "Action": "ec2:CreateSnapshots",
   "Resource": [
    "arn:aws:ec2:*:*:instance/*",
    "arn:aws:ec2:*:*:volume/*"
   ]
  },
  {
```

```
    "Sid": "DenyCreateSnapshotsOnExcludedInstances",
    "Effect": "Deny",
    "Action": "ec2:CreateSnapshots",
    "Resource": "arn:aws:ec2:*:*:instance/*",
    "Condition": {
     "StringEquals": {
      "ec2:ResourceTag/InspectorEc2Exclusion": "true"
     }
    }
   },
   {
    "Sid": "CreateSnapshotsOnAnySnapshotOnlyWithTag",
    "Effect": "Allow",
    "Action": "ec2:CreateSnapshots",
    "Resource": "arn:aws:ec2:*:*:snapshot/*",
    "Condition": {
     "Null": {
      "aws:TagKeys": "false"
     },
     "ForAllValues:StringEquals": {
      "aws:TagKeys": "InspectorScan"
     }
    }
   },
   {
    "Sid": "CreateOnlyInspectorScanTagOnlyUsingCreateSnapshots",
    "Effect": "Allow",
    "Action": "ec2:CreateTags",
    "Resource": "arn:aws:ec2:*:*:snapshot/*",
    "Condition": {
     "StringLike": {
      "ec2:CreateAction": "CreateSnapshots"
     },
     "Null": {
      "aws:TagKeys": "false"
     },
     "ForAllValues:StringEquals": {
      "aws:TagKeys": "InspectorScan"
     }
    }
   },
   {
    "Sid": "DeleteOnlySnapshotsTaggedForScanning",
    "Effect": "Allow",
```

```
    "Action": "ec2:DeleteSnapshot",
    "Resource": "arn:aws:ec2:*:*:snapshot/*",
    "Condition": {
     "StringLike": {
      "ec2:ResourceTag/InspectorScan": "*"
     }
    }
   },
   {
    "Sid": "DenyKmsDecryptForExcludedKeys",
    "Effect": "Deny",
    "Action": "kms:Decrypt",
    "Resource": "arn:aws:kms:*:*:key/*",
    "Condition": {
     "StringEquals": {
      "aws:ResourceTag/InspectorEc2Exclusion": "true"
     }
    }
   },
   {
    "Sid": "DecryptSnapshotBlocksVolContext",
    "Effect": "Allow",
    "Action": "kms:Decrypt",
    "Resource": "arn:aws:kms:*:*:key/*",
    "Condition": {
     "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
     },
     "StringLike": {
      "kms:ViaService": "ec2.*.amazonaws.com",
      "kms:EncryptionContext:aws:ebs:id": "vol-*"
     }
    }
   },
   {
    "Sid": "DecryptSnapshotBlocksSnapContext",
    "Effect": "Allow",
    "Action": "kms:Decrypt",
    "Resource": "arn:aws:kms:*:*:key/*",
    "Condition": {
     "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
     },
     "StringLike": {
```

```
      "kms:ViaService": "ec2.*.amazonaws.com",
      "kms:EncryptionContext:aws:ebs:id": "snap-*"
     }
    }
   },
   {
    "Sid": "DescribeKeysForEbsOperations",
    "Effect": "Allow",
    "Action": "kms:DescribeKey",
    "Resource": "arn:aws:kms:*:*:key/*",
    "Condition": {
     "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
     },
     "StringLike": {
      "kms:ViaService": "ec2.*.amazonaws.com"
     }
    }
   },
   {
    "Sid": "ListKeyResourceTags",
    "Effect": "Allow",
    "Action": "kms:ListResourceTags",
    "Resource": "arn:aws:kms:*:*:key/*"
   }
  ]
}
```

创建用于无代理扫描的服务相关角色

您无需手动创建服务相关角色。当你在 AWS Management Console、或 AWS API 中激活 Amazon Inspector 时，Amazon Inspector 会为你创建与服务相关的角色。 AWS CLI

编辑用于无代理扫描的服务相关角色

Amazon Inspector 不允许编辑 `AWSServiceRoleForAmazonInspector2Agentless` 服务相关角色。在创建服务相关角色后，您无法更改角色的名称，因为可能有多个实体会引用该角色。但是可以使用 IAM 编辑角色说明。有关更多信息，请参阅《IAM 用户指南》中的编辑服务相关角色。

删除用于无代理扫描的服务相关角色

如果不再需要使用某个需要服务相关角色的功能或服务，我们建议您删除该角色。这样您就没有未被主动监控或维护的未使用实体。

> **Important**
>
> 要删除 AWSServiceRoleForAmazonInspector2Agentless 角色，您必须在所有支持无代理扫描的区域中将扫描模式设置为基于代理。有关更多信息，请参阅 [待定的扫描模式设置链接]。

使用 IAM 手动删除服务相关角色

使用 IAM 控制台 AWS CLI、或 AWS API 删除 AWSServiceRoleForAmazonInspector2Agentless 服务相关角色。有关更多信息，请参阅《IAM 用户指南》中的删除服务相关角色。

Amazon Inspector 身份和访问问题排查

您可以使用以下信息，帮助诊断和修复在使用 Amazon Inspector 和 IAM 时可能遇到的常见问题。

主题

- 我无权在 Amazon Inspector 中执行操作
- 我无权执行 iam：PassRole
- 我想允许我以外的人访问我的 Amazon Inspector 资源 AWS 账户

我无权在 Amazon Inspector 中执行操作

如果您收到错误提示，表明您无权执行某个操作，则您必须更新策略以允许执行该操作。

当 mateojackson IAM 用户尝试使用控制台查看有关虚构 *my-example-widget* 资源的详细信息，但不拥有虚构 inspector2:*GetWidget* 权限时，会发生以下示例错误。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 inspector2:GetWidget on resource: my-example-widget
```

在此情况下，必须更新 mateojackson 用户的策略，以允许使用 inspector2:*GetWidget* 操作访问 *my-example-widget* 资源。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我无权执行 iam：PassRole

如果您收到一个错误，指明您无权执行 `iam:PassRole` 操作，则必须更新策略以允许您将角色传递给 Amazon Inspector。

有些 AWS 服务 允许您将现有角色传递给该服务，而不是创建新的服务角色或服务相关角色。为此，您必须具有将角色传递到服务的权限。

当名为 `marymajor` 的 IAM 用户尝试使用控制台在 Amazon Inspector 中执行操作时，会发生以下示例错误。但是，服务必须具有服务角色所授予的权限才可执行此操作。Mary 不具有将角色传递到服务的权限。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

在这种情况下，必须更新 Mary 的策略以允许她执行 `iam:PassRole` 操作。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我想允许我以外的人访问我的 Amazon Inspector 资源 AWS 账户

您可以创建一个角色，以便其他账户中的用户或您组织外的人员可以使用该角色来访问您的资源。您可以指定谁值得信赖，可以代入角色。对于支持基于资源的策略或访问控制列表（ACL）的服务，您可以使用这些策略向人员授予对您的资源的访问权。

要了解更多信息，请参阅以下内容：

- 要了解 Amazon Inspector 是否支持这些功能，请参阅Amazon Inspector 如何与 IAM 配合使用。
- 要了解如何提供对您拥有的资源的访问权限 AWS 账户 ，请参阅 IAM 用户指南中的向您拥有 AWS 账户 的另一个 IAM 用户提供访问权限。
- 要了解如何向第三方提供对您的资源的访问权限 AWS 账户，请参阅 IAM 用户指南中的向第三方提供访问权限。 AWS 账户
- 要了解如何通过身份联合验证提供访问权限，请参阅《IAM 用户指南》中的为经过外部身份验证的用户（身份联合验证）提供访问权限。
- 要了解使用角色和基于资源的策略进行跨账户访问之间的差别，请参阅《IAM 用户指南》中的 IAM 角色与基于资源的策略有何不同。

监控 Amazon Inspector

监控是维护 Amazon Inspector 和其他 AWS 解决方案的可靠性、可用性和性能的重要组成部分。 AWS 提供监控工具，用于监视 Amazon Inspector，在出现问题时进行报告，并在适当时自动采取行动：

- Amazon EventBridge 是一项无服务器事件总线服务，可以轻松地将您的应用程序与来自各种来源的数据连接起来。 EventBridge 提供来自您自己的应用程序、S oftware-as-a-Service (SaaS) 应用程序和 AWS 服务的实时数据流，并将这些数据路由到 Lambda 等目标。这使您能够监控服务中发生的事件，并构建事件驱动的架构。有关更多信息，请参阅 Amazon EventBridge 用户指南。
- AWS CloudTrail 捕获由某个 AWS 账户发出或代表该账户发出的 API 调用和相关事件。 CloudTrail 然后将日志文件传送到您指定的 Amazon S3 存储桶。您可以识别哪些用户和帐户拨打了电话 AWS、发出呼叫的源 IP 地址以及呼叫发生的时间。有关更多信息，请参阅 AWS CloudTrail 用户指南。

使用 AWS CloudTrail记录 Amazon Inspector API 调用

Amazon Inspector 与 AWS CloudTrail一项服务集成，该服务提供了 IAM 用户或角色或 Amazon Inspect AWS 服务 or 中的角色所采取的操作的记录。 CloudTrail 将 Amazon Inspector 的所有 API 调用捕获为事件。捕获调用中包括通过 Amazon Inspector 控制台的调用和对 Amazon Inspector API 操作的调用。如果您创建了跟踪，则可以允许将 CloudTrail 事件持续传输到 Amazon S3 存储桶，包括针对 Amazon Inspector 的事件。如果您不配置跟踪记录，则仍可在 CloudTrail 控制台的 Event history（事件历史记录）中查看最新事件。使用收集的信息 CloudTrail，您可以确定：

- 向 Amazon Inspector 发出的请求。
- 已从中发出请求的 IP 地址。
- 谁发出了请求。
- 发出请求的时间。

要了解更多信息 CloudTrail，请参阅《AWS CloudTrail 用户指南》。

Amazon Inspector 中的信息 CloudTrail

CloudTrail 在您创建账户 AWS 账户 时已在您的账户上启用。当 Amazon Inspector 中发生活动时，该活动会与其他 CloudTrail AWS 服务 事件一起记录在事件历史记录中。您可以在中查看、搜索和下载最近发生的事件 AWS 账户。有关更多信息，请参阅使用事件历史记录查看 CloudTrail 事件。

要持续记录您的事件 AWS 账户，包括 Amazon Inspector 的事件，请创建跟踪。跟踪允许 CloudTrail 将日志文件传输到 Amazon S3 存储桶。预设情况下，在控制台中创建跟踪记录时，此跟踪记录应用于所有 AWS 区域。此跟踪记录在 AWS 分区中记录所有区域中的事件，并将日志文件传送至您指定的 Amazon S3 存储桶。此外，您可以配置其他 AWS 服务 ，以进一步分析和处理 CloudTrail 日志中收集的事件数据。有关更多信息，请参阅以下主题：

- 创建跟踪记录概述
- CloudTrail 支持的服务和集成
- 为 CloudTrail 配置 Amazon SNS 通知
- 接收来自多个账户的 CloudTrail 日志文件
- 接收来自多个区域的 CloudTrail 日志文件

Amazon Inspector 的所有操作都由记录 CloudTrail。Amazon Inspector 可以执行的所有操作都记录在 Amazon Inspector API 参考中。例如，对 `CreateFindingsReport`、`ListCoverage` 和 `UpdateOrganizationConfiguration` 操作的调用会在 CloudTrail 日志文件中生成条目。

每个事件或日记账条目都包含有关生成请求的人员信息。身份信息有助于您确定以下内容：

- 请求是使用根用户凭证还是 IAM 用户凭证发出的。
- 请求是使用角色还是联合用户的临时安全凭证发出的。
- 请求是否由其他 AWS 服务发出。

有关更多信息，请参阅 CloudTrail userIdentity 元素。

了解 Amazon Inspector 日志文件条目

跟踪是一种配置，允许将事件作为日志文件传输到您指定的 Amazon S3 存储桶。 CloudTrail 日志文件包含一个或多个日志条目。事件表示来自任何源的单个请求。事件包括有关所请求操作的信息、操作的日期和时间、请求参数等。 CloudTrail 日志文件不是公共 API 调用的有序堆栈跟踪，因此它们不会按任何特定的顺序出现。

Amazon Inspector 扫描中的信息 CloudTrail

Amazon Inspector Scan 已与集成 CloudTrail。所有 Amazon Inspector Scan API 操作都记录为管理事件。有关亚马逊 Inspector 登录的亚马逊 Inspector Scan API 操作的列表 CloudTrail，请参阅《亚马逊检查器 API 参考》中的 Amazon Inspector Scan。

以下示例显示了演示该ScanSbom操作的 CloudTrail 日志条目：

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROA123456789EXAMPLE:akua_mansa",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/akua_mansa",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROA123456789EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-10-17T15:22:59Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-10-17T16:02:34Z",
    "eventSource": "gamma-inspector-scan.amazonaws.com",
    "eventName": "ScanSbom",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "203.0.113.0",
    "userAgent": "aws-sdk-java/2.20.162 Mac_OS_X/13.5.2 OpenJDK_64-
Bit_Server_VM/17.0.8+7-LTS Java/17.0.8 vendor/Amazon.com_Inc. io/sync http/
UrlConnection cfg/retry-mode/legacy",
    "requestParameters": {
        "sbom": {
            "specVersion": "1.5",
            "metadata": {
                "component": {
                    "name": "debian",
                    "type": "operating-system",
                    "version": "9"
                }
            },
```

```
            "components": [
                {
                    "name": "packageOne",
                    "purl": "pkg:deb/debian/packageOne@1.0.0?arch=x86_64&distro=9",
                    "type": "application"
                }
            ],
            "bomFormat": "CycloneDX"
        }
    },
    "responseElements": null,
    "requestID": "f041a27f-f33e-4f70-b09b-5fbc5927282a",
    "eventID": "abc8d1e4-d214-4f07-bc56-8a31be6e36fe",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


Amazon Inspector 的合规性验证

要了解是否属于特定合规计划的范围，请参阅AWS 服务 “按合规计划划分的范围” ”，然后选择您感兴趣的合规计划。 AWS 服务 有关一般信息，请参阅AWS 合规计划AWS。

您可以使用下载第三方审计报告 AWS Artifact。有关更多信息，请参阅中的 “下载报告” 中的 “ AWS Artifact。

您在使用 AWS 服务 时的合规责任取决于您的数据的敏感性、贵公司的合规目标以及适用的法律和法规。 AWS 提供了以下资源来帮助实现合规性：

- 安全与合规性快速入门指南 — 这些部署指南讨论了架构注意事项，并提供了部署以安全性和合规性为重点 AWS 的基准环境的步骤。
- 在 A@@ mazon Web Services 上构建 HIPAA 安全与合规性 — 本白皮书描述了各公司如何使用 AWS 来创建符合 HIPAA 资格的应用程序。

> **Note**
>
> 并非所有 AWS 服务 人都符合 HIPAA 资格。有关更多信息，请参阅符合 HIPAA 要求的服务参考。

- AWS 合规资源AWS — 此工作簿和指南集可能适用于您所在的行业和所在地区。
- AWS 客户合规指南 — 从合规角度了解责任共担模式。这些指南总结了保护的最佳实践， AWS 服务并将指南映射到跨多个框架（包括美国国家标准与技术研究院 (NIST)、支付卡行业安全标准委员会 (PCI) 和国际标准化组织 (ISO)）的安全控制。
- 使用AWS Config 开发人员指南中的规则评估资源 — 该 AWS Config 服务评估您的资源配置在多大程度上符合内部实践、行业准则和法规。
- AWS Security Hub— 这 AWS 服务 提供了您内部安全状态的全面视图 AWS。Security Hub 通过安全控件评估您的 AWS 资源并检查其是否符合安全行业标准和最佳实践。有关受支持服务及控制的列表，请参阅 Security Hub 控制参考。
- AWS Audit Manager— 这 AWS 服务 可以帮助您持续审计 AWS 使用情况，从而简化风险管理以及对法规和行业标准的合规性。

Amazon Inspector 故障恢复能力

AWS 全球基础设施是围绕 AWS 区域 可用区构建的。 AWS 区域 提供多个物理隔离和隔离的可用区，这些可用区通过低延迟、高吞吐量和高度冗余的网络连接。利用可用区，您可以设计和操作在可用区之间无中断地自动实现失效转移的应用程序和数据库。与传统的单个或多个数据中心基础设施相比，可用区具有更高的可用性、容错性和可扩展性。

Amazon Inspector 基础设施安全性

作为一项托管服务，Amazon Inspector 受到 AWS 全球网络安全的保护。有关 AWS 安全服务以及如何 AWS 保护基础设施的信息，请参阅AWS 云安全。要使用基础设施安全的最佳实践来设计您的 AWS 环境，请参阅 S AWS ecurity Pillar Well-Architected Fram ework 中的基础设施保护。

您可以使用 AWS 已发布的 API 调用通过网络访问 Amazon Inspector。客户端必须支持以下内容：

- 传输层安全性协议 (TLS) 我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 具有完全向前保密 (PFS) 的密码套件，例如 DHE（临时 Diffie-Hellman）或 ECDHE（临时椭圆曲线 Diffie-Hellman）。大多数现代系统（如 Java 7 及更高版本）都支持这些模式。

此外，必须使用访问密钥 ID 和与 IAM 主体关联的秘密访问密钥来对请求进行签名。或者，您可以使用 AWS Security Token Service（AWS STS）生成临时安全凭证来对请求进行签名。

Amazon Inspector 中的事件响应

AWS非常重视安全性。作为 AWS 云责任共担模式的一部分， AWS 管理满足大多数安全敏感组织要求的数据中心、网络和软件架构。 AWS 负责与 AWS Config 服务本身有关的任何事件响应。此外，作为 AWS 客户，您也有责任维护云端的安全。这意味着您可以从自己有权访问的 AWS 工具和功能中控制您选择实施的安全性，并负责在责任共担模型中对事件做出响应。

通过建立符合云端运行应用程序目标的安全基准，您可以检测出可以响应的偏差。由于安全事件响应可能是一个复杂的主题，因此我们鼓励您查看以下资源，以便更好地了解事件响应 (IR) 和您的选择对企业目标的影响：AWS 安全事件响应指南、AWS 安全最佳实践白皮书和AWS 云采用框架 (CAF) 的安全视角白皮书。

Amazon Inspector 集成

Amazon Inspector 与其他 AWS 服务集成。这些服务可以从 Amazon Inspector 摄取数据，以便您以新的方式查看调查发现。查看以下集成选项，详细了解如何设置服务以便与 Amazon Inspector 配合使用。

Amazon Inspector 与 Amazon ECR 集成

Amazon Elastic Container Registry (Amazon ECR) 是一个完全托管式 Docker 容器注册表，可轻松地存储、共享和部署容器映像。Amazon ECR 私有注册表在可用性和可扩展性都非常高的架构中托管您的容器映像。您可以使用 Amazon Inspector 扫描驻留在 Amazon ECR 存储库中的容器映像，查找易受攻击的操作系统程序包和编程语言包。

有关将 Amazon ECR 与 Amazon Inspector 结合使用的更多信息，请参阅Amazon Inspector 与 Amazon Elastic Container Registry (Amazon ECR) 集成。

Amazon Inspector 与 AWS Security Hub 集成

AWS Security Hub 从您的 AWS 账户、服务和其他受支持产品收集安全数据，以根据行业标准和最佳实践评测您环境的安全状态。除了评估您的安全状况之外，Security Hub 还为所有集成的 AWS 服务和 AWS 合作伙伴网络产品的调查发现提供了一个集中位置。使用 Amazon Inspector 激活 Security Hub 可自动允许 Security Hub 摄取 Amazon Inspector 调查发现数据。

有关将 Security Hub 与 Amazon Inspector 结合使用的更多信息，请参阅Amazon Inspector 与 AWS Security Hub 集成。

Amazon Inspector 与 Amazon Elastic Container Registry (Amazon ECR) 集成

Amazon ECR 是一个完全托管的容器注册表，支持 Docker、OCI 映像和 AWS 构件。如果您正在使用 Amazon ECR，可以激活注册表的增强扫描，以允许 Amazon Inspector 自动检测容器映像，并对其进行扫描，查找易受攻击的操作系统程序包和编程语言包。

此集成使您可以在 Amazon ECR 控制台中查看容器映像的 Amazon Inspector 调查发现。此外，在 Amazon ECR 控制台中，您还可以通过创建包含筛选条件来管理扫描频率和调整扫描范围。

激活集成

您可以通过使用 Amazon Inspector 控制台或 API 激活 Amazon Inspector 扫描来激活此集成，也可以通过 Amazon ECR 控制台或 API 配置您的存储库以使用 Amazon Inspector 的增强扫描，从而激活此集成。

有关通过 Amazon Inspector 激活集成的更多信息，请参阅使用 Amazon Inspector 自动扫描资源。

有关在 Amazon ECR 中激活和配置增强扫描的信息，请参阅 Amazon ECR 用户指南中的增强扫描。

使用与多账户环境的集成

如果您是多账户环境的成员，则可以通过 Amazon ECR 激活增强型扫描。但是，一旦激活，则只能由您的 Amazon Inspector 委托管理员停用。如果停用，则恢复为基本扫描。有关更多信息，请参阅停用 Amazon Inspector。

Amazon Inspector 与 AWS Security Hub 集成

Security Hub 提供了 AWS 中安全状态的全面视图，可帮助您检查环境是否符合安全行业标准和最佳实践。Security Hub 从 AWS 账户、服务和其他受支持产品中收集安全数据。您可以使用它提供的信息来分析安全趋势并确定优先级最高的安全问题。

Amazon Inspector 与 Security Hub 的集成让您可以将调查发现从 Amazon Inspector 发送到 Security Hub。随后，Security Hub 可以在对您的安全状况进行分析时使用这些调查发现。

在 AWS Security Hub 中，安全问题以调查发现的形式进行跟踪。有些调查发现来自其他 AWS 服务或第三方产品检测到的问题。Security Hub 还有一套用于检测安全问题和生成结果的规则。Security Hub 提供了管理来自所有这些来源的结果的工具。您可以查看和筛选调查发现列表，并查看调查发现的详细信息。有关 Security Hub 中的调查发现的更多信息，请参阅 AWS Security Hub 用户指南中的查看调查发现。您还可以跟踪调查发现的调查状态。请参阅 AWS Security Hub 用户指南中的对调查发现采取措施。

Security Hub 中的所有调查发现都使用标准的 JSON 格式，即 AWS Security Finding 格式 (ASFF)。ASFF 包含有关问题根源、受影响资源以及调查发现当前状态的详细信息。请参阅 AWS Security Hub 用户指南中的 AWS Security Finding 格式 (ASFF) 。

Amazon Inspector 调查发现得到解决并在 Amazon Inspector 中关闭后，Security Hub 将存档这些调查发现。

在 AWS Security Hub 中查看 Amazon Inspector 调查发现

Amazon Inspector Classic 和新 Amazon Inspector 的调查发现都可在 Security Hub 的同一面板中查看。但是，您可以通过在筛选栏中添加 "aws/inspector/ProductVersion": "2" 来筛选新 Amazon Inspector 的调查发现。添加此筛选条件会从 Security Hub 控制面板中排除 Amazon Inspector Classic 的调查发现。

Amazon Inspector 调查发现示例


```
{
  "SchemaVersion": "2018-10-08",
  "Id": "arn:aws:inspector2:us-east-1:123456789012:finding/FINDING_ID",
  "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/inspector",
  "ProductName": "Inspector",
  "CompanyName": "Amazon",
  "Region": "us-east-1",
  "GeneratorId": "AWSInspector",
  "AwsAccountId": "123456789012",
  "Types": [
    "Software and Configuration Checks/Vulnerabilities/CVE"
  ],
  "FirstObservedAt": "2023-01-31T20:25:38Z",
  "LastObservedAt": "2023-05-04T18:18:43Z",
  "CreatedAt": "2023-01-31T20:25:38Z",
  "UpdatedAt": "2023-05-04T18:18:43Z",
  "Severity": {
    "Label": "HIGH",
    "Normalized": 70
  },
  "Title": "CVE-2022-34918 - kernel",
  "Description": "An issue was discovered in the Linux kernel through 5.18.9. A type
confusion bug in nft_set_elem_init (leading to a buffer overflow) could be used by a
local attacker to escalate privileges, a different vulnerability than CVE-2022-32250.
(The attacker can obtain root access, but must start with an unprivileged user
namespace to obtain CAP_NET_ADMIN access.) This can be fixed in nft_setelem_parse_data
in net/netfilter/nf_tables_api.c.",
  "Remediation": {
    "Recommendation": {
      "Text": "Remediation is available. Please refer to the Fixed version in the
vulnerability details section above. For detailed remediation guidance for each of the
affected packages, refer to the vulnerabilities section of the detailed finding JSON."
    }
  },
```

```
  "ProductFields": {
    "aws/inspector/FindingStatus": "ACTIVE",
    "aws/inspector/inspectorScore": "7.8",
    "aws/inspector/resources/1/resourceDetails/awsEc2InstanceDetails/platform":
 "AMAZON_LINUX_2",
    "aws/inspector/ProductVersion": "2",
    "aws/inspector/instanceId": "i-0f1ed287081bdf0fb",
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-east-1::product/aws/inspector/
arn:aws:inspector2:us-east-1:123456789012:finding/FINDING_ID",
    "aws/securityhub/ProductName": "Inspector",
    "aws/securityhub/CompanyName": "Amazon"
  },
  "Resources": [
    {
      "Type": "AwsEc2Instance",
      "Id": "arn:aws:ec2:us-east-1:123456789012:i-0f1ed287081bdf0fb",
      "Partition": "aws",
      "Region": "us-east-1",
      "Tags": {
        "Patch Group": "SSM",
        "Name": "High-SEv-Test"
      },
      "Details": {
        "AwsEc2Instance": {
          "Type": "t2.micro",
          "ImageId": "ami-0cff7528ff583bf9a",
          "IpV4Addresses": [
            "52.87.229.97",
            "172.31.57.162"
          ],
          "KeyName": "ACloudGuru",
          "IamInstanceProfileArn": "arn:aws:iam::123456789012:instance-profile/
AmazonSSMRoleForInstancesQuickSetup",
          "VpcId": "vpc-a0c2d7c7",
          "SubnetId": "subnet-9c934cb1",
          "LaunchedAt": "2022-07-26T21:49:46Z"
        }
      }
    }
  ],
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NEW"
  },
```

```
  "RecordState": "ACTIVE",
  "Vulnerabilities": [
    {
      "Id": "CVE-2022-34918",
      "VulnerablePackages": [
        {
          "Name": "kernel",
          "Version": "5.10.118",
          "Epoch": "0",
          "Release": "111.515.amzn2",
          "Architecture": "X86_64",
          "PackageManager": "OS",
          "FixedInVersion": "0:5.10.130-118.517.amzn2",
          "Remediation": "yum update kernel"
        }
      ],
      "Cvss": [
        {
          "Version": "2.0",
          "BaseScore": 7.2,
          "BaseVector": "AV:L/AC:L/Au:N/C:C/I:C/A:C",
          "Source": "NVD"
        },
        {
          "Version": "3.1",
          "BaseScore": 7.8,
          "BaseVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H",
          "Source": "NVD"
        },
        {
          "Version": "3.1",
          "BaseScore": 7.8,
          "BaseVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H",
          "Source": "NVD",
          "Adjustments": []
        }
      ],
      "Vendor": {
        "Name": "NVD",
        "Url": "https://nvd.nist.gov/vuln/detail/CVE-2022-34918",
        "VendorSeverity": "HIGH",
        "VendorCreatedAt": "2022-07-04T21:15:00Z",
        "VendorUpdatedAt": "2022-10-26T17:05:00Z"
      },
```

```
      "ReferenceUrls": [
        "https://git.kernel.org/pub/scm/linux/kernel/git/netdev/net.git/commit/?
id=7e6bc1f6cabcd30aba0b11219d8e01b952eacbb6",
        "https://lore.kernel.org/netfilter-devel/cd9428b6-7ffb-dd22-d949-
d86f4869f452@randorisec.fr/T/",
        "https://www.debian.org/security/2022/dsa-5191"
      ],
      "FixAvailable": "YES"
    }
  ],
  "FindingProviderFields": {
    "Severity": {
      "Label": "HIGH"
    },
    "Types": [
      "Software and Configuration Checks/Vulnerabilities/CVE"
    ]
  },
  "ProcessedAt": "2023-05-05T20:28:38.822Z"
}
```


激活和配置集成

要使用 Amazon Inspector 与 AWS Security Hub 的集成，必须激活 Security Hub。有关如何激活 Security Hub 的信息，请参阅 AWS Security Hub 用户指南中的设置 Security Hub。

同时激活 Amazon Inspector 和 Security Hub 后，集成会自动激活，Amazon Inspector 将开始向 Security Hub 发送调查发现。Amazon Inspector 使用 AWS Security Finding 格式 (ASFF) 将其生成的所有调查发现发送到 Security Hub。

停止向 AWS Security Hub 发布调查发现

如何停止发送调查发现

要停止向 Security Hub 发送结果，您可以使用 Security Hub 控制台或 API。

请参阅 AWS Security Hub 用户指南中的停用和激活来自集成（控制台）的调查发现流或停用来自集成（Security Hub API、AWS CLI）的调查发现流。

Amazon Inspector 支持的操作系统和编程语言

Amazon Inspector 可以扫描安装在亚马逊弹性计算云 (Amazon EC2) 实例上的软件应用程序、存储在亚马逊弹性容器注册表 (Amazon ECR) Elastic Registry 存储库中的容器映像以及函数。 AWS Lambda 对于 ECR 容器镜像，Amazon Inspector 可以扫描操作系统和编程语言包的漏洞。对于 Lambda 函数，Amazon Inspector 可以扫描代码漏洞。当 Amazon Inspector 扫描资源时，它使用自己的专用扫描引擎，并采集 50 多个数据源来生成常见脆弱性和风险 (CVE) 调查发现。数据源包括供应商安全公告、NVD、MITRE、开源信息源、内部研究和许可的数据源。

要让 Amazon Inspector 扫描资源，相应资源必须运行支持的操作系统或使用支持的编程语言。本节中的主题列出了 Amazon Inspector 目前为不同资源和扫描类型所支持的操作系统、运行时和编程语言。它们还列出了以前支持 Amazon Inspector 但后来被供应商停产的操作系统。在供应商停止对操作系统的支持后，Amazon Inspector 只能为相应操作系统提供有限的支持。

主题

支持的操作系统：Amazon EC2 扫描

下表列出了 Amazon Inspector 目前支持扫描亚马逊 EC2 实例的操作系统。它还列出了每份供应商安全公告的来源，以及该操作系统是否可以使用基于代理的扫描方法或无代理的扫描方法进行扫描。有关扫描方法的更多信息，请参阅基于代理的扫描和无代理扫描。

> **Note**
>
> 仅默认软件包管理器存储库支持 Linux 操作系统检测，不包括第三方应用程序、扩展支持存储库（例如 BYOS RHEL、PAYG RHEL 和 SAP 版 RHEL）和可选存储库，例如红帽应用程序流。

操作系统	版本	供应商安全公告	无代理扫描支持	基于代理的扫描支持
AlmaLinux	8	ALSA	支持	支持
AlmaLinux	9	ALSA	支持	支持
Amazon Linux (AL2)	AL2	ALAS	支持	支持
Amazon Linux 2023（AL2023）	AL2023	ALAS	支持	支持
Bottlerocket	1.7.0 及更高版本	GHSA、CVE	不支持	支持
CentOS Linux (CentOS)	7	CESA	支持	支持
Debian 服务器 (Buster)	10	DSA	支持	支持
Debian 服务器 (Bullseye)	11	DSA	支持	支持
Debian 服务器 (Bookworm)	12	DSA	支持	支持
Fedora	38	CVE	支持	支持
Fedora	39	CVE	支持	支持
OpenSUSE	15.5	CVE	支持	支持

操作系统	版本	供应商安全公告	无代理扫描支持	基于代理的扫描支持
Oracle Linux (Oracle)	7	ELSA	支持	支持
Oracle Linux (Oracle)	8	ELSA	支持	支持
Oracle Linux (Oracle)	9	ELSA	支持	支持
Red Hat Enterprise Linux (RHEL)	7	RHSA	支持	支持
Red Hat Enterprise Linux (RHEL)	8	RHSA	支持	支持
Red Hat Enterprise Linux (RHEL)	9	RHSA	支持	支持
Rocky Linux	8	RLSA	支持	支持
Rocky Linux	9	RLSA	支持	支持
SUSE Linux Enterprise Server (SLES)	12.4	SUSE CVE	支持	支持
SUSE Linux Enterprise Server (SLES)	12.5	SUSE CVE	支持	支持
SUSE Linux Enterprise Server (SLES)	15.3	SUSE CVE	支持	支持

操作系统	版本	供应商安全公告	无代理扫描支持	基于代理的扫描支持
SUSE Linux Enterprise Server (SLES)	15.4	SUSE CVE	支持	支持
SUSE Linux Enterprise Server (SLES)	15.5	SUSE CVE	支持	支持
Ubuntu (Trusty)	14.04 (ESM)	USN、Ubuntu Pro	支持	支持
Ubuntu (Xenial)	16.04 (ESM)	USN、Ubuntu Pro	支持	支持
Ubuntu (Bionic)	18.04 (ESM)	USN、Ubuntu Pro	支持	支持
Ubuntu (Focal)	20.04 (LTS)	USN	支持	支持
Ubuntu (Jammy)	22.04 (LTS)	USN	支持	支持
Ubuntu (Mantic Minotaur)	23.10	USN	支持	支持
Windows Server	2016	MSKB	不支持	支持
Windows Server	2019	MSKB	不支持	支持
Windows Server	2022	MSKB	不支持	支持
macOS（莫哈韦沙漠）	10.14	APPLE-SA	不支持	支持
macOS（卡塔琳娜）	10.15	APPLE-SA	不支持	支持

操作系统	版本	供应商安全公告	无代理扫描支持	基于代理的扫描支持
macOS（大苏尔）	11	APPLE-SA	不支持	支持
macOS（蒙特雷）	12	APPLE-SA	不支持	支持
macOS（Ventura）	13	APPLE-SA	不支持	支持

支持的编程语言：Amazon EC2 深度检查

在扫描 Amazon EC2 Linux 实例以查找第三方软件包中的漏洞时，Amazon Inspector 目前支持以下编程语言：

- Java
- JavaScript
- Python

Amazon Inspector 使用 Systems Manager Distributor 在你的亚马逊 EC2 实例中部署用于深度检查的插件。Systems Manager Distributor 支持 Systems Manager 指南内的支持的程序包平台和架构中列出的操作系统。您的 Amazon EC2 实例的操作系统是 Systems Manager Distributor 和 Amazon Inspector 支持的系统，Amazon Inspector 才能执行深度检查扫描。

> **Note**
>
> Bottlerocket 操作系统不支持深度检查。

支持的操作系统：CIS 扫描

下表列出了 Amazon Inspector 目前支持 CIS 扫描的操作系统。该表还包括用于对该操作系统执行扫描的 CIS 基准测试版本。

操作系统	版本	CIS 基准测试版本
Amazon Linux 2	AL2	2.0.0
Amazon Linux 2023	AL2023	1.0.0
Windows Server	2019	2.0.0
Windows Server	2022	2.0.0

支持的操作系统：使用 Amazon Inspector 进行亚马逊 ECR 扫描

Amazon Inspector 目前支持在扫描 Amazon ECR 存储库中的容器映像时扫描以下操作系统：。该表还列出了每个操作系统的供应商安全公告来源。

操作系统	版本	供应商安全公告
Alpine Linux (Alpine)	3.16	Alpine SecDB
Alpine Linux (Alpine)	3.17	Alpine SecDB
Alpine Linux (Alpine)	3.18	Alpine SecDB
Alpine Linux (Alpine)	3.19	Alpine SecDB
AlmaLinux	8	ALSA
AlmaLinux	9	ALSA
Amazon Linux (AL2)	AL2	ALAS
Amazon Linux 2023 (AL2023)	AL2023	ALAS
CentOS Linux (CentOS)	7	CESA
Debian Server (Buster)	10	DSA
Debian Server (Bullseye)	11	DSA

操作系统	版本	供应商安全公告
Debian Server (Bookworm)	12	DSA
Fedora	38	CVE
Fedora	39	CVE
OpenSUSE	15.5	CVE
Oracle Linux (Oracle)	7	ELSA
Oracle Linux (Oracle)	8	ELSA
Oracle Linux (Oracle)	9	ELSA
Photon OS	3	PHSA
Photon OS	4	PHSA
Photon OS	5	PHSA
Red Hat Enterprise Linux (RHEL)	7	RHSA
Red Hat Enterprise Linux (RHEL)	8	RHSA
Red Hat Enterprise Linux (RHEL)	9	RHSA
Rocky Linux	8	RLSA
Rocky Linux	9	RLSA
SUSE Linux Enterprise Server (SLES)	12.4	SUSE CVE
SUSE Linux Enterprise Server (SLES)	12.5	SUSE CVE

操作系统	版本	供应商安全公告
SUSE Linux Enterprise Server (SLES)	15.3	SUSE CVE
SUSE Linux Enterprise Server (SLES)	15.4	SUSE CVE
SUSE Linux Enterprise Server (SLES)	15.5	SUSE CVE
Ubuntu (Trusty)	14.04 (ESM)	USN, Ubuntu Pro
Ubuntu (Xenial)	16.04 (ESM)	USN, Ubuntu Pro
Ubuntu (Bionic)	18.04 (ESM)	USN, Ubuntu Pro
Ubuntu (Focal)	20.04 (LTS)	USN
Ubuntu (Jammy)	22.04 (LTS)	USN
Ubuntu (Mantic Minotaur)	23.10	USN

支持的编程语言：Amazon ECR 扫描

在扫描 Amazon ECR 存储库中的容器映像时，Amazon Inspector 目前支持以下编程语言：

- C#
- Go
- Java
- JavaScript
- PHP
- Python
- Ruby
- Rust

支持的运行时系统：Amazon Inspector Lambda 标准扫描

在扫描 Lambda 函数以查找第三方软件包中的漏洞时，Amazon Inspector Lambda 标准扫描目前支持以下编程语言：

- Java
  - java8
  - java8.al2
  - java11
  - java17
- Node.js
  - nodejs12.x
  - nodejs14.x
  - nodejs16.x
  - nodejs18.x
  - nodejs20.x
- Python
  - python3.7
  - python3.8
  - python3.9
  - python3.10
  - python3.11
- Go
  - go1.x
- Ruby
  - ruby2.7
  - ruby3.2
- .NET
  - .NET 6

支持的运行时系统：Amazon Inspector Lambda 代码扫描

在扫描 Lambda 函数中是否存在代码漏洞时，Amazon Inspector Lambda 代码扫描目前支持以下编程语言：

- Java
  - java8
  - java8.al2
  - java11
  - java17
- Node.js
  - nodejs12.x
  - nodejs14.x
  - nodejs16.x
  - nodejs18.x
  - nodejs20.x
- Python
  - python3.7
  - python3.8
  - python3.9
  - python3.10
  - python3.11
- Ruby
  - ruby2.7
  - ruby3.2

停产的操作系统

供应商已停止对下表所列操作系统的标准供应商支持。在表格中，已停产列显示了供应商何时停止对操作系统的标准支持。

Amazon Inspector 之前为这些操作系统提供了全面支持，并将继续扫描运行这些实例的亚马逊 EC2 实例和亚马逊 ECR 容器映像。但是，根据供应商政策，这些操作系统不再更新修补程序，而且在许多情

况下，也不再发布新的安全公告。此外，当受影响的操作系统的标准支持期结束时，一些供应商会从他们的信息源中删除现有的安全公告和检测。因此，Amazon Inspector 可能会停止为已知 CVE 生成调查发现。Amazon Inspector 仍为已停产操作系统生成的所有调查发现都应仅供参考。

作为安全方面的最佳实践，同时为了持续保障 Amazon Inspector 的覆盖范围，我们建议您改用最新的、受支持的操作系统版本。

停产操作系统：Amazon EC2 扫描

操作系统	版本	已停产
Amazon Linux (AL1)	2012 年	2021 年 12 月 31 日
CentOS Linux (CentOS)	8	2021 年 12 月 31 日
Debian Server (Stretch)	9	2022 年 6 月 30 日
Fedora	35	2022 年 12 月 13 日
Fedora	36	2023 年 5 月 16 日
Fedora	37	2023 年 12 月 5 日
OpenSUSE	15.3	2022 年 12 月 1 日
OpenSUSE	15.4	2023 年 12 月 7 日
OpenSUSE Leap (SUSE Leap)	15.2	2021 年 12 月 1 日
Oracle Linux (Oracle)	6	2021 年 3 月 1 日
SUSE Linux Enterprise Server (SLES)	12	2019 年 7 月 1 日
SUSE Linux Enterprise Server (SLES)	12.1	2020 年 5 月 31 日
SUSE Linux Enterprise Server (SLES)	12.2	2021 年 3 月 31 日

操作系统	版本	已停产
SUSE Linux Enterprise Server (SLES)	12.3	2022 年 6 月 30 日
SUSE Linux Enterprise Server (SLES)	15	2019 年 12 月 31 日
SUSE Linux Enterprise Server (SLES)	15.1	2021 年 1 月 31 日
SUSE Linux Enterprise Server (SLES)	15.2	2021 年 12 月 31 日
Ubuntu (Groovy)	20.10	2021 年 7 月 22 日
Ubuntu (Hirsute)	21.04	2022 年 1 月 20 日
Ubuntu (Impish)	21.10	2022 年 7 月 31 日
Ubuntu (Kinetic)	22.10	July 20, 2023
Ubuntu (Lunar Lobster)	23.04	January 25, 2024
Windows Server	2012 年	2023 年 10 月 10 日
Windows Server	2012 R2	2023 年 10 月 10 日

停产操作系统：Amazon ECR 扫描

操作系统	版本	已停产
Alpine Linux (Alpine)	3.12	2022 年 5 月 1 日
Alpine Linux (Alpine)	3.13	2022 年 11 月 1 日
Alpine Linux (Alpine)	3.14	May 1, 2023
Alpine Linux (Alpine)	3.15	November 1, 2023

操作系统	版本	已停产
Amazon Linux (AL1)	2012 年	2021 年 12 月 31 日
CentOS Linux (CentOS)	8	2021 年 12 月 31 日
Debian Server (Stretch)	9	2022 年 6 月 30 日
Fedora	35	2022 年 12 月 13 日
Fedora	36	2023 年 5 月 16 日
OpenSUSE	15.3	2022 年 12 月 1 日
OpenSUSE	15.4	December 7, 2023
OpenSUSE Leap (SUSE Leap)	15.2	2021 年 12 月 1 日
Oracle Linux (Oracle)	6	2021 年 3 月 1 日
SUSE Linux Enterprise Server (SLES)	12	2019 年 7 月 1 日
SUSE Linux Enterprise Server (SLES)	12.1	2020 年 5 月 31 日
SUSE Linux Enterprise Server (SLES)	12.2	2021 年 3 月 31 日
SUSE Linux Enterprise Server (SLES)	12.3	2022 年 6 月 30 日
SUSE Linux Enterprise Server (SLES)	15	2019 年 12 月 31 日
SUSE Linux Enterprise Server (SLES)	15.1	2021 年 1 月 31 日
SUSE Linux Enterprise Server (SLES)	15.2	2021 年 12 月 31 日

操作系统	版本	已停产
Ubuntu (Groovy)	20.10	2021 年 7 月 22 日
Ubuntu (Hirsute)	21.04	2022 年 1 月 20 日
Ubuntu (Impish)	21.10	2022 年 7 月 31 日
Ubuntu (Kinetic)	22.10	July 20, 2023
Ubuntu (Lunar Lobster)	23.04	January 25, 2024

停用 Amazon Inspector

可使用 Amazon Inspector 控制台或 API 在任意 AWS 区域停用 Amazon Inspector。请按照本主题末尾的说明停用 Amazon Inspector。如果您停用对 AWS 账户的所有 Amazon Inspector 扫描，则该账户的 Amazon Inspector 将自动停用。有关停用不同资源的扫描类型的信息，请参阅使用 Amazon Inspector 自动扫描资源。

停用某个账户的 Amazon Inspector 后，该账户在该地区的所有扫描类型都将被停用。此外，该账户在该地区的所有 Amazon Inspector 扫描设置、禁止规则以及筛选条件和调查发现都将被删除。

当您在该地区的账户停用 Amazon Inspector 时，使用 Amazon Inspector 不会向您收费。在停用 Amazon Inspector 之后，您可以选择稍后将其重新激活。

> **Note**
>
> 在停用 Amazon Inspector 之前，我们建议您先导出调查发现。有关更多信息，请参阅从 Amazon Inspector 导出调查发现报告。

停用 Amazon Inspector Amazon EC2 扫描时，Amazon Inspector 使用的以下 SSM 关联将被删除：

- `InspectorDistributor-do-not-delete`
- `InspectorInventoryCollection-do-not-delete`
- `InvokeInspectorSsmPlugin-do-not-delete`。 此外，通过此关联安装的 Amazon Inspector SSM 插件已从您的所有Windows主机上移除。有关更多信息，请参阅扫描 Windows 实例。

先决条件

根据您的账户类型，在停用 Amazon Inspector 之前，您可能需要采取以下额外步骤：

- 如果您有一个独立的 Amazon Inspector 账户，则可以随时将其停用。
- 如果您是 Amazon Inspector 多账户环境中的成员账户，则无法停用自己的服务。您必须联系贵组织的委托管理员才能停用您的服务。
- 如果您是委托管理员，则必须先解除所有成员账户的关联，然后才能停用 Amazon Inspector。有关更多信息，请参阅在 Amazon Inspector 中取消成员账户的关联。

> **Note**
>
> 取消关联账户并不会停用该账户的 Amazon Inspector，相反，取消关联的成员账户会变成独立账户。

> **Note**
>
> 当您以委托管理员身份停用 Amazon Inspector 时，贵组织的自动激活功能将停用。

停用 Amazon Inspector

Console

要停用 Amazon Inspector

1. 打开 Amazon Inspector 控制台，网址为 https://console.aws.amazon.com/inspector/v2/home。
2. 使用页面右上角的 AWS 区域 选择器，选择要停用 Amazon Inspector 的区域。
3. 在导航窗格中，选择常规设置。
4. 选择停用 Inspector。
5. 出现确认提示时，在文本框中输入停用，然后选择停用 Inspector。
6. （推荐）在您要停用 Amazon Inspector 的每个区域中重复这些步骤。

API

运行“禁用 API”操作。在请求中，提供您要停用的账户 ID，以及用于 `resourceTypes` 的 `EC2, ECR, LAMBDA`，以停用所有扫描，这将停用该账户。

Amazon Inspector 配额

您的 AWS 账户在每个区域具有以下 Amazon Inspector 配额。

资源	默认值	注释
抑制规则	500	每个区域每个 AWS 账户可保存的最大抑制规则数量。 您无法请求提高配额。
Amazon EC2 网络调查发现	10000	每个 AWS 账户的最大 Amazon EC2 网络调查发现数量。 您无法请求提高配额。
成员账户	10000	与 Amazon Inspector 委派管理员账户关联的成员账户的最大数量。此限制基于 AWS Organizations，请参阅 AWS Organizations 配额。
CIS 扫描配置	500	CIS 扫描配置的最大数量。 您无法请求提高配额。

有关与 Amazon Inspector Classic 相关的配额列表，请参阅 AWS 一般参考 中的 Amazon Inspector 服务配额。

有关与组织相关的配额列表，请参阅 AWS 一般参考 中的组织服务配额。

区域和端点

适用于 Amazon EC2 的 Amazon Inspector 无代理扫描目前为预览版。您对无代理 Amazon EC2 扫描功能的使用受 AWS 服务条款第 2 部分（“测试版和预览”）的约束。

要查看 Amazon Inspector 的可用 AWS 区域，请参阅 Amazon Web Services 一般参考 中的 Amazon Inspector 端点。

Amazon Inspector Scan API 的端点

下表显示了在调用 Amazon Inspector Scan API 时可以使用的区域端点。使用该 API 时，您必须提供端点及其对应的区域，即您当前已通过身份验证的 AWS 区域。

Amazon Inspector 扫描端点的命名约定是 `inspector-scan.region.amazonaws.com`。例如，如果您在 `us-west-2` 中进行了身份验证，则将使用端点 `inspector-scan.us-west-2.amazonaws.com` 来调用 `inspector-scan` API。

区域名称	区域	端点	协议
美国东部（俄亥俄州）	us-east-2	inspector-scan.us-east-2.amazonaws.com inspector-scan-fips.us-east-2.amazonaws.com	HTTPS
美国东部（弗吉尼亚州北部）	us-east-1	inspector-scan.us-east-1.amazonaws.com inspector-scan-fips.us-east-1.amazonaws.com	HTTPS

区域名称	区域	端点	协议
美国西部（北加利福尼亚）	us-west-1	inspector-scan.us-west-1.amazonaws.com inspector-scan-fips.us-west-1.amazonaws.com	HTTPS
美国西部（俄勒冈州）	us-west-2	inspector-scan.us-west-2.amazonaws.com inspector-scan-fips.us-west-2.amazonaws.com	HTTPS
非洲（开普敦）	af-south-1	inspector-scan.af-south-1.amazonaws.com	HTTPS
亚太地区（香港）	ap-east-1	inspector-scan.ap-east-1.amazonaws.com	HTTPS
亚太地区（雅加达）	ap-southeast-3	inspector-scan.ap-southeast-3.amazonaws.com	HTTPS
亚太地区（孟买）	ap-south-1	inspector-scan.ap-south-1.amazonaws.com	HTTPS
亚太地区（大阪）	ap-northeast-3	inspector-scan.ap-northeast-3.amazonaws.com	HTTPS

区域名称	区域	端点	协议
亚太地区（首尔）	ap-northeast-2	inspector-scan.ap-northeast-2.amazonaws.com	HTTPS
亚太地区（新加坡）	ap-southeast-1	inspector-scan.ap-southeast-1.amazonaws.com	HTTPS
亚太地区（悉尼）	ap-southeast-2	inspector-scan.ap-southeast-2.amazonaws.com	HTTPS
亚太地区（东京)	ap-northeast-1	inspector-scan.ap-northeast-1.amazonaws.com	HTTPS
加拿大（中部）	ca-central-1	inspector-scan.ca-central-1.amazonaws.com	HTTPS
欧洲地区（法兰克福）	eu-central-1	inspector-scan.eu-central-1.amazonaws.com	HTTPS
欧洲地区（爱尔兰）	eu-west-1	inspector-scan.eu-west-1.amazonaws.com	HTTPS
欧洲地区（伦敦）	eu-west-2	inspector-scan.eu-west-2.amazonaws.com	HTTPS
欧洲地区（米兰）	eu-south-1	inspector-scan.eu-south-1.amazonaws.com	HTTPS

区域名称	区域	端点	协议
欧洲地区（巴黎）	eu-west-3	inspector-scan.eu-west-3.amazonaws.com	HTTPS
欧洲地区（斯德哥尔摩）	eu-north-1	inspector-scan.eu-north-1.amazonaws.com	HTTPS
欧洲（苏黎世）	eu-central-2	inspector-scan.eu-central-2.amazonaws.com	HTTPS
中东（巴林）	me-south-1	inspector-scan.me-south-1.amazonaws.com	HTTPS
南美洲（圣保罗）	sa-east-1	inspector-scan.sa-east-1.amazonaws.com	HTTPS
AWS GovCloud（美国东部）	us-gov-east-1	inspector-scan.us-gov-east-1.amazonaws.com inspector-scan-fips.us-gov-east-1.amazonaws.com	HTTPS
AWS GovCloud（美国西部）	us-gov-west-1	inspector-scan.us-gov-west-1.amazonaws.com inspector-scan-fips.us-gov-west-1.amazonaws.com	HTTPS

特定于区域的功能可用性

本节介绍按 AWS 区域划分的 Amazon Inspector 可用功能。

适用于 Amazon EC2 的无代理 EC2 扫描功能可用的区域

下表列出了目前支持适用于 Amazon EC2 的无代理扫描功能的 AWS 区域。

区域名称	区域代码
美国东部（弗吉尼亚州北部）	us-east-1
美国西部（俄勒冈州）	us-west-2
欧洲地区（爱尔兰）	eu-west-1

Lambda 代码扫描区域

下表列出了目前支持 Lambda 代码扫描的 AWS 区域。

区域名称	区域代码
美国东部（弗吉尼亚州北部）	us-east-1
美国西部（俄勒冈州）	us-west-2
美国东部（俄亥俄州）	us-east-2
亚太地区（悉尼）	ap-southeast-2
亚太地区（东京）	ap-northeast-1
欧洲地区（法兰克福）	eu-central-1
欧洲地区（爱尔兰）	eu-west-1
欧洲地区（伦敦）	eu-west-2
欧洲地区（斯德哥尔摩）	eu-north-1

区域名称	区域代码
亚太地区（新加坡）	ap-southeast-1

AWS GovCloud (US) 区域

有关最新信息，请参阅 AWS GovCloud (US) 用户指南中的 Amazon Inspector。

《Amazon Inspector 用户指南》的文档历史记录

下表列出了自 Amazon Inspector 上一次发布以来对文档所做的重要更改。要获得本文档的更新通知，您可以订阅 RSS 源。

变更	说明	日期
更新了功能	Amazon Inspector 将已结调查结果的保留期从 30 天更新为 7 天。有关更多信息，请参阅了解 Amazon Inspector 中的调查结果。	2024年2月12日
更新了功能	Amazon Inspector 在 AmazonInspector2ServiceRolePolicy 策略中增加了一个新语句。新声明允许 Amazon Inspector 为您的实例启动 CIS 扫描。	2024 年 1 月 23 日
新策略	Amazon Inspector 添加了一项新的AmazonInspector2ManagedCisPolicy策略，即策略，您可以将其用作实例配置文件的一部分，以允许对实例进行 CIS 扫描。	2024 年 1 月 23 日
新特征	现在，当你拉取容器镜像时，Amazon In spector 将刷新容器镜像的 ECR 重新扫描持续时间。要根据推送或拉取日期更改重新扫描持续时间，请参阅配置 ECR 重新扫描持续时间。	2024 年 1 月 23 日
新特征	Amazon Inspector 现在可以对 EC2 实例运行互联网安全中心	2024 年 1 月 23 日

	(CIS) 扫描。有关更多信息，请参阅 Amazon Inspector CIS 扫描。	
新特征	Amazon Inspector 现在可以扫描 CI/CD 管道中的容器映像。有关更多信息，请参阅 CI/CD 与 Amazon Inspector 的集成。	2023 年 11 月 30 日
新策略	Amazon Inspector 添加了一项新策略，允许 Amazon Inspector 对 EC2 实例中的 Amazon EBS 快照进行无代理扫描。有关该策略的更多信息，请参阅无代理扫描。	2023 年 11 月 27 日
新特征	Amazon Inspector 现在支持通过无代理扫描在没有 SSM 代理的情况下扫描支持的 Linux Amazon EC2 实例。有关更多信息，请参阅无代理扫描。	2023 年 11 月 27 日
新的支持的资源	Amazon Inspector 现在支持扫描 MacOS Amazon EC2 实例。请参阅支持的操作系统：Amazon EC2 扫描，了解支持的 MacOS 版本。	2023 年 10 月 5 日
新区域	Amazon Inspector 现已在亚太地区（雅加达）、非洲（开普敦）、亚太地区（大阪）和欧洲地区（苏黎世）发布。	2023 年 9 月 29 日
新特征	现在，您可以使用排除标签将 EC2 实例从 Amazon Inspector 扫描中排除。	2023 年 9 月 14 日

新特征	Amazon Inspector 增加了新的权限，允许 Amazon Inspector 扫描属于 Elastic Load Balancing 目标组的 Amazon EC2 实例的网络配置。	2023 年 8 月 31 日
新特征	Amazon Inspector 现在可为程序包漏洞调查发现提供漏洞情报详细信息。	2023 年 7 月 31 日
更新了功能	Amazon Inspector 增加了新的权限，允许只读用户为其资源导出软件材料清单 (SBOM)。	2023 年 6 月 29 日
新特征	现在，您可以导出 Amazon Inspector 正在扫描的资源的 SBOM。	2023 年 6 月 13 日
新特征	Lambda 代码扫描现已全面推出。新增功能允许您对 Lambda 代码扫描结果中发现的代码进行加密。此外，Lambda 代码扫描现在还可提供代码修复重写建议。	2023 年 6 月 13 日
更新了功能	Amazon Inspector 在 AmazonInspector2ReadOnlyAccess 策略中增加了一个新语句。新语句允许只读用户检索其账户的 Lambda 代码扫描状态和结果的详细信息。	2023 年 5 月 2 日
新特征	Amazon Inspector 增加了漏洞数据库搜索功能，支持检查 Amazon Inspector 是否涵盖了特定的 CVE。	2023 年 5 月 1 日

更新了功能	Amazon Inspector 在 AmazonInspector2ServiceRolePolicy 策略中增加了新的权限，允许 Amazon Inspector 在您激活 Lambda 扫描时在您的账户中创建 AWS CloudTrail 服务相关通道。这样，Amazon Inspector 就可以监控您账户中的 CloudTrail 事件。	2023 年 4 月 30 日
更新了功能	Amazon Inspector 在 AmazonInspector2FullAccess 策略中增加了一个新语句。新语句允许用户从 Lambda 代码扫描中检索代码漏洞调查发现的详细信息。	2023 年 4 月 17 日
更新了功能	Amazon Inspector 在 AmazonInspector2ServiceRolePolicy 策略中增加了一个新语句。新声明允许 Amazon Inspector 向亚马逊 EC2 Systems Manager 发送有关您为亚马逊 EC2 深度检查定义的自定义路径的信息。	2023 年 4 月 17 日
新特征	Amazon Inspector 以 Amazon Inspector 深度检查的形式增加了对 Linux EC2 实例的额外支持，它可以扫描您的实例中是否存在应用程序编程语言包中的软件包漏洞。	2023 年 4 月 17 日

更新了功能	Amazon Inspector 在 AmazonInspector2ServiceRolePolicy 策略中增加了一个新语句。新的语句允许 Amazon Inspector 请求对AWS Lambda 函数中的开发者代码进行扫描，并从亚马逊 CodeGuru 安全部门接收扫描数据。此外，Amazon Inspector 还增加了审查 IAM policy 的权限。Amazon Inspector 使用这些信息扫描 Lambda 函数中是否存在代码漏洞。	2023 年 2 月 28 日
新特征	Amazon Inspector 以 Lambda 代码扫描的形式增加了对 Lambda 函数的额外支持，这会扫描 Lambda 函数的开发者代码中是否存在安全漏洞。	2023 年 2 月 28 日
更新了功能	Amazon Inspector 在 AmazonInspector2ServiceRolePolicy 策略中增加了一个新语句。新语句允许 Amazon Inspector 检索 CloudWatch 有关上次调用AWS Lambda 函数的时间的信息。使用这些信息将扫描重点放在您环境中过去 90 天内处于活动状态的 Lambda 函数上。	2023 年 2 月 20 日

更新了功能	Amazon Inspector 在 AmazonInspector2ServiceRolePolicy 策略中增加了一个新语句。新语句允许 Amazon Inspector 检索有关 AWS Lambda 函数的信息。Amazon Inspector 使用这些信息扫描 Lambda 函数中是否存在安全漏洞。	2022 年 11 月 28 日
新特征	Amazon Inspector 增加了对 AWS Lambda 函数扫描功能的支持。	2022 年 11 月 28 日
更新的内容	增加了将调查发现报告从 Amazon Inspector 导出到 Amazon Simple Storage Service (Amazon S3) 存储桶的程序、策略示例和提示。	2022 年 10 月 14 日
新增内容	增加了有关使用 Amazon Inspector 控制台评测 AWS 环境的 Amazon Inspector 覆盖率的信息。这些信息包括环境中各个资源的状态值说明。	2022 年 10 月 7 日
新特征	Amazon Inspector 现在提供有关如何修复程序包漏洞的更多详细信息。调查发现详细信息增加了新字段。新字段提供了是否可以通过程序包更新获得修复的上下文信息。如果有可用的修复方法，则调查发现的建议的补救措施部分会显示您可以运行的修复命令。	2022 年 9 月 2 日

更新了功能	Amazon Inspector 在 AmazonInspector2ServiceRolePolicy 策略中增加了一个新操作。此新操作允许 Amazon Inspector 描述 SSM 关联的执行情况。Amazon Inspector 还增加了额外的资源范围，允许 Amazon Inspector 使用 `AmazonInspector2` 拥有的 SSM 文档创建、更新、删除和启动 SSM 关联。	2022 年 8 月 31 日
新特征	Amazon Inspector 现在支持对 Windows 实例进行扫描。Amazon Inspector 现在可以扫描运行支持的 Windows 操作系统的 SSM 托管实例。Windows主机扫描由 Amazon Inspector SSM 插件执行，该插件是通过 Amazon Inspector 自动创建的新 SSM 关联安装和调用的。	2022 年 8 月 31 日
更新了功能	Amazon Inspector 更新了 AmazonInspector2ServiceRolePolicy 策略的资源范围，允许 Amazon Inspector 收集其他 AWS 分区中的软件清单。	2022 年 8 月 12 日
更新了功能	在 AmazonInspector2ServiceRolePolicy 策略中，Amazon Inspector 重组了操作的资源范围，允许 Amazon Inspector 创建、删除和更新 SSM 关联。	2022 年 8 月 10 日

新特征	Amazon Inspector 现在支持更改 ECR 自动重新扫描持续时间设置。Amazon ECR 自动重新扫描持续时间设置决定了 Amazon Inspector 持续监控推送到存储库的映像的时间。当映像存在时间超过扫描持续时间时，Amazon Inspector 将不再扫描该映像并将关闭它的所有现有结果。所有新账户的 ECR 自动重新扫描持续时间都将自动设置为生命周期。之前创建的账户的 ECR 自动重新扫描持续时间为 30 天，但现在您可以将扫描持续时间设置为 30 天、180 天或生命周期。	2022 年 6 月 25 日
新功能	Amazon Inspector 增加了一项新的 AWS 托管策略，即 AmazonInspector2ReadOnlyAccess 策略，允许以只读方式访问 Amazon Inspector 功能。	2022 年 1 月 21 日
正式发布	这是 Amazon Inspector 用户指南的第一个公开发行版。	2021 年 11 月 29 日

AWS 术语表

有关最新的 AWS 术语，请参阅《AWS 词汇表参考》中的 AWS 词汇表。

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。