Amazon Inspector

User Guide

Version Latest

Amazon Inspector: User Guide

Table of Contents

What is Amazon Inspector?

Amazon Inspector tests the network accessibility of your Amazon EC2 instances and the security state of your applications that run on those instances. Amazon Inspector assesses applications for exposure, vulnerabilities, and deviations from best practices. After performing an assessment, Amazon Inspector produces a detailed list of security findings that is organized by level of severity.

With Amazon Inspector, you can automate security vulnerability assessments throughout your development and deployment pipelines or for static production systems. This allows you to make security testing a regular part of development and IT operations.

Amazon Inspector also offers predefined software called an *agent* that you can optionally install in the operating system of the EC2 instances that you want to assess. The agent monitors the behavior of the EC2 instances, including network, file system, and process activity. It also collects a wide set of behavior and configuration data (telemetry).

Important
AWS doesn't guarantee that following the provided recommendations will resolve every potential security issue. The findings generated by Amazon Inspector depend on your choice of rules packages included in each assessment template, the presence of non-AWS components in your system, and other factors. You are responsible for the security of applications, processes, and tools that run on AWS services. For more information, see the AWS Shared Responsibility Model for security.

Note
AWS is responsible for protecting the global infrastructure that runs the services offered in the AWS Cloud. This infrastructure consists of the hardware, software, networking, and facilities that run AWS services. AWS provides several reports from third-party auditors who have verified our compliance with a variety of computer security standards and regulations. For more information, see AWS Cloud Compliance.

For information about Amazon Inspector terminology, see Amazon Inspector terminology and concepts (p. 2).

Benefits of Amazon Inspector

Here are some of the main benefits of Amazon Inspector:

- **Integrate automated security checks into your regular deployment and production processes** – Assess the security of your AWS resources for forensics, troubleshooting, or active auditing purposes. Run the assessments during the development process, or run them in a stable production environment.
- **Find application security issues** – Automate the security assessment of your applications and proactively identify vulnerabilities. This allows you to develop and iterate on new applications quickly, and assess compliance with best practices and policies.
- **Gain a deeper understanding of your AWS resources** – Stay informed about the activity and configuration data of your AWS resources by reviewing the findings that Amazon Inspector produces.

Features of Amazon Inspector

Here are some of the main features of Amazon Inspector:

- **Configuration scanning and activity monitoring engine** – Amazon Inspector provides an agent that analyzes system and resource configuration. It also monitors activity to determine what an assessment

target looks like, how it behaves, and its dependent components. The combination of this telemetry provides a complete picture of the target and its potential security or compliance issues.

- **Built-in content library** – Amazon Inspector includes a built-in library of rules and reports. These include checks against best practices, common compliance standards, and vulnerabilities. The checks include detailed recommended steps for resolving potential security issues.
- **Automation through an API** – Amazon Inspector can be fully automated through an API. This allows you to incorporate security testing into the development and design process, including selecting, executing, and reporting the results of those tests.

Amazon Inspector pricing

Amazon Inspector pricing is based on the number of EC2 instances included in each assessment and the rules packages used in those assessments. For more information about Amazon Inspector pricing, see Amazon Inspector Pricing.

Accessing Amazon Inspector

You can work with the Amazon Inspector service in any of the following ways:

Amazon Inspector Console

Sign in to the AWS Management Console and open the Amazon Inspector console at https://console.aws.amazon.com/inspector/.

The console is a browser-based interface that lets you access and use the Amazon Inspector service.

AWS SDKs

AWS provides software development kits (SDKs) that consist of libraries and sample code for various programming languages and platforms. These include Java, Python, Ruby, .NET, iOS, Android, and more. The SDKs provide a convenient way to create programmatic access to the Amazon Inspector service. For information about the AWS SDKs, including how to download and install them, see Tools for Amazon Web Services.

Amazon Inspector HTTPS API

You can access Amazon Inspector and AWS programmatically by using the Amazon Inspector HTTPS API, which lets you issue HTTPS requests directly to the service. For more information, see the Amazon Inspector API Reference.

AWS Command Line Tools

You can use the AWS command line tools to run commands at your system's command line to perform Amazon Inspector tasks. The command line tools are also useful if you want to build scripts that perform AWS tasks. For more information, see the Amazon Inspector AWS Command Line Interface.

Amazon Inspector terminology and concepts

As you get started with Amazon Inspector, you can benefit from learning about its key concepts.

Amazon Inspector agent

A software agent that you can install on the EC2 instances that are included in the assessment target. The agent collects a wide set of configuration data (telemetry). For more information, see Amazon Inspector agents (p. 34).

Assessment run

The process of discovering potential security issues through the analysis of your assessment target's configuration against specified rules packages. During an assessment run, Amazon Inspector monitors, collects, and analyzes configuration data (telemetry) from resources within the specified target. Next, Amazon Inspector analyzes the data and compares it against a set of security rules packages that are specified in the assessment template used during the assessment run. A completed assessment run produces a list of findings, which are potential security issues of various levels of severity. For more information, see Amazon Inspector assessment templates and assessment runs (p. 62).

Assessment target

In the context of Amazon Inspector, a collection of AWS resources that work together as a unit to help you accomplish your business goals. Amazon Inspector evaluates the security state of the resources that constitute the assessment target.

> **Important**
> Currently, your Amazon Inspector assessment targets can consist only of EC2 instances. For more information, see Amazon Inspector service limits (p. 3)

To create an Amazon Inspector assessment target, you must first tag your EC2 instances with key-value pairs of your choice. Next, you can create a view of these tagged EC2 instances that have common keys or common values. For more information, see Amazon Inspector assessment targets (p. 48).

Assessment template

A configuration that is used during your assessment run. The template includes the following:

- Rules packages that Amazon Inspector uses to evaluate your assessment target
- Amazon SNS topics that you want Amazon Inspector to send notifications to about assessment run states and findings
- Tags (key-value pairs) that you can assign to findings that are generated by the assessment run
- The duration of the assessment run

Finding

A potential security issue that Amazon Inspector discovers during an assessment run of the specified target. Findings are displayed in the Amazon Inspector console or retrieved through the API. They contain both a detailed description of the security issue and a recommendation on how to fix it. For more information, see Amazon Inspector findings (p. 68).

Rule

In the context of Amazon Inspector, a security check performed during an assessment run. When a rule detects a potential security issue, Amazon Inspector generates a finding that describes the issue.

Rules package

In the context of Amazon Inspector, a collection of rules. A rules package corresponds to a security goal that you might have. You can specify your security goal by selecting the appropriate rules package when you create an Amazon Inspector assessment template. For more information, see Amazon Inspector rules packages and rules (p. 51).

Telemetry

Installed package information and software configuration for an EC2 instance. Amazon Inspector collects the data during an assessment run.

Amazon Inspector service limits

The following table shows the Amazon Inspector limits for an AWS account.

Important
Currently, your assessment targets can consist only of EC2 instances.

The following are Amazon Inspector limits per AWS account per region:

Resource	Default Limit	Comments
Instances in running assessments	500	The maximum number of EC2 instances that can be included across all running assessments per account per region.
Assessment runs	50000	The maximum number of assessment runs that you can create per account per region. You can have multiple assessment runs happening at the same time as long as the assessment targets used for these runs do not contain overlapping EC2 instances.
Assessment Templates	500	The maximum number of assessment templates that you can have at any given time per account per region.
Assessment Targets	50	The maximum number of assessment targets that you can have at any given time per account per region.

Unless otherwise noted, these limits can be increased upon request by contacting the AWS Support Center.

Amazon Inspector supported operating systems and Regions

This chapter provides information about the operating systems and AWS Regions that Amazon Inspector supports.

Important
Currently, Amazon Inspector assessment targets can consist only of EC2 instances. You can run an agentless assessment with the Network Reachability (p. 52) rules package on any EC2 instances regardless of operating system.

For information about the Amazon Inspector rules packages that are available across supported operating systems, see Amazon Inspector rules packages for supported operating systems (p. 78).

Topics

Supported Linux-based operating systems for the Amazon Inspector agent

You can use the Amazon Inspector agent on 64-bit x86 and Arm EC2 instances. The agent is compatible with the following versions of Linux-based operating systems:

- **64-bit x86 instances**
  - Amazon Linux 2
  - Amazon Linux (2018.03, 2017.09, 2017.03, 2016.09, 2016.03, 2015.09, 2015.03, 2014.09, 2014.03, 2013.09, 2013.03, 2012.09, 2012.03)
  - Ubuntu (20.04 LTS, 18.04 LTS, 16.04 LTS, 14.04 LTS)
  - Debian (10.x, 9.0 - 9.5, 8.0 - 8.7)
  - Red Hat Enterprise Linux (8.x, 7.2 - 7.x, 6.2 - 6.9)
  - CentOS (7.2 - 7.x, 6.2 - 6.9)
- **Arm instances**
  - Amazon Linux 2
  - Red Hat Enterprise Linux (7.6 - 7.x)
  - Ubuntu (18.04 LTS, 16.04 LTS)

Supported Windows-based operating systems for the Amazon Inspector agent

You can use the Amazon Inspector agent only on EC2 instances that run the 64-bit version of the following Windows-based operating systems:

- Windows Server 2019 Base
- Windows Server 2016 Base
- Windows Server 2012 R2
- Windows Server 2012
- Windows Server 2008 R2

Supported AWS Regions

Amazon Inspector is supported in the following AWS Regions:

- US East (Ohio) us-east-2
- US East (N. Virginia) us-east-1
- US West (N. California) us-west-1
- US West (Oregon) us-west-2
- Asia Pacific (Mumbai) ap-south-1
- Asia Pacific (Seoul) ap-northeast-2

- Asia Pacific (Sydney) ap-southeast-2
- Asia Pacific (Tokyo) ap-northeast-1
- Europe (Frankfurt) eu-central-1
- Europe (Ireland) eu-west-1
- Europe (London) eu-west-2
- Europe (Stockholm) eu-north-1
- AWS GovCloud (US-East) gov-us-east-1
- AWS GovCloud (US-West) gov-us-east-2

Note
The Network Reachability (p. 52) rules package is not available in the AWS GovCloud (US) Regions.

Getting started with Amazon Inspector

This tutorial shows you how to set up Amazon Inspector and get started by creating and running your first assessment.

Important
To use Amazon Inspector, you must have an AWS account. When you sign up for AWS, your account is automatically signed up for all services in AWS, including Amazon Inspector. If you don't have an AWS account, use the following procedure to create one.

To sign up for AWS

1. Open https://portal.aws.amazon.com/billing/signup.
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

Topics

Prerequisites for using Amazon Inspector

When you launch the Amazon Inspector console for the first time, choose **Get Started** and complete the following prerequisite tasks. You must complete these tasks before you can perform an Amazon Inspector assessment run:

- You must have at least one Amazon EC2 instance running in your AWS environment to run an Amazon Inspector assessment. For information about launching EC2 instances, see the Amazon Elastic Compute Cloud Documentation.
- In most cases, the Amazon Inspector agent must be running on each EC2 instance in your assessment target. For information about installing an agent, see Installing Amazon Inspector agents (p. 36). Alternatively, you can use Systems Manager Run Command to install the agent on your Amazon EC2 instances. For more information about Amazon Inspector agents, see Amazon Inspector agents (p. 34).

One-click setup

The following procedure shows you how to create and run an automatic assessment using a pre-built template and pre-defined scheduling parameters (once a week or one time only) on all available EC2 instances in the current AWS account and Region.

1. Sign in to the AWS Management Console and open the Amazon Inspector console at https://console.aws.amazon.com/inspector/.
2. On the **Welcome** page, choose the type of assessment that you would like to run. **Network Assessments** analyze the network configurations of your AWS environment for vulnerabilities, and do not require an Amazon Inspector agent. **Host Assessments** analyze the on-host software and configurations of your EC2 instances for vulnerabilities, and requires an agent to be installed on the EC2 instances.

   Choose either **Run weekly (recommended)** or **Run once**. As soon as you make your choice, the service automatically creates the assessment for you. Specifically, the service does the following:

   a. Creates a service-linked role (p. 29).

      Note
      To identify the EC2 instances that are specified in the assessment targets, Amazon Inspector needs to enumerate your EC2 instances and tags. Amazon Inspector gets access to these resources in your AWS account through a service-linked role called `AWSServiceRoleForAmazonInspector`. For more information about service-linked roles, see Using service-linked roles for Amazon Inspector (p. 29) and Using Service-Linked Roles.

   b. If applicable, installs an Amazon Inspector agent (p. 34) on all available Amazon EC2 instances in your AWS account and AWS Region.

      Note
      The service installs an Amazon Inspector agent only on those EC2 instances that allow AWS Systems Manager Run Command. To use this option, make sure that all of your EC2 instances in the current AWS account and AWS Region have the SSM Agent installed and have an IAM role that allows Run Command. For more information, see Installing the agent on multiple EC2 instances using the Systems Manager Run Command (p. 37).

   c. Adds those instances to an assessment target (p. 48).

   d. Includes that target in an assessment template (p. 62) with a standardized set of rules packages.

   e. Runs the assessment weekly or only once, depending on whether you chose **Run weekly (recommended)** or **Run once**.
3. In the **Confirmation** dialog box, choose **OK**. Amazon Inspector automatically runs your assessment.

Advanced setup

The following procedure shows you how to choose specific Amazon EC2 instances, rules packages, and scheduling parameters to include in an assessment target and template.

1. On the **Welcome** page, choose **Advanced setup**.
2. On the **Define an assessment target** page, enter the name of your assessment target.
3. For **All Instances**, you can keep the check box selected to include all EC2 instances in your AWS account and Region in the assessment target. If you want to choose which EC2 instances to include, clear the **All Instances** check box, and enter the **Key** and **Value** tags that are associated with the target EC2 instances. For more information about tagging your EC2 instances, see Tagging Your Amazon EC2 Resources.
4. For **Install Agents**, you can keep the check box selected by default if your instances allows System Manager Run Command. The service installs an Amazon Inspector agent on all EC2 instances in the assessment target that allow System Manager Run Command. To use this option, make sure that all of your EC2 instances in the current AWS account and AWS region have the SSM Agent installed and have an IAM role that allows Run Command. For more information, see Installing the agent

on multiple EC2 instances using the Systems Manager Run Command (p. 37). If you want to manually install the agent, see Installing Amazon Inspector Agents (p. 36).

5. Choose **Next**.
6. On the **Define an assessment template** page, enter the name of your assessment template.
7. For **Rules packages**, choose the rules packages to include in the assessment template. For more information about rules packages, see Amazon Inspector Rules Packages and Rules (p. 51).
8. For **Duration**, choose the duration of your assessment run.
9. For **Assessment Schedule**, you can set a schedule for recurring assessment runs.
10. Choose **Next**.
11. On the **Review** page, review your choices for the assessment target and template. If you are satisfied with the configuration, choose **Create**. If you set an assessment schedule for your assessment template, the assessment automatically runs after you choose **Create**.

    Note
    To identify the EC2 instances that are specified in the assessment targets, Amazon Inspector needs to enumerate your EC2 instances and tags. Amazon Inspector gets access to these resources in your AWS account through a service-linked role called `AWSServiceRoleForAmazonInspector`. For more information about service-linked roles, see Using service-linked roles for Amazon Inspector (p. 29) and Using Service-Linked Roles.
12. If you didn't set up an assessment schedule, navigate to your assessment template through the console, and then choose **Run**.
13. To track the progress of the assessment run, in the navigation pane of the console, choose **Assessment runs**, and then choose **Findings**. For more information about findings, see *Amazon Inspector findings* (p. 68).

Tutorials for Amazon Inspector

The following tutorials show you how to perform Amazon Inspector assessment runs on the Red Hat Enterprise Linux and Ubuntu operating systems.

Tutorials

Amazon Inspector tutorial - Red Hat Enterprise Linux

Before you follow the instructions in this tutorial, we recommend that you get familiar with the Amazon Inspector terminology and concepts (p. 2).

This tutorial shows how to use Amazon Inspector to analyze the behavior of an EC2 instance that runs the Red Hat Enterprise Linux 7.5 operating system. It provides step-by-step instructions on how to navigate the Amazon Inspector workflow. The workflow includes preparing Amazon EC2 instances, running an assessment template, and performing the recommended security fixes generated in the assessment's findings. If you are a first-time user and would like to set up and run an Amazon Inspector assessment with one click, see Creating a Basic Assessment (p. 7).

Topics

Step 1: Set up an Amazon EC2 instance to use with Amazon Inspector

For this tutorial, create one EC2 instance that runs Red Hat Enterprise Linux 7.5, and tag it using the **Name** key and a value of **`InspectorEC2InstanceLinux`**.

> **Note**
> For more information about tagging EC2 instances, see Resources and Tags.

Step 2: Modify your Amazon EC2 instance

For this tutorial, you modify your target EC2 instance to expose it to the potential security issue CVE-2018-1111. For more information, see https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-1111 and Common vulnerabilities and exposures (p. 54).

Connect to your instance, **`InspectorEC2InstanceLinux`**, and run the following command:

```
sudo yum install dhclient-12:4.2.5-68.el7
```

For instructions on how to connect to an EC2 instance, see Connect to Your Instance in the *Amazon EC2 User Guide*.

Step 3: Create an assessment target and install an agent on the EC2 instance

Amazon Inspector uses assessment targets to designate the AWS resources that you want to evaluate.

To create an assessment target and install an agent on an EC2 instance

1. Sign in to the AWS Management Console and open the Amazon Inspector console at https://console.aws.amazon.com/inspector/.
2. In the navigation pane, choose **Assessment targets**, and then choose **Create**.

   Do the following:

   a. For **Name**, enter the name for your assessment target.

      For this tutorial, enter **`MyTargetLinux`**.

   b. For **Use Tags**, choose the EC2 instances that you want to include in this assessment target by entering values for the **Key** and **Value** fields.

      For this tutorial, choose the EC2 instance that you created in the preceding step by entering **`Name`** in the **Key** field and **`InspectorEC2InstanceLinux`** in the **Value** field.

      To include all EC2 instances in your AWS account and Region in the assessment target, select the **All Instances** check box.

   c. Choose **Save**.

   d. Install an Amazon Inspector agent on your tagged EC2 instance. To install an agent on all EC2 instances included in an assessment target, select the **Install Agents** check box.

      Note
      You can also install the Amazon Inspector agent using the AWS Systems Manager Run Command (p. 37). To install the agent on all instances in the assessment target, you can specify the same tags that you used when creating the assessment target. Or you can install the Amazon Inspector agent on your EC2 instance manually. For more information, see Installing Amazon Inspector agents (p. 36).

   e. Choose **Save**.

   Note
   At this point, Amazon Inspector creates a service-linked role called `AWSServiceRoleForAmazonInspector`. The role grants Amazon Inspector the necessary access to your resources. For more information, see Creating a service-linked role for Amazon Inspector (p. 30).

Step 4: Create and run your assessment template

To create and run your template

1. In the navigation pane, choose **Assessment templates**, and then choose **Create**.
2. For **Name**, enter the name for your assessment template. For this tutorial, enter **`MyFirstTemplateLinux`**.

3. For **Target name**, choose the assessment target that you created above, **`MyTargetLinux`**.
4. For **Rules packages**, choose the rules packages that you want to use in this assessment template.

   For this tutorial, choose **Common Vulnerabilities and Exposures-1.1**.
5. For **Duration**, specify the duration for your assessment template.

   For this tutorial, select **15 minutes**.
6. Choose **Create and run**.

Step 5: Locate and analyze your finding

A completed assessment run produces a set of findings, or potential security issues that Amazon Inspector discovers in your assessment target. You can review the findings and follow the recommended steps to resolve the potential security issues.

In this tutorial, if you complete the preceding steps, your assessment run produces a finding against the common vulnerability CVE-2018-1111.

To locate and analyze your finding

1. In the navigation pane, choose **Assessment runs**. Verify that the status of the run for the assessment template called **MyFirstTemplateLinux** is set to **Collecting data**. This indicates that the assessment run is currently in progress, and the telemetry data for your target is being collected and analyzed against the selected rules packages.
2. You can't view the findings generated by the assessment run while it is still in progress. Let the assessment run complete its entire duration. However, for this tutorial, you can stop the run after several minutes.

   The status of **MyFirstTemplateLinux** changes first to **Stopping**, then in a few minutes to **Analyzing**, and then finally to **Analysis complete**. To see this change in status, choose the **Refresh** icon.
3. In the navigation pane, choose **Findings**.

   You can see a new finding of **High** severity called **Instance InspectorEC2InstanceLinux is vulnerable to CVE-2018-1111**.

   > **Note**
   > If you don't see the new finding, choose the **Refresh** icon.

   To expand the view and see the details of this finding, choose the arrow to the left of the finding. The details of the finding include the following:

   - ARN of the finding
   - Name of the assessment run that produced this finding
   - Name of the assessment target that produced this finding
   - Name of the assessment template that produced this finding
   - Assessment run start time
   - Assessment run end time
   - Assessment run status
   - Name of the rules package that includes the rule that triggered this finding
   - Amazon Inspector agent ID
   - Name of the finding
   - Severity of the finding
   - Description of the finding

- Recommended remediation steps that you can complete to fix the potential security issue described by the finding

Step 6: Apply the recommended fix to your assessment target

For this tutorial, you modified your assessment target to expose it to the potential security issue CVE-2018-1111. In this procedure, you apply the recommended fix for the issue.

To apply the fix to your target

1. Connect to your instance **`InspectorEC2InstanceLinux`** that you created in the preceding section, and run the following command:

   ```
   sudo yum update dhclient-12:4.2.5-68.el7
   ```
2. On the **Assessment templates** page, choose **MyFirstTemplateLinux**, and then choose **Run** to start a new assessment run using this template.
3. Follow the steps in Step 5: Locate and analyze your finding (p. 12) to see the findings that result from this subsequent run of the **MyFirstTemplateLinux** template.

   Because you resolved the CVE-2018-1111 security issue, you should no longer see a finding for it.

Amazon Inspector tutorial - Ubuntu Server

Before you follow the instructions in this tutorial, we recommend that you get familiar with the Amazon Inspector terminology and concepts (p. 2).

This tutorial shows how to use Amazon Inspector to analyze the behavior of an EC2 instance that runs the Ubuntu Server 16.04 LTS operating system. It provides step-by-step instructions on how to navigate the Amazon Inspector workflow.

If you are a first-time user and would like to set up and run an Amazon Inspector assessment with one click, see Creating a Basic Assessment (p. 7).

Topics

Step 1: Set up an Amazon EC2 instance to use with Amazon Inspector

To set up an EC2 instance

- For this tutorial, create one EC2 instance running Ubuntu Server 16.04 LTS and tag it using the **Name** key and a value of **`InspectorEC2InstanceUbuntu`**.

Note
For more information about tagging EC2 instances, see Resources and Tags.

Step 2: Create an assessment target and install an agent on the EC2 instance

Amazon Inspector uses assessment targets to designate the AWS resources to evaluate.

To create an assessment target and install an agent on the EC2 instance

1. Sign in to the AWS Management Console and open the Amazon Inspector console at https://console.aws.amazon.com/inspector/.
2. In the navigation pane, choose **Assessment targets**, and then choose **Create**.
3. For **Name**, enter the name for your assessment target.

   For this tutorial, type **`MyTargetUbuntu`**.
4. For **Use Tags**, choose the EC2 instances that you want to include in this assessment target by entering values for the **Key** and **Value** fields.

   For this tutorial, choose the EC2 instance that you created in the preceding step by entering **`Name`** in the **Key** field and **`InspectorEC2InstanceUbuntu`** in the **Value** field.

   To include all EC2 instances in your AWS account and Region in the assessment target, select the **All Instances** box.
5. Install an Amazon Inspector Agent on your tagged EC2 instance. To install an agent on all EC2 instances included in an assessment target, select the **Install Agents** box.

   Note
   You can also install the Amazon Inspector Agent using the Systems Manager Run Command (p. 37). To install the agent on all instances in the assessment target, you can specify the same tags used for creating the assessment target. Or you can install the Amazon Inspector Agent on your EC2 instance manually. For more information, see Installing Amazon Inspector agents (p. 36).
6. Choose **Save**.

Note
At this point, a service-linked role called `AWSServiceRoleForAmazonInspector` is created to grant Amazon Inspector access to your resources. For more information, see Creating a service-linked role for Amazon Inspector (p. 30).

Step 3: Create and run your assessment template

To create and run your template

1. If you are using **Advanced setup**, you are directed to the **Define an assessment template** page. Otherwise, navigate to the **Assessment templates** page, and then choose **Create**.
2. For **Name**, enter the name for your assessment template. For this tutorial, enter **`MyFirstTemplateUbuntu`**.
3. For **Target name**, choose the assessment target that you created above, **`MyTargetUbuntu`**.
4. For **Rules packages**, use the dropdown menu to choose the rules packages that you want to use in this assessment template.

For this tutorial, choose **Common Vulnerabilities and Exposures-1.1**.

5. For **Duration**, specify the duration for your assessment template.

   For this tutorial, choose **15 minutes**.

6. If you are using **Advanced setup**, choose **Next**. On the following **Review** page, choose **Create**. Otherwise, choose **Create and run**.

Step 4: Locate and analyze generated findings

A completed assessment run produces a set of findings, or potential security issues that Amazon Inspector discovers in your assessment target. You can review the findings and follow the recommended steps to resolve the potential security issues.

1. Navigate to the **Assessment Runs** page. Verify that the status of the run for the assessment template called **MyFirstTemplateUbuntu** that you created in the preceding step is set to **Collecting data**. This indicates that the assessment run is currently in progress, and the telemetry data for your target is being collected and analyzed against the selected rules packages.
2. You can't view the findings generated by the assessment run while it is still in progress. Let the assessment run complete its entire duration.

   The status of **MyFirstTemplateUbuntu** changes first to **Stopping**, then in a few minutes to **Analyzing**, and then finally to **Analysis complete**. To see this change in status, choose the **Refresh** icon.

3. Navigate to the **Findings** page.

   To expand the view and see the details of a finding, choose the arrow to the left of the finding. The details of the finding include the following:

   - ARN of the finding
   - Name of the assessment run that produced this finding
   - Name of the assessment target that produced this finding
   - Name of the assessment template that produced this finding
   - Assessment run start time
   - Assessment run end time
   - Assessment run status
   - Name of the rules package that includes the rule that triggered the finding
   - Amazon Inspector agent ID
   - Name of the finding
   - Severity of the finding
   - Description of the finding
   - Recommended remediation steps that you can complete to fix the potential security issue described by the finding

Step 5: Apply the recommended fix to your assessment target

In this procedure, you apply an update to fix the uncovered issues.

1. Connect to your instance **`InspectorEC2InstanceUbuntu`**, and perform a package update.

2. On the **Assessment templates** page, choose **MyFirstTemplateUbuntu**, and then choose **Run** to start a new run using this template.
3. Follow the steps in Step 4: Locate and analyze generated findings (p. 15) to see the findings that result from this subsequent run of the **MyFirstTemplateUbuntu** template.

   The package update should have resolved the findings from the first run of the template.

Security in Amazon Inspector

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from data centers and network architectures that are built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security *of* the cloud and security *in* the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs. To learn about the compliance programs that apply to Amazon Inspector, see AWS Services in Scope by Compliance Program.
- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when using Amazon Inspector. The following topics show you how to configure Amazon Inspector to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Amazon Inspector resources.

Topics

Data protection in Amazon Inspector

The AWS shared responsibility model applies to data protection in Amazon Inspector. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. This content includes the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see the Data Privacy FAQ. For information about data protection in Europe, see the AWS Shared Responsibility Model and GDPR blog post on the *AWS Security Blog*.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual user accounts with AWS Identity and Access Management (IAM). That way each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We recommend TLS 1.2 or later.

- Set up API and user activity logging with AWS CloudTrail.
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing personal data that is stored in Amazon S3.
- If you require FIPS 140-2 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-2.

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form fields such as a **Name** field. This includes when you work with Amazon Inspector or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form fields used for names may be used for billing or diagnostic logs. If you provide a URL to an external server, we strongly recommend that you do not include credentials information in the URL to validate your request to that server.

Topics

Encryption of data at rest

The telemetry data that an Amazon Inspector agent generates during assessment runs is formatted in JSON files. These files are delivered in near-real-time over TLS to Amazon Inspector, where they are encrypted with a per-assessment-run, ephemeral AWS KMS-derived key.

The files are securely stored in S3 buckets that are dedicated to Amazon Inspector. The rules engine of Amazon Inspector does the following:

- Accesses the encrypted telemetry data in the S3 bucket
- Decrypts it in memory
- Processes the data against the configured assessment rules to generate findings

Encryption of data in transit

During an assessment, the agent gathers telemetry data from the system to send back to Amazon Inspector over a TLS-protected channel.

Clients must support Transport Layer Security (TLS) 1.0 or later. We recommend TLS 1.2 or later. Clients must also support cipher suites with perfect forward secrecy (PFS) such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems such as Java 7 and later support these modes.

Identity and access management for Amazon Inspector

AWS Identity and Access Management (IAM) is an AWS service that helps an administrator securely control access to AWS resources. IAM administrators control who can be *authenticated* (signed in) and *authorized* (have permissions) to use Amazon Inspector resources. IAM is an AWS service that you can use with no additional charge.

Topics

Audience

How you use AWS Identity and Access Management (IAM) differs, depending on the work that you do in Amazon Inspector.

Service user – If you use the Amazon Inspector service to do your job, then your administrator provides you with the credentials and permissions that you need. As you use more Amazon Inspector features to do your work, you might need additional permissions. Understanding how access is managed can help you request the right permissions from your administrator. If you cannot access a feature in Amazon Inspector, see Troubleshooting Amazon Inspector identity and access (p. 27).

Service administrator – If you're in charge of Amazon Inspector resources at your company, you probably have full access to Amazon Inspector. It's your job to determine which Amazon Inspector features and resources your employees should access. You must then submit requests to your IAM administrator to change the permissions of your service users. Review the information on this page to understand the basic concepts of IAM. To learn more about how your company can use IAM with Amazon Inspector, see How Amazon Inspector works with IAM (p. 23).

IAM administrator – If you're an IAM administrator, you might want to learn details about how you can write policies to manage access to Amazon Inspector. To view example Amazon Inspector identity-based policies that you can use in IAM, see Amazon Inspector identity-based policy examples (p. 25).

Authenticating with identities

Authentication is how you sign in to AWS using your identity credentials. For more information about signing in using the AWS Management Console, see Signing in to the AWS Management Console as an IAM user or root user in the *IAM User Guide*.

You must be *authenticated* (signed in to AWS) as the AWS account root user, an IAM user, or by assuming an IAM role. You can also use your company's single sign-on authentication or even sign in using Google or Facebook. In these cases, your administrator previously set up identity federation using IAM roles. When you access AWS using credentials from another company, you are assuming a role indirectly.

To sign in directly to the AWS Management Console, use your password with your root user email address or your IAM user name. You can access AWS programmatically using your root user or IAM users access keys. AWS provides SDK and command line tools to cryptographically sign your request using your credentials. If you don't use AWS tools, you must sign the request yourself. Do this using *Signature Version 4*, a protocol for authenticating inbound API requests. For more information about authenticating requests, see Signature Version 4 signing process in the *AWS General Reference*.

Regardless of the authentication method that you use, you might also be required to provide additional security information. For example, AWS recommends that you use multi-factor authentication (MFA) to increase the security of your account. To learn more, see Using multi-factor authentication (MFA) in AWS in the *IAM User Guide*.

AWS account root user

When you first create an AWS account, you begin with a single sign-in identity that has complete access to all AWS services and resources in the account. This identity is called the AWS account *root user* and is accessed by signing in with the email address and password that you used to create the account. We strongly recommend that you do not use the root user for your everyday tasks, even the administrative ones. Instead, adhere to the best practice of using the root user only to create your first IAM user. Then securely lock away the root user credentials and use them to perform only a few account and service management tasks.

IAM users and groups

An *IAM user* is an identity within your AWS account that has specific permissions for a single person or application. An IAM user can have long-term credentials such as a user name and password or a set of access keys. To learn how to generate access keys, see Managing access keys for IAM users in the *IAM User Guide*. When you generate access keys for an IAM user, make sure you view and securely save the key pair. You cannot recover the secret access key in the future. Instead, you must generate a new access key pair.

An *IAM group* is an identity that specifies a collection of IAM users. You can't sign in as a group. You can use groups to specify permissions for multiple users at a time. Groups make permissions easier to manage for large sets of users. For example, you could have a group named *IAMAdmins* and give that group permissions to administer IAM resources.

Users are different from roles. A user is uniquely associated with one person or application, but a role is intended to be assumable by anyone who needs it. Users have permanent long-term credentials, but roles provide temporary credentials. To learn more, see When to create an IAM user (instead of a role) in the *IAM User Guide*.

IAM roles

An *IAM role* is an identity within your AWS account that has specific permissions. It is similar to an IAM user, but is not associated with a specific person. You can temporarily assume an IAM role in the AWS Management Console by switching roles. You can assume a role by calling an AWS CLI or AWS API operation or by using a custom URL. For more information about methods for using roles, see Using IAM roles in the *IAM User Guide*.

IAM roles with temporary credentials are useful in the following situations:

- **Temporary IAM user permissions** – An IAM user can assume an IAM role to temporarily take on different permissions for a specific task.
- **Federated user access** – Instead of creating an IAM user, you can use existing identities from AWS Directory Service, your enterprise user directory, or a web identity provider. These are known as *federated users*. AWS assigns a role to a federated user when access is requested through an identity provider. For more information about federated users, see Federated users and roles in the *IAM User Guide*.
- **Cross-account access** – You can use an IAM role to allow someone (a trusted principal) in a different account to access resources in your account. Roles are the primary way to grant cross-account access. However, with some AWS services, you can attach a policy directly to a resource (instead of using a role as a proxy). To learn the difference between roles and resource-based policies for cross-account access, see How IAM roles differ from resource-based policies in the *IAM User Guide*.
- **Cross-service access** – Some AWS services use features in other AWS services. For example, when you make a call in a service, it's common for that service to run applications in Amazon EC2 or store objects in Amazon S3. A service might do this using the calling principal's permissions, using a service role, or using a service-linked role.

  - **Principal permissions** – When you use an IAM user or role to perform actions in AWS, you are considered a principal. Policies grant permissions to a principal. When you use some services, you might perform an action that then triggers another action in a different service. In this case, you must have permissions to perform both actions. To see whether an action requires additional dependent actions in a policy, see Actions, Resources, and Condition Keys for Amazon Inspector in the *Service Authorization Reference*.
  - **Service role** – A service role is an IAM role that a service assumes to perform actions on your behalf. An IAM administrator can create, modify, and delete a service role from within IAM. For more information, see Creating a role to delegate permissions to an AWS service in the *IAM User Guide*.
  - **Service-linked role** – A service-linked role is a type of service role that is linked to an AWS service. The service can assume the role to perform an action on your behalf. Service-linked roles appear in your IAM account and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.
- **Applications running on Amazon EC2** – You can use an IAM role to manage temporary credentials for applications that are running on an EC2 instance and making AWS CLI or AWS API requests. This is preferable to storing access keys within the EC2 instance. To assign an AWS role to an EC2 instance and make it available to all of its applications, you create an instance profile that is attached to the instance. An instance profile contains the role and enables programs that are running on the EC2 instance to get temporary credentials. For more information, see Using an IAM role to grant permissions to applications running on Amazon EC2 instances in the *IAM User Guide*.

To learn whether to use IAM roles or IAM users, see When to create an IAM role (instead of a user) in the *IAM User Guide*.

Managing access using policies

You control access in AWS by creating policies and attaching them to IAM identities or AWS resources. A policy is an object in AWS that, when associated with an identity or resource, defines their permissions. You can sign in as the root user or an IAM user, or you can assume an IAM role. When you then make a request, AWS evaluates the related identity-based or resource-based policies. Permissions in the policies determine whether the request is allowed or denied. Most policies are stored in AWS as JSON documents. For more information about the structure and contents of JSON policy documents, see Overview of JSON policies in the *IAM User Guide*.

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

Every IAM entity (user or role) starts with no permissions. In other words, by default, users can do nothing, not even change their own password. To give a user permission to do something, an administrator must attach a permissions policy to a user. Or the administrator can add the user to a group that has the intended permissions. When an administrator gives permissions to a group, all users in that group are granted those permissions.

IAM policies define permissions for an action regardless of the method that you use to perform the operation. For example, suppose that you have a policy that allows the `iam:GetRole` action. A user with that policy can get role information from the AWS Management Console, the AWS CLI, or the AWS API.

Identity-based policies

Identity-based policies are JSON permissions policy documents that you can attach to an identity, such as an IAM user, group of users, or role. These policies control what actions users and roles can perform, on which resources, and under what conditions. To learn how to create an identity-based policy, see Creating IAM policies in the *IAM User Guide*.

Identity-based policies can be further categorized as *inline policies* or *managed policies*. Inline policies are embedded directly into a single user, group, or role. Managed policies are standalone policies that

you can attach to multiple users, groups, and roles in your AWS account. Managed policies include AWS managed policies and customer managed policies. To learn how to choose between a managed policy or an inline policy, see Choosing between managed policies and inline policies in the *IAM User Guide*.

Resource-based policies

Resource-based policies are JSON policy documents that you attach to a resource. Examples of resource-based policies are IAM *role trust policies* and Amazon S3 *bucket policies*. In services that support resource-based policies, service administrators can use them to control access to a specific resource. For the resource where the policy is attached, the policy defines what actions a specified principal can perform on that resource and under what conditions. You must specify a principal in a resource-based policy. Principals can include accounts, users, roles, federated users, or AWS services.

Resource-based policies are inline policies that are located in that service. You can't use AWS managed policies from IAM in a resource-based policy.

Access control lists (ACLs)

Access control lists (ACLs) control which principals (account members, users, or roles) have permissions to access a resource. ACLs are similar to resource-based policies, although they do not use the JSON policy document format.

Amazon S3, AWS WAF, and Amazon VPC are examples of services that support ACLs. To learn more about ACLs, see Access control list (ACL) overview in the *Amazon Simple Storage Service Developer Guide*.

Other policy types

AWS supports additional, less-common policy types. These policy types can set the maximum permissions granted to you by the more common policy types.

- **Permissions boundaries** – A permissions boundary is an advanced feature in which you set the maximum permissions that an identity-based policy can grant to an IAM entity (IAM user or role). You can set a permissions boundary for an entity. The resulting permissions are the intersection of entity's identity-based policies and its permissions boundaries. Resource-based policies that specify the user or role in the `Principal` field are not limited by the permissions boundary. An explicit deny in any of these policies overrides the allow. For more information about permissions boundaries, see Permissions boundaries for IAM entities in the *IAM User Guide*.
- **Service control policies (SCPs)** – SCPs are JSON policies that specify the maximum permissions for an organization or organizational unit (OU) in AWS Organizations. AWS Organizations is a service for grouping and centrally managing multiple AWS accounts that your business owns. If you enable all features in an organization, then you can apply service control policies (SCPs) to any or all of your accounts. The SCP limits permissions for entities in member accounts, including each AWS account root user. For more information about Organizations and SCPs, see How SCPs work in the *AWS Organizations User Guide*.
- **Session policies** – Session policies are advanced policies that you pass as a parameter when you programmatically create a temporary session for a role or federated user. The resulting session's permissions are the intersection of the user or role's identity-based policies and the session policies. Permissions can also come from a resource-based policy. An explicit deny in any of these policies overrides the allow. For more information, see Session policies in the *IAM User Guide*.

Multiple policy types

When multiple types of policies apply to a request, the resulting permissions are more complicated to understand. To learn how AWS determines whether to allow a request when multiple policy types are involved, see Policy evaluation logic in the *IAM User Guide*.

How Amazon Inspector works with IAM

Before you use IAM to manage access to Amazon Inspector, you should understand what IAM features are available to use with Amazon Inspector. To get a high-level view of how Amazon Inspector and other AWS services work with IAM, see AWS Services That Work with IAM in the *IAM User Guide*.

Topics

Amazon Inspector identity-based policies

With IAM identity-based policies, you can specify allowed or denied actions and resources as well as the conditions under which actions are allowed or denied. Amazon Inspector supports specific actions, resources, and condition keys. To learn about all of the elements that you use in a JSON policy, see IAM JSON Policy Elements Reference in the *IAM User Guide*.

Actions

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

The `Action` element of a JSON policy describes the actions that you can use to allow or deny access in a policy. Policy actions usually have the same name as the associated AWS API operation. There are some exceptions, such as *permission-only actions* that don't have a matching API operation. There are also some operations that require multiple actions in a policy. These additional actions are called *dependent actions*.

Include actions in a policy to grant permissions to perform the associated operation.

Policy actions in Amazon Inspector use the following prefix before the action: `inspector:`. For example, the `inspector:ListFindings` permission allows the user permissions to perform the Amazon Inspector `ListFindings` operation. Policy statements must include either an `Action` or `NotAction` element. Amazon Inspector defines its own set of actions that describe tasks that you can perform with this service.

To specify multiple actions in a single statement, separate them with commas as follows:

```
"Action": [
      "inspector:action1",
      "inspector:action2"
```

You can specify multiple actions using wildcards (*). For example, to specify all actions that begin with the word `Describe`, include the following action:

```
"Action": "inspector:Describe*"
```

To see a list of Amazon Inspector actions, see Actions Defined by Amazon Inspector in the *IAM User Guide*.

Resources

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

The `Resource` JSON policy element specifies the object or objects to which the action applies. Statements must include either a `Resource` or a `NotResource` element. As a best practice, specify a resource using its Amazon Resource Name (ARN). You can do this for actions that support a specific resource type, known as *resource-level permissions*.

For actions that don't support resource-level permissions, such as listing operations, use a wildcard (*) to indicate that the statement applies to all resources.

```
"Resource": "*"
```

Amazon Inspector does not support specifying a resource ARN in the `Resource` element of an IAM policy statement. To allow access to Amazon Inspector, specify `"Resource": "*"` in your policy.

Condition keys

Amazon Inspector does not provide any service-specific condition keys, but it does support using some global condition keys. To see all AWS global condition keys, see AWS global condition context keys in the *IAM User Guide*.

Managed policies for Amazon Inspector

Amazon Inspector provides the following AWS managed policies, which you can attach to IAM users in your account.

- `AmazonInspectorFullAccess` – Provides full access to Amazon Inspector.
- `AmazonInspectorReadOnlyAccess` – Provides read-only access to Amazon Inspector.

Examples

To view examples of Amazon Inspector identity-based policies, see Amazon Inspector identity-based policy examples (p. 25).

Amazon Inspector resource-based policies (not supported)

Amazon Inspector does not support resource-based policies.

Authorization based on Amazon Inspector tags (not supported)

Amazon Inspector does not support tagging resources or controlling access based on tags

Amazon Inspector IAM roles

An IAM role is an entity within your AWS account that has specific permissions.

Using temporary credentials with Amazon Inspector

You can use temporary credentials to sign in with federation, to assume an IAM role, or to assume a cross-account role. You obtain temporary security credentials by calling AWS STS API operations such as AssumeRole or GetFederationToken.

Amazon Inspector supports using temporary credentials.

Service-linked roles

Service-linked roles allow AWS services to access resources in other services to complete an action on your behalf. Service-linked roles appear in your IAM account and are owned by the service. An IAM administrator can view but not edit the permissions for service-linked roles.

Amazon Inspector supports service-linked roles. For details about creating or managing Amazon Inspector service-linked roles, see the section called “Using service-linked roles” (p. 29).

Service roles

This feature allows a service to assume a service role on your behalf. This role allows the service to access resources in other services to complete an action on your behalf. Service roles appear in your IAM account and are owned by the account. This means that an IAM administrator can change the permissions for this role. However, doing so might break the functionality of the service.

Amazon Inspector supports service roles.

Amazon Inspector identity-based policy examples

By default, IAM users and roles don't have permission to create or modify Amazon Inspector resources. They also can't perform tasks using the AWS Management Console, AWS CLI, or AWS API. An IAM administrator must create IAM policies that grant users and roles permission to perform specific API operations on the specified resources they need. The administrator must then attach those policies to the IAM users or groups that require those permissions.

To learn how to create an IAM identity-based policy using these example JSON policy documents, see Creating Policies on the JSON Tab in the *IAM User Guide*.

Topics

Policy best practices

Identity-based policies are very powerful. They determine whether someone can create, access, or delete Amazon Inspector resources in your account. These actions can incur costs for your AWS account. When you create or edit identity-based policies, follow these guidelines and recommendations:

- **Get started using AWS managed policies** – To start using Amazon Inspector quickly, use AWS managed policies to give your employees the permissions they need. These policies are already available in your account and are maintained and updated by AWS. For more information, see Get started using permissions with AWS managed policies in the *IAM User Guide*.
- **Grant least privilege** – When you create custom policies, grant only the permissions required to perform a task. Start with a minimum set of permissions and grant additional permissions as necessary. Doing so is more secure than starting with permissions that are too lenient and then trying to tighten them later. For more information, see Grant least privilege in the *IAM User Guide*.
- **Enable MFA for sensitive operations** – For extra security, require IAM users to use multi-factor authentication (MFA) to access sensitive resources or API operations. For more information, see Using multi-factor authentication (MFA) in AWS in the *IAM User Guide*.

- **Use policy conditions for extra security** – To the extent that it's practical, define the conditions under which your identity-based policies allow access to a resource. For example, you can write conditions to specify a range of allowable IP addresses that a request must come from. You can also write conditions to allow requests only within a specified date or time range, or to require the use of SSL or MFA. For more information, see IAM JSON policy elements: Condition in the *IAM User Guide*.

Using the Amazon Inspector console

To access the Amazon Inspector console, you must have a minimum set of permissions. These permissions must allow you to list and view details about the Amazon Inspector resources in your AWS account. If you create an identity-based policy that is more restrictive than the minimum required permissions, the console won't function as intended for entities (IAM users or roles) with that policy.

To ensure that those entities can still use the Amazon Inspector console, also attach one of the following AWS managed policies to the entities. For more information, see Adding Permissions to a User in the *IAM User Guide*.

- `AmazonInspectorFullAccess`
- `AmazonInspectorReadOnlyAccess`

You don't need to allow minimum console permissions for users that are making calls only to the AWS CLI or the AWS API. Instead, allow access to only the actions that match the API operation that you're trying to perform.

Allow users to view their own permissions

This example shows how you might create a policy that allows IAM users to view the inline and managed policies that are attached to their user identity. This policy includes permissions to complete this action on the console or programmatically using the AWS CLI or AWS API.

```
{
    "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": "*"
```

```
        }
    ]
}
```

Allow a user to perform any describe and list operations on any Amazon Inspector resource

The following permissions policy grants a user permission to run all the operations that begin with `Describe` and `List`. These operations show information about an Amazon Inspector resource, such as an assessment target or finding. The wildcard character (*) in the `Resource` element indicates that the operations are allowed for all Amazon Inspector resources that are owned by the account:

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action": [
            "inspector:Describe*",
            "inspector:List*"
            ],
         "Resource":"*"
      }
   ]
}
```

Example 2: Allow a user to perform describe and list operations only on Amazon Inspector findings

The following permissions policy grants a user permission to run only `ListFindings` and `DescribeFindings` operations. These operations show information about Amazon Inspector findings. The wildcard character (*) in the `Resource` element indicates that the operations are allowed for all Amazon Inspector resources that are owned by the account.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action": [
                  "inspector:DescribeFindings",
                  "inspector:ListFindings"
         ],
         "Resource":"*"
      }
   ]
}
```

Troubleshooting Amazon Inspector identity and access

Use the following information to help you diagnose and fix common issues that you might encounter when working with Amazon Inspector and IAM.

Topics

I am not authorized to perform an action in Amazon Inspector

If the AWS Management Console tells you that you're not authorized to perform an action, then you must contact your administrator for assistance. Your administrator is the person that provided you with your user name and password.

The following example error occurs when the `mateojackson` IAM user tries to use the console to create an assessment template but does not have `inspector:CreateAssessmentTemplate` permissions.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 inspector:CreateAssessmentTemplate
```

In this case, Mateo asks his administrator to update his policies to allow him access to the `inspector:CreateAssessmentTemplate` action.

I am not authorized to perform iam:PassRole

If you receive an error that you're not authorized to perform the `iam:PassRole` action, then you must contact your administrator for assistance. Your administrator is the person that provided you with your user name and password. Ask that person to update your policies to allow you to pass a role to Amazon Inspector.

Some AWS services allow you to pass an existing role to that service, instead of creating a new service role or service-linked role. To do this, you must have permissions to pass the role to the service.

The following example error occurs when an IAM user named `marymajor` tries to use the console to perform an action in Amazon Inspector. However, the action requires the service to have permissions granted by a service role. Mary does not have permissions to pass the role to the service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform: iam:PassRole
```

In this case, Mary asks her administrator to update her policies to allow her to perform the `iam:PassRole` action.

I want to view my access keys

After you create your IAM user access keys, you can view your access key ID at any time. However, you can't view your secret access key again. If you lose your secret key, you must create a new access key pair.

Access keys consist of two parts: an access key ID (for example, `AKIAIOSFODNN7EXAMPLE`) and a secret access key (for example, `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`). Like a user name and password, you must use both the access key ID and secret access key together to authenticate your requests. Manage your access keys as securely as you do your user name and password.

> **Important**
> Do not provide your access keys to a third party, even to help find your canonical user ID. By doing this, you might give someone permanent access to your account.

When you create an access key pair, you are prompted to save the access key ID and secret access key in a secure location. The secret access key is available only at the time you create it. If you lose your secret access key, you must add new access keys to your IAM user. You can have a maximum of two access keys. If you already have two, you must delete one key pair before creating a new one. To view instructions, see Managing access keys in the *IAM User Guide*.

I'm an administrator and want to allow others to access Amazon Inspector

To allow others to access Amazon Inspector, you must create an IAM entity (user or role) for the person or application that needs access. They will use the credentials for that entity to access AWS. You must then attach a policy to the entity that grants them the correct permissions in Amazon Inspector.

To get started right away, see Creating your first IAM delegated user and group in the *IAM User Guide*.

I want to allow people outside of my AWS account to access my Amazon Inspector resources

You can create a role that users in other accounts or people outside of your organization can use to access your resources. You can specify who is trusted to assume the role. For services that support resource-based policies or access control lists (ACLs), you can use those policies to grant people access to your resources.

To learn more, consult the following:

- To learn whether Amazon Inspector supports these features, see How Amazon Inspector works with IAM (p. 23).
- To learn how to provide access to your resources across AWS accounts that you own, see Providing access to an IAM user in another AWS account that you own in the *IAM User Guide*.
- To learn how to provide access to your resources to third-party AWS accounts, see Providing access to AWS accounts owned by third parties in the *IAM User Guide*.
- To learn how to provide access through identity federation, see Providing access to externally authenticated users (identity federation) in the *IAM User Guide*.
- To learn the difference between using roles and resource-based policies for cross-account access, see How IAM roles differ from resource-based policies in the *IAM User Guide*.

Using service-linked roles for Amazon Inspector

Amazon Inspector uses AWS Identity and Access Management (IAM) service-linked roles. A service-linked role is a unique type of IAM role that is linked directly to Amazon Inspector. Service-linked roles are predefined by Amazon Inspector and include all the permissions that the service requires to call other AWS services on your behalf.

A service-linked role makes setting up Amazon Inspector easier because you don't have to manually add the necessary permissions. Amazon Inspector defines the permissions of its service-linked roles, and unless defined otherwise, only Amazon Inspector can assume its roles. The defined permissions include the trust policy and the permissions policy, and that permissions policy can't be attached to any other IAM entity.

You can delete a service-linked role only after first deleting your assessment targets for an AWS account in all the Regions where you have Amazon Inspector running.

For information about other services that support service-linked roles, see AWS Services That Work with IAM and look for the services that have **Yes** in the **Service-Linked Role** column. Choose a **Yes** with a link to view the service-linked role documentation for that service.

Service-linked role permissions for Amazon Inspector

Amazon Inspector uses the service-linked role named `AWSServiceRoleForAmazonInspector`. The `AWSServiceRoleForAmazonInspector` service-linked role trusts Amazon Inspector to assume the role.

The permissions policy of the role allows Amazon Inspector to complete the following action on the specified resources:

- Action: `iam:CreateServiceLinkedRole` on `arn:aws:iam::*:role/aws-service-role/inspector.amazonaws.com/AWSServiceRoleForAmazonInspector`

For the `AWSServiceRoleForAmazonInspector` role to be successfully created, the IAM identity (user, role, or group) that you use when you work with Amazon Inspector must have the required permissions. To grant the required permissions, attach the `AmazonInspectorFullAccess` managed policy to the IAM user, group, or role. For more information about the managed policy, see the section called "Managed policies for Amazon Inspector" (p. 24).

For more information about service-linked roles, see Service-Linked Role Permissions in the *IAM User Guide*.

Creating a service-linked role for Amazon Inspector

You don't need to manually create the `AWSServiceRoleForAmazonInspector` service-linked role.

The `AWSServiceRoleForAmazonInspector` service-linked role is created automatically, but you might need to do some minimal setup first. The following sections describe the details of setting up and using the `AWSServiceRoleForAmazonInspector` service-linked role.

If you are getting started with Amazon Inspector for the first time

- The `AWSServiceRoleForAmazonInspector` service-linked role is created automatically when you go through the **Get Started with Amazon Inspector** wizard on the console or when you run the CreateAssessmentTarget API operation.
- The `AWSServiceRoleForAmazonInspector` service-linked role is created for your AWS account only in the Region that you are currently signed in to. It grants Amazon Inspector access to the resources in your AWS account only in that Region. If you then use the same AWS account to go through the **Get Started with Amazon Inspector** console wizard or run the CreateAssessmentTarget API operation in other Regions, the same service-linked role that is already created in your AWS account is applied in these other Regions and grants Amazon Inspector access to the resources in your AWS account in those Regions.

If you already have Amazon Inspector running in your AWS account

- If you already have Amazon Inspector running in your AWS account, the IAM role that grants Amazon Inspector access to your resources already exists in your AWS account. In this case, the `AWSServiceRoleForAmazonInspector` service-linked role is generated when you create an assessment target or an assessment template (either through the Amazon Inspector console or the API operations). This newly created service-linked role replaces the previously created IAM role that up until now granted Amazon Inspector access to your resources.

  You can also create the `AWSServiceRoleForAmazonInspector` service-linked role manually by choosing the **Manage Amazon Inspector service-linked role** link in the **Accounts Setting** section on the Amazon Inspector **Dashboard** page. This newly created service-linked role replaces the previously created IAM role that up until now granted Amazon Inspector access to your resources.

Note
This previously created IAM role is not deleted. It remains intact, but it is no longer used to grant Amazon Inspector access to your resources. You can use the IAM console to further manage or delete this IAM role.

- The `AWSServiceRoleForAmazonInspector` service-linked role is created for your AWS account only in the Region that you are currently signed in to. It grants Amazon Inspector access to the resources in your AWS account only in this Region. Suppose you use the same AWS account to create an assessment target or an assessment template for your Amazon Inspector service running in other Regions. In that case, the same service-linked role that is already created in your AWS account is applied. This role grants Amazon Inspector access to the resources in your AWS account in those Regions.

You can also use the IAM console to create an Inspector service-linked role. In the IAM CLI or the IAM API, create a service-linked role with the `Amazon Inspector` service name. For more information, see Creating a Service-Linked Role in the *IAM User Guide*.

If you delete this service-linked role, and then need to create it again, you can use the same process to recreate the role in your account. When you Get started with Amazon Inspector again, the service-linked role is automatically created for you again.

Editing a service-linked role for Amazon Inspector

Amazon Inspector does not allow you to edit the `AWSServiceRoleForAmazonInspector` service-linked role. After you create a service-linked role, you can't change the name of the role because various entities might reference the role. However, you can edit the description of the role using IAM. For more information, see Editing a Service-Linked Role in the *IAM User Guide*.

Deleting a service-linked role for Amazon Inspector

If you no longer need to use a feature or service that requires a service-linked role, we recommend that you delete that role. That way you don't have an unused entity that is not actively monitored or maintained. However, you must clean up the resources for your service-linked role before you can manually delete it.

Note
If the Amazon Inspector service is using the role when you try to delete the resources, then the deletion might fail. If that happens, wait for a few minutes and try the operation again.

To delete Amazon Inspector resources used by `AWSServiceRoleForAmazonInspector`

- Delete your assessment targets for this AWS account in all the Regions where you have Amazon Inspector running. For more information, see Amazon Inspector assessment targets (p. 48).

To manually delete the service-linked role using IAM

Use the IAM console, the IAM CLI, or the IAM API to delete the `AWSServiceRoleForAmazonInspector` service-linked role. For more information, see Deleting a Service-Linked Role in the *IAM User Guide*.

Logging and monitoring in Amazon Inspector

Amazon Inspector is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Amazon Inspector. CloudTrail captures all API calls for Amazon Inspector as events, including calls from the Amazon Inspector console and code calls to the Amazon Inspector API operations.

For information on using CloudTrail logging in Amazon Inspector, see *Logging Amazon Inspector API calls with AWS CloudTrail* (p. 81).

You can monitor Amazon Inspector using Amazon CloudWatch, which collects and processes raw data into readable, near-real time metrics. By default, Amazon Inspector sends metric data to CloudWatch in 5-minute periods.

For information on using CloudWatch with Amazon Inspector, see *Monitoring Amazon Inspector using Amazon CloudWatch* (p. 84).

Incident response in Amazon Inspector

Incident response for Amazon Inspector is an AWS responsibility. AWS has a formal, documented policy and program that governs incident response.

AWS operational issues with broad impact are posted on the AWS Service Health Dashboard.

Operational issues are also posted to individual accounts via the AWS Personal Health Dashboard. For information on how to use the AWS Personal Health Dashboard, see the AWS Health User Guide.

Compliance validation for Amazon Inspector

Third-party auditors assess the security and compliance of Amazon Inspector as part of multiple AWS compliance programs. These include SOC, PCI, FedRAMP, HIPAA, and others.

For a list of AWS services in scope of specific compliance programs, see AWS Services in Scope by Compliance Program. For general information, see AWS Compliance Programs.

You can download third-party audit reports using AWS Artifact. For more information, see Downloading Reports in AWS Artifact.

Your compliance responsibility when using Amazon Inspector is determined by the sensitivity of your data, your company's compliance objectives, and applicable laws and regulations. AWS provides the following resources to help with compliance:

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying security- and compliance-focused baseline environments on AWS.
- Architecting for HIPAA Security and Compliance Whitepaper – This whitepaper describes how companies can use AWS to create HIPAA-compliant applications.
- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- Evaluating Resources with Rules in the *AWS Config Developer Guide* – The AWS Config service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- AWS Security Hub – This AWS service provides a comprehensive view of your security state within AWS that helps you check your compliance with security industry standards and best practices.

Resilience in Amazon Inspector

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency,

high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

For more information about AWS Regions and Availability Zones, see AWS Global Infrastructure.

Amazon Inspector is highly available and executes queries using compute resources across multiple Availability Zones. It automatically routes queries appropriately if a particular Availability Zone is unreachable.

Amazon Inspector uses Amazon S3 as its underlying data store, which makes your data highly available and durable. Amazon S3 provides durable infrastructure to store important data. It is designed for durability of 99.999999999% of objects. Your data is redundantly stored across multiple facilities and multiple devices in each facility.

Infrastructure security in Amazon Inspector

As a managed service, Amazon Inspector is protected by the AWS global network security procedures that are described in the Amazon Web Services: Overview of Security Processes whitepaper.

You use AWS published API calls to access Amazon Inspector through the network. Clients must support Transport Layer Security (TLS) 1.0 or later. We recommend TLS 1.2 or later. Clients must also support cipher suites with perfect forward secrecy (PFS) such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems such as Java 7 and later support these modes.

Additionally, requests must be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.

For more information about Amazon Inspector network and agent security, see the section called "Network and Amazon Inspector agent security" (p. 34).

Configuration and vulnerability analysis in Amazon Inspector

Amazon Inspector offers predefined software called an agent that you can optionally install in the operating system of the EC2 instances that you want to assess. The agent collects a wide set of configuration data, known as telemetry. For more information about Amazon Inspector agents, see *Amazon Inspector agents* (p. 34).

Security best practices for Amazon Inspector

Amazon Inspector provides a number of security features to consider as you develop and implement your own security policies. These best practices are general guidelines and don't represent a complete security solution. Because these best practices might not be appropriate or sufficient for your environment, treat them as helpful considerations rather than prescriptions.

For the list of security best practices for Amazon Inspector, see the section called "Security best practices for Amazon Inspector" (p. 57).

Amazon Inspector agents

The Amazon Inspector agent is an entity that collects installed package information and software configuration for an Amazon EC2 instance. Though not required in all cases, you should install the Amazon Inspector agent on each of your target Amazon EC2 instances in order to fully assess their security.

For more information about how to install, uninstall, and reinstall the agent, how to verify whether the installed agent is running, and how to configure proxy support for the agent, see Working with Amazon Inspector agents on Linux-based operating systems (p. 39) and Working with Amazon Inspector agents on Windows-based operating systems (p. 41).

Note
An Amazon Inspector agent is not required to run the Network Reachability (p. 52) rules package.

Important
The Amazon Inspector agent relies on Amazon EC2 instance metadata to function correctly. It accesses instance metadata using version 1 or version 2 of the Instance Metadata Service (IMDSv1 or IMDSv2). See Instance Metadata and User Data to learn more about EC2 instance metadata and access methods.

Topics

Amazon Inspector agent privileges

You must have administrative or root permissions to install the Amazon Inspector agent. On supported Linux-based operating systems, the agent consists of a user mode executable that runs with root access. On supported Windows-based operating systems, the agent consists of an updater service and an agent service, each running in user mode with `LocalSystem` privileges.

Network and Amazon Inspector agent security

The Amazon Inspector agent initiates all communication with the Amazon Inspector service. This means that the agent must have an outbound network path to public endpoints so that it can send telemetry

data. For example, the agent might connect to `arsenal.<region>.amazonaws.com`, or the endpoint might be an Amazon S3 bucket at `s3.dualstack.<region>.amazonaws.com`. Make sure to replace `<region>` with the actual AWS Region where you are running Amazon Inspector. For more information, see AWS IP Address Ranges. Because all connections from the agent are established outbound, it is not necessary to open ports in your security groups to allow inbound communications to the agent from Amazon Inspector.

The agent periodically communicates with Amazon Inspector over a TLS-protected channel, which is authenticated using either the AWS identity associated with the role of the EC2 instance, or, if no role is assigned, with the instance's metadata document. When authenticated, the agent sends heartbeat messages to the service and receives instructions from the service in response. If an assessment has been scheduled, the agent receives the instructions for that assessment. These instructions are structured JSON files, and they tell the agent to enable or disable specific preconfigured sensors in the agent. Each instruction action is predefined within the agent. Arbitrary instructions can't be executed.

During an assessment, the agent gathers telemetry data from the system to send back to Amazon Inspector over a TLS-protected channel. The agent doesn't make changes to the system that it collects data from. After the agent collects the telemetry data, it sends the data back to Amazon Inspector for processing. Beyond the telemetry data that it generates, the agent is not capable of collecting or transmitting any other data about the system or assessment targets. Currently, there is no method exposed for intercepting and examining telemetry data at the agent.

Amazon Inspector agent updates

As updates for the Amazon Inspector agent become available, they are automatically downloaded from Amazon S3 and applied. This also updates any required dependencies. The auto-update feature eliminates the need for you to track and manually maintain the versioning of the agents that you have installed on your EC2 instances. All updates are subject to audited Amazon change control processes to ensure compliance with applicable security standards.

To further ensure the security of the agent, all communication between the agent and the auto-update release site (S3) is performed over a TLS connection, and the server is authenticated. All binaries involved in the auto-update process are digitally signed, and the signatures are verified by the updater before installation. The auto-update process is executed only during non-assessment periods. If any errors are detected, the update process can rollback and retry the update. Finally, the agent update process serves to upgrade only the agent capabilities. None of your specific information is ever sent from the agent to Amazon Inspector as part of the update workflow. The only information that is communicated as part of the update process is the basic installation success or fail telemetry and, if applicable, any update failure diagnostic information.

Telemetry data lifecycle

The telemetry data that is generated by the Amazon Inspector agent during assessment runs is formatted in JSON files. The files are delivered in near-real-time over TLS to Amazon Inspector, where they are encrypted with a per-assessment-run, ephemeral KMS-derived key. The files are securely stored in an Amazon S3 bucket this is dedicated for Amazon Inspector. The rules engine of Amazon Inspector accesses the encrypted telemetry data in the S3 bucket, decrypts it in memory, and processes the data against the configured assessment rules to generate findings. The telemetry data that is stored in S3 is retained only to allow for assistance with support requests. It isn't used or aggregated by Amazon for any other purpose. After 30 days, telemetry data is permanently deleted according to a standard S3 bucket lifecycle policy for Amazon Inspector data. Currently, Amazon Inspector does not provide an API or an S3 bucket access mechanism to collected telemetry.

Access control from Amazon Inspector into AWS accounts

As a security service, Amazon Inspector accesses your AWS accounts and resources only when it needs to find EC2 instances to assess by querying for tags. It does this through standard IAM access through the role created during the initial setup of the Amazon Inspector service. During an assessment, all communications with your environment are initiated by the Amazon Inspector agent that is installed locally on EC2 instances. The Amazon Inspector service objects that are created, such as assessment targets, assessment templates, and findings generated by the service, are stored in a database managed by and accessible only to Amazon Inspector.

Amazon Inspector agent limits

For information about Amazon Inspector agent limits, see Amazon Inspector service limits (p. 3).

Installing Amazon Inspector agents

You can install the Amazon Inspector agent using the Systems Manager Run Command on multiple instances (including both Linux-based and Windows-based instances). Alternatively, you can install the agent individually by signing in to each EC2 instance. The procedures in this chapter provide instructions for both methods.

As another option, you can quickly install the agent on all Amazon EC2 instances included in an assessment target by selecting the **Install Agents** check box on the **Define an Assessment target** page on the console.

Topics

> **Note**
> The procedures in this chapter apply to all AWS Regions that are supported by Amazon Inspector.

Amazon Linux 2 AMI with the Amazon Inspector Agent

To skip the manual Amazon Inspector agent installation on the Amazon Linux EC2 instances that you want to include in your assessment targets, you can use the **Amazon Linux 2 AMI with Amazon Inspector Agent**. This AMI has the agent preinstalled and requires no additional steps to install or set up the agent. To start using Amazon Inspector with these EC2 instances, tag them to match the assessment target that you want. The configuration of **Amazon Linux 2 AMI with Amazon Inspector Agent** enhances security by focusing on two main security goals: limiting access and reducing software vulnerabilities.

This is the only currently available EC2 instance AMI with the preinstalled Amazon Inspector agent. For the EC2 instances that run Ubuntu Server or Windows Server, you must complete the manual agent installation steps.

The **Amazon Linux 2 AMI with Amazon Inspector Agent** is available on the EC2 console and also at the AWS Marketplace.

Installing the agent on multiple EC2 instances using the Systems Manager Run Command

You can install the Amazon Inspector agent on your EC2 instances using the Systems Manager Run Command. This enables you to install the agent remotely and on multiple instances (both Linux-based and Windows-based instances with the same command) at once.

Important
Agent installation using the Systems Manager Run Command is not currently supported for the Debian operating system.

Important
To use this option, make sure that your EC2 instance has the SSM Agent installed and has an IAM role that allows Run Command. The SSM Agent is installed, by default, on Amazon EC2 Windows instances and Amazon Linux instances. Amazon EC2 Systems Manager requires an IAM role for EC2 instances that processes commands and a separate role for users executing commands. For more information, see Installing and configuring SSM Agent and Configuring security roles for System Manager.

To install the agent on multiple EC2 instances using the Systems Manager Run Command

1. Open the AWS Systems Manager console at https://console.aws.amazon.com/systems-manager/.
2. In the navigation pane under **Instances & nodes**, choose **Run Command**.
3. Choose **Run a command**.
4. For **Command document**, choose the document named **AmazonInspector-ManageAWSAgent** that is owned by **Amazon**. This document contains the script for installing the Amazon Inspector agent on EC2 instances.
5. For **Targets**, you can select EC2 instances using different methods. To install the agent on all of the instances in the assessment target, you can specify the tags that were used to create the assessment target.
6. Provide your choices for the rest of the available options using the instructions in Running commands from the console, and then choose **Run**.

Note
You can also install the agent on multiple EC2 instances (both Linux-based and Windows-based) when you create an assessment target, or you can use the **Install Agents with Run Command** button for an existing target. For more information, see Creating an assessment target (p. 49).

Installing the agent on a Linux-based EC2 instance

Perform the following procedure to install the Amazon Inspector agent on a Linux-based EC2 instance.

To install the agent on a Linux-based EC2 instance

1. Sign in to your EC2 instance running a Linux-based operating system where you want to install the Amazon Inspector agent.

Note
For information about the operating systems that Amazon Inspector supports, see Amazon Inspector supported operating systems and Regions (p. 4).

2. Download the agent installation script by running one of the following commands:

   - **wget https://inspector-agent.amazonaws.com/linux/latest/install**
   - **curl -O https://inspector-agent.amazonaws.com/linux/latest/install**

3. (Optional) Verify that the agent installation script is not altered or corrupted. For more information, see (Optional) Verify the signature of the Amazon Inspector agent installation script on Linux-based operating systems (p. 44).
4. To install the agent, run **sudo bash install**.

   Note
   If you are installing the agent in a SELinux environment the Amazon Inspector may be detected as an unconfined daemon. You can avoid this by changing the domain of the agent process from the default `initrc_t` to `bin_t`. Use the following commands to assign the `bin_t` context to the Amazon Inspector run scripts before installing the agent for SELinux:
   sudo semanage fcontext -a -t bin_t /etc/rc\.d/init\.d/awsagent
   sudo semanage fcontext -a -t bin_t /etc/init\.d/awsagent

   Note
   As updates for the agent become available, they are automatically downloaded from Amazon S3 and applied. For more information, see Amazon Inspector agent updates (p. 35).
   If you want to skip this auto-update process, run the following command when you install the agent:
   sudo bash install -u false

   Note
   (Optional) To remove the agent installation script, run **rm install**.

5. Verify that the following files required for the agent to be successfully installed and functioning properly are installed:

   - `libcurl4` (required to install the agent on Ubuntu 18.04)
   - `libcurl3`
   - `libgcc1`
   - `libc6`
   - `libstdc++6`
   - `libssl1.0.1`
   - `libssl1.0.2` (required to install the agent on Debian 9)
   - `libssl1.1` (required to install the agent on Ubuntu 20.04 LTS)
   - `libpcap0.8`

Installing the agent on a Windows-based EC2 instance

Perform the following procedure to install the Amazon Inspector agent on a Windows-based EC2 instance.

To install the agent on a Windows-based EC2 instance

1. Sign in to your EC2 instance running a Windows-based operating system where you want to install the agent.

Note
For more information about the operating systems that Amazon Inspector supports, see Amazon Inspector supported operating systems and Regions (p. 4).

2. Download the following .exe file:

   `https://inspector-agent.amazonaws.com/windows/installer/latest/AWSAgentInstall.exe`

3. Open a command prompt window (with administrative permissions), navigate to the location where you saved the downloaded `AWSAgentInstall.exe`, and run the .exe file to install the agent.

   Note
   As updates for the agent become available, they are automatically downloaded from Amazon S3 and applied. For more information, see Amazon Inspector agent updates (p. 35).
   If you want to skip this auto-update process, run the following command when you install the agent:
   AWSAgentInstall.exe AUTOUPDATE=No

Working with Amazon Inspector agents on Linux-based operating systems

You can install, remove, verify, and modify the behavior of Amazon Inspector agents. Sign in to your Amazon EC2 instance running a Linux-based operating system, and run any of the following procedures. For more information about the operating systems that are supported for Amazon Inspector, see Amazon Inspector supported operating systems and Regions (p. 4).

Important
The Amazon Inspector agent relies on Amazon EC2 instance metadata to function correctly. It accesses instance metadata using version 1 or version 2 of the Instance Metadata Service (IMDSv1 or IMDSv2). See Instance Metadata and User Data to learn more about EC2 instance metadata and access methods.

Note
The commands in this section function in all AWS Regions that are supported by Amazon Inspector.

Topics

Verifying that the Amazon Inspector agent is running

- To verify that the agent is installed and running, sign in to your EC2 instance and run the following command:

  sudo /opt/aws/awsagent/bin/awsagent status

  This command returns the status of the currently running agent, or an error stating that the agent cannot be contacted.

Stopping the Amazon Inspector agent

- To stop the agent, run the following command:

  sudo /etc/init.d/awsagent stop

Starting the Amazon Inspector agent

- To start the agent, run the following command:

  sudo /etc/init.d/awsagent start

Modifying Amazon Inspector agents settings

After the Amazon Inspector agent is installed and running on your EC2 instance, you can modify the settings in the `agent.cfg` file to alter the agent's behavior. On Linux-based operating systems, the `agent.cfg` file is located in the `/opt/aws/awsagent/etc` directory. After you modify and save the `agent.cfg` file, you must stop and start the agent for the changes to take effect.

Important
We highly recommend that you modify the `agent.cfg` file only with the guidance of AWS Support.

Configuring proxy support for an Amazon Inspector agent

To get proxy support for an agent on a Linux-based operating system, use an agent-specific configuration file with specific environment variables. For more information, see https://wiki.archlinux.org/index.php/proxy_settings.

Complete one of the following procedures:

To install an agent on an EC2 instance that uses a proxy server

1. Create a file called `awsagent.env` and save it in the `/etc/init.d/` directory.
2. Edit `awsagent.env` to include these environment variables in the following format:

   - `export https_proxy=hostname:port`
   - `export http_proxy=hostname:port`
   - `export no_proxy=169.254.169.254`

   Note
   Substitute values in the preceding examples with valid hostname and port number combinations only. Specify the IP address of the instance metadata endpoint (169.254.169.254) for the `no_proxy` variable.
3. Install the Amazon Inspector agent by completing the steps in the Installing the agent on a Linux-based EC2 instance (p. 37) procedure.

To configure proxy support on an EC2 instance with a running agent

1. To configure proxy support, the version of the agent that is running on your EC2 instance must be 1.0.800.1 or later. If you enabled the auto-update process for the agent, you can verify that

your agent's version is 1.0.800.1 or later by using the Verifying that the Amazon Inspector agent is running (p. 39) procedure. If you didn't enable the auto-update process for the agent, you must install the agent on this EC2 instance again by following the Installing the agent on a Linux-based EC2 instance (p. 37) procedure.

2. Create a file called `awsagent.env`, and save it in the `/etc/init.d/` directory.
3. Edit `awsagent.env` to include these environment variables in the following format:

   - `export https_proxy=hostname:port`
   - `export http_proxy=hostname:port`
   - `export no_proxy=169.254.169.254`

   Note
   Substitute values in the preceding examples with valid hostname and port number combinations only. Specify the IP address of the instance metadata endpoint (169.254.169.254) for the `no_proxy` variable.
4. Restart the agent by first stopping it using the following command:

   ```
   sudo /etc/init.d/awsagent restart
   ```

   Proxy settings are picked up and used by both the agent and the auto-update process.

Uninstalling the Amazon Inspector agent

To uninstall the agent

1. Sign in to your EC2 instance running a Linux-based operating system where you want to uninstall the agent.

   Note
   For more information about the operating systems that are supported for Amazon Inspector, see Amazon Inspector supported operating systems and Regions (p. 4).
2. To uninstall the agent, use one of the following commands:

   - On Amazon Linux, CentOS, and Red Hat, run the following command:

     sudo yum remove 'AwsAgent*'
   - On Ubuntu Server, run the following command:

     sudo apt-get purge 'awsagent*'

Working with Amazon Inspector agents on Windows-based operating systems

You can start, stop, and modify the behavior of Amazon Inspector agents. Sign in to your EC2 instance running a Windows-based operating system and perform any of the procedures in this chapter. For more information about the operating systems that are supported for Amazon Inspector, see Amazon Inspector supported operating systems and Regions (p. 4).

Important
The Amazon Inspector agent relies on Amazon EC2 instance metadata to function correctly. It accesses instance metadata using version 1 or version 2 of the Instance Metadata Service (IMDSv1or IMDSv2). See Instance Metadata and User Data to learn more about EC2 instance metadata and access methods.

Note
The commands in this chapter function in all AWS Regions that are supported by Amazon Inspector.

Topics

Starting or stopping an Amazon Inspector agent or verifying that the agent is running

To start, stop, or verify an agent

1. On your EC2 instance, choose **Start**, **Run**, and then enter **`services.msc`**.
2. If the agent is successfully running, two services are listed with their status set to **Started** or **Running** in the **Services** window: **AWS Agent Service** and **AWS Agent Updater Service**.
3. To start the agent, right-click **AWS Agent Service**, and then choose **Start**. If the service successfully starts, the status is updated to **Started** or **Running**.
4. To stop the agent, right-click **AWS Agent Service**, and then choose **Stop**. If the service successfully stops, the status is cleared (appears as blank). We don't recommend stopping the **AWS Agent Updater Service** because it disables the installation of all future enhancements and fixes to the agent.
5. To verify that the agent is installed and running, sign in to your EC2 instance, and open a command prompt using administrative permissions. Navigate to `C:/Program Files/Amazon Web Services/AWS Agent`, and then run the following command:

   AWSAgentStatus.exe

   This command returns the status of the currently running agent, or an error stating that the agent can't be contacted.

Modifying Amazon Inspector agent settings

After the Amazon Inspector agent is installed and running on your EC2 instance, you can modify the settings in the `agent.cfg` file to alter the agent's behavior. On Windows-based operating systems, the file is located in the `C:\ProgramData\Amazon Web Services\AWS Agent` directory. After you modify and save the `agent.cfg` file, you must stop and start the agent for the changes to take effect.

Important
We highly recommend that you modify the `agent.cfg` file only with the guidance of AWS Support.

Configuring proxy support for an Amazon Inspector agent

To get proxy support for an agent on a Windows-based operating system, use the `WinHTTP` proxy. To set up the `WinHTTP` proxy using the `netsh` utility, see Netsh Commands for Windows Hypertext Transfer Protocol (WINHTTP).

Important
Only HTTPS proxies are supported for Windows-based instances.

Complete one of the following procedures:

To install an agent on an EC2 instance that uses a proxy server

1. Download the following .exe file: `https://d1wk0tztpsntt1.cloudfront.net/windows/installer/latest/AWSAgentInstall.exe`
2. Open a command prompt window or PowerShell window (using administrative permissions). Navigate to the location where you saved the downloaded `AWSAgentInstall.exe`, and then run the following command:

   ```
   .\AWSAgentInstall.exe /install USEPROXY=1
   ```

To configure proxy support on an EC2 instance with a running agent

1. To configure proxy support, the version of the Amazon Inspector agent that is running on your EC2 instance must be 1.0.0.59 or later. If you enabled the auto-update process for the agent, you can verify that your agent's version is 1.0.0.59 or later by using the Starting or stopping an Amazon Inspector agent or verifying that the agent is running (p. 42) procedure. If you didn't enable the auto-update process for the agent, you must install the agent on this EC2 instance again by following the Installing the agent on a Windows-based EC2 instance (p. 38) procedure.
2. Open the registry editor (`regedit.exe`).
3. Navigate to the following registry key: `"HKEY_LOCAL_MACHINE/SOFTWARE/Amazon Web Services/AWS Agent Updater"`.
4. Inside this registry key, create a registry `DWORD(32bit)` value called `"UseProxy"`.
5. Double-click on the value, and set the value to 1.
6. Enter **`services.msc`**, locate the **AWS Agent Service** and the **AWS Agent Updater Service** in the **Services** window, and restart each process. After both processes have successfully restarted, run the `AWSAgentStatus.exe` file (see step 5 in Starting or stopping an Amazon Inspector agent or verifying that the agent is running (p. 42)). View the status of your agent and verify that it is using the configured proxy.

Uninstalling the Amazon Inspector agent

To uninstall the agent

1. Sign in to your EC2 instance running a Windows-based operating system where you want to uninstall the Amazon Inspector agent.

   Note
   For more information about the operating systems that are supported for Amazon Inspector, see Amazon Inspector supported operating systems and Regions (p. 4).
2. On your EC2 instance, navigate to **Control Panel**, **Add/Remove Programs**.
3. In the list of installed programs, choose **AWS Agent**, and then choose **Uninstall**.

(Optional) Verify the signature of the Amazon Inspector agent installation script on Linux-based operating systems

This topic describes the recommended process of verifying the validity of the Amazon Inspector agent's installations script for Linux-based operating systems.

Whenever you download an application from the internet, we recommend that you authenticate the identity of the software publisher and check that the application is not altered or corrupted since it was published. This protects you from installing a version of the application that contains a virus or other malicious code.

If after running the steps in this topic, you determine that the software for the Amazon Inspector agent is altered or corrupted, do NOT run the installation file. Instead, contact AWS Support.

Amazon Inspector agent files for Linux-based operating systems are signed using `GnuPG`, an open source implementation of the Pretty Good Privacy (OpenPGP) standard for secure digital signatures. `GnuPG` (also known as `GPG`) provides authentication and integrity checking through a digital signature. Amazon EC2 publishes a public key and signatures that you can use to verify the downloaded Amazon EC2 CLI tools. For more information about `PGP` and `GnuPG` (`GPG`), see http://www.gnupg.org.

The first step is to establish trust with the software publisher. Download the public key of the software publisher, check that the owner of the public key is who they claim to be, and then add the public key to your *keyring*. Your keyring is a collection of known public keys. After you establish the authenticity of the public key, you can use it to verify the signature of the application.

Topics

Installing the GPG tools

If your operating system is Linux or Unix, the GPG tools are likely already installed. To test whether the tools are installed on your system, type **gpg** at a command prompt. If the GPG tools are installed, you see a GPG command prompt. If the GPG tools are not installed, you see an error stating that the command cannot be found. You can install the GnuPG package from a repository.

To install GPG tools on Debian-based Linux

- From a terminal, run the following command: **apt-get install gnupg**.

To install GPG tools on Red Hat–based Linux

- From a terminal, run the following command: **yum install gnupg**.

Authenticating and importing the public key

The next step in the process is to authenticate the Amazon Inspector public key and add it as a trusted key in your `GPG` keyring.

To authenticate and import the Amazon Inspector public key

1. Obtain a copy of our public GPG build key by doing one of the following:

   - Download from https://d1wk0tztpsntt1.cloudfront.net/linux/latest/inspector.gpg.
   - Copy the key from the following text and paste it into a file called `inspector.key`. Make sure to include everything that follows:

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v2.0.18 (GNU/Linux)

mQINBFYDlfEBEADFpfNt/mdCtsmfDoga+PfHY9bdXAD68yhp2m9NyH3BOzle/MXI
8siNfoRgzDwuWnIaezHwwLWkDw2paRxp1NMQ9qRe8Phq0ewheLrQu95dwDgMcw90
gf9m1iKVHjdVQ9qNHlB2OFknPDxMDRHcrmlJYDKYCX3+MODEHnlK25tIH2KWezXP
FPSU+TkwjLRzSMYH1L8IwjFUIIi78jQS9a31R/cOl4zuC5fOVghYlSomLI8irfoD
JSa3csVRujSmOAf9o3beiMR/kNDMpgDOxgiQTu/Kh39cl6o8AKe+QKK48kqO7hra
h1dpzLbfeZEVU6dWMZtlUksG/zKxuzD6d8vXYH7Z+x09POPFALQCQQMC3WisIKgj
zJEFhXMCCQ3NLC3CeyMq3vP7MbVRBYE7t3d2uDREkZBgIf+mbUYfYPhrzy0qT9Tr
PgwcnUvDZuazxuuPzucZGOJ5kbptat3DcUpstjdkMGAId3JawBbps77qRZdA+swr
o9o3jbowgmf0y5ZS6KwvZnC6XyTAkXy2io7mSrAIRECrANrzYzfp5v7uD7w8Dk0X
1OrfOm1VufMzAyTu0YQGBWaQKzSB8tCkvFw54PrRuUTcV826XU7SIJNzmNQo58uL
bKyLVBSCVabfs0lkECIesq8PT9xMYfQJ421uATHyYUnFTU2TYrCQEab7oQARAQAB
tCdBbWF6b24gSW5zcGVjdG9yIDxpbnNwZWN0b3JAYW1hem9uLmNvbT6JAjgEEwEC
ACIFAlYDlfECGwMGCwkIBwMCBhUIAgkKCwQWAgMBAh4BAheAAAoJECR0CWBYNgQY
8yUP/2GpIl40f3mKBUiSTe0XQLvwiBCHmY+V9fOuKqDTinxssjEMCnz0vsKeCZF/
L35pwNa/oW0OJa8D7sCkKG+8LuyMpcPDyqptLrYPprUWtz2+qLCHgpWsrku7ateF
x4hWS0jUVeHPaBzI9V1NTHsCx9+nbpWQ5Fk+7VJI8hbMDY7NQx6fcse8WTlP/0r/
HIkKzzqQQaaOf5t9zc5DKwi+dFmJbRUyaq22xs8C81UODjHunhjHdZ21cnsgk91S
fviuaum9aR4/uVIYOTVWnjC5J3+VlczyUt5FaYrrQ5ov0dM+biTUXwve3X8Q85Nu
DPnO/+zxb7Jz3QCHXnuTbxZTjvvl60Oi8//uRTnPXjz4wZLwQfibgHmk1++hzND7
wOYA02Js6v5FZQlLQAod7q2wuA1pq4MroLXzziDfy/9ea8B+tzyxlmNVRpVZY4Ll
DOHyqGQhpkyV3drjjNZlEofwbfu7m6ODwsgMl5ynzhKklJzwPJFfB3mMc7qLi+qX
MJtEX8KJ/iVUQStHHAG7daL1bxpWSI3BRuaHsWbBGQ/mcHBgUUOQJyEp5LAdg9Fs
VP55gWtF7pIqifiqlcfgG0Ov+A3NmVbmiGKSZvfrc5KsF/k43rCGqDx1RV6gZvyI
LfO9+3sEIlNrsMib0KRLDeBt3EuDsaBZgOkqjDhgJUesqiCy
=iEhB
-----END PGP PUBLIC KEY BLOCK-----
```

2. At a command prompt in the directory where you saved **inspector.key**, use the following command to import the Amazon Inspector public key into your keyring:

```
gpg --import inspector.key
```

   The command returns results that are similar to the following:

```
gpg: key 58360418: public key "Amazon Inspector <inspector@amazon.com>" imported
                   gpg: Total number processed: 1
                   gpg:               imported: 1  (RSA: 1)
```

   Make a note of the key value; you need it in the next step. In the preceding example, the key value is `58360418`.

3. Verify the fingerprint by running the following command, replacing *key-value* with the value from the preceding step:

```
gpg --fingerprint key-value
```

   This command returns results similar to the following:

```
pub   4096R/58360418 2015-09-24
```

```
                Key fingerprint = DDA0 D4C5 10AE 3C20 6F46  6DC0 2474 0960 5836 0418
                uid                  Amazon Inspector <inspector@amazon.com>
```

Additionally, the fingerprint string should be identical to `DDA0 D4C5 10AE 3C20 6F46 6DC0 2474 0960 5836 0418`, as shown in the preceding example. Compare the key fingerprint that is returned to the one published on this page. They should match. If they don't match, don't install the Amazon Inspector agent installation script, and contact AWS Support.

Verify the signature of the package

After you install the `GPG` tools, authenticate and import the Amazon Inspector public key, and verify that the public key is trusted, you are ready to verify the signature of the installation script.

To verify the installation script signature

1. At a command prompt, run the following command to download the signature file for the installation script:

```
curl -O https://inspector-agent.amazonaws.com/linux/latest/install.sig
```

2. Verify the signature by running the following command at a command prompt in the directory where you saved `install.sig` and the Amazon Inspector installation file. Both files must be present.

```
gpg --verify ./install.sig
```

The output should look something like the following:

```
gpg: Signature made Thu 24 Sep 2015 03:19:09 PM UTC using RSA key ID 58360418
gpg: Good signature from "Amazon Inspector <inspector@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: DDA0 D4C5 10AE 3C20 6F46  6DC0 2474 0960 5836 0418
```

If the output contains the phrase `Good signature from "Amazon Inspector <inspector@amazon.com>"`, it means that the signature has successfully been verified, and you can proceed to run the Amazon Inspector installation script.

If the output includes the phrase `BAD signature`, check whether you performed the procedure correctly. If you continue to get this response, don't run the installation file that you downloaded previously, and contact AWS Support.

The following are details about the warnings you might see:

- **WARNING: This key is not certified with a trusted signature! There is no indication that the signature belongs to the owner.** This refers to your personal level of trust in your belief that you possess an authentic public key for Amazon Inspector. In an ideal world, you would visit an AWS office and receive the key in person. However, more often you download it from a website. In this case, the website is an AWS website.
- **gpg: no ultimately trusted keys found.** This means that the specific key is not "ultimately trusted" by you (or by other people whom you trust).

For more information, see http://www.gnupg.org.

(Optional) Verify the signature of the Amazon Inspector agent installation script on Windows-based operating systems

This topic describes the recommended process of verifying the validity of the Amazon Inspector agent's installations script for Windows-based operating systems.

Whenever you download an application from the internet, we recommend that you authenticate the identity of the software publisher and check that the application is not altered or corrupted since it was published. This protects you from installing a version of the application that contains a virus or other malicious code.

If after running the steps in this topic, you determine that the software for the Amazon Inspector agent is altered or corrupted, do NOT run the installation file. Instead, contact AWS Support.

To verify the validity of the downloaded agent installation script on Windows-based operating systems, make sure that the thumbprint of its Amazon Services LLC signer certificate is equal to this value:

16 67 49 A7 B8 CC 5B 8A 57 1D DF 4B 7A 37 9D B1 6A 5E 65 80

To verify this value, perform the following procedure:

1. Right-click the downloaded `AWSAgentInstall.exe`, and open the **Properties** window.
2. Choose the **Digital Signatures** tab.
3. From the **Signature List**, choose **Amazon Services LLC**, and then choose **Details**.
4. Choose the **General** tab, if not already selected, and then choose **View Certificate**.
5. Choose the **Details** tab, and then choose **All** in the **Show** dropdown list, if not already selected.
6. Scroll down until you see the **Thumbprint** field and then choose **Thumbprint**. This displays the entire thumbprint value in the lower window.

   - If the thumbprint value in the lower window is identical to the following value:

     16 67 49 A7 B8 CC 5B 8A 57 1D DF 4B 7A 37 9D B1 6A 5E 65 80

     then your downloaded agent installation script is authentic and can be safely installed.
   - If the thumbprint value in the lower details window is not identical to the value above, do not run `AWSAgentInstall.exe`.

Amazon Inspector assessment targets

You can use Amazon Inspector to evaluate whether your AWS assessment targets (your collections of AWS resources) have potential security issues that you should address.

Important
Currently, your assessment targets can consist only of EC2 instances that run on supported operating systems. For information about supported operating systems and supported AWS Regions, see the section called “Supported operating systems and Regions” (p. 4).

Note
For information about launching EC2 instances, see the Amazon Elastic Compute Cloud documentation.

Topics

Tagging resources to create an assessment target

To create an assessment target for Amazon Inspector to assess, you start by tagging the EC2 instances that you want to include in your target. Tags are words or phrases that act as metadata for identifying and organizing your instances and other AWS resources. Amazon Inspector uses the tags that you create to identify the instances that belong to your target.

Every AWS tag consists of a key and value pair of your choice. For example, you might choose to name your key "Name" and your value "MyFirstInstance". After you tag your instances, you use the Amazon Inspector console to add the instances to your assessment target. It is not necessary that any instance match more than one tag key-value pair.

When you tag your EC2 instances to build assessment targets, you can create your own custom tag keys or use tag keys created by others in the same AWS account. You can also use the tag keys that AWS automatically creates. For example, AWS automatically creates a **Name** tag key for the EC2 instances that you launch.

You can add tags to EC2 instances when you create them, or you can add, change, or remove those tags one at a time on the console page for each EC2 instance. You can also add tags to multiple EC2 instances at once using the Tag Editor.

For more information, see Tag Editor. For more information about tagging EC2 instances, see Resources and Tags.

Amazon Inspector assessment target limits

You can create up to 50 assessment targets per AWS account. For more information, see Amazon Inspector service limits (p. 3).

Creating an assessment target

You can use the Amazon Inspector console to create assessment targets.

To create an assessment target

1. Sign in to the AWS Management Console and open the Amazon Inspector console at https://console.aws.amazon.com/inspector/.
2. In the navigation pane, choose **Assessment Targets**, and then choose **Create**.
3. For **Name**, enter a name for your assessment target.
4. Do one of the following:
   - To include all EC2 instances in this AWS account and Region in this assessment target, select the **All instances** check box.

     Note
     The limit on the maximum number of agents that you can include in an assessment run applies when you use this option. For more information, see Amazon Inspector service limits (p. 3).
   - To choose the EC2 instances that you want to include in this assessment target, for **Use Tags**, enter the tag key names and key-value pairs.
5. (Optional) While creating a target, you can select the **Install Agents** check box to install the agent on all EC2 instances in this target. To use this option, your EC2 instances must have the SSM Agent installed and an IAM role that allows Run Command. The SSM Agent is installed, by default, on Amazon EC2 Windows instances and Amazon Linux instances. Amazon EC2 Systems Manager requires an IAM role for EC2 instances that process commands and a separate role for users that execute commands. For more information, see Installing and Configuring SSM Agent and Configuring Security Roles for System Manager.

   Important
   If an EC2 instance already has an agent running on it, using this option replaces the agent currently running on the instance with the latest agent version.

   Note
   For your existing assessment targets, you can choose the **Install Agents with Run Command button** to install the agent on all EC2 instances in this target.

   Note
   You can also install the agent on multiple EC2 instances (both Linux-based and Windows-based instances with the same command) remotely by using the Systems Manager Run Command. For more information, see Installing the Amazon Inspector Agent on Multiple EC2 Instances Using the Systems Manager Run Command (p. 37).
6. Choose **Save**.

Note
You can use the **Preview Target** button on the **Assessment Targets** page to review all EC2 instances included in the assessment target. For each EC2 instance, you can review the hostname, instance ID, IP address, and, if applicable, the status of the agent. The agent status can have the following values: **HEALTHY**, **UNHEALTHY**, and **UNKNOWN**. Amazon Inspector displays an **UNKNOWN** status when it can't determine whether there is an agent running on the EC2 instance.

Deleting an assessment target

To delete an assessment target, perform the following procedure.

To delete an assessment target

- On the **Assessment targets** page, choose the target that you want to delete, and then choose **Delete**. When prompted for confirmation, choose **Yes**.

  Important
  When you delete an assessment target, all assessment templates, assessment runs, findings, and versions of the reports that are associated with the target are also deleted.

You can also delete an assessment target by using the `DeleteAssessmentTarget` API.

Amazon Inspector rules packages and rules

You can use Amazon Inspector to assess your assessment targets (collections of AWS resources) for potential security issues and vulnerabilities. Amazon Inspector compares the behavior and the security configuration of the assessment targets to selected security *rules packages*. In the context of Amazon Inspector, a *rule* is a security check that Amazon Inspector performs during the assessment run.

In Amazon Inspector, rules are grouped into distinct *rules packages* either by category, severity, or pricing. This gives you choices for the kinds of analysis that you can perform. For example, Amazon Inspector offers a large number of rules that you can use to assess your applications. But you might want to include a smaller subset of the available rules to target a specific area of concern or to uncover specific security problems. Companies with large IT departments might want to determine whether their application is exposed to any security threat. Others might want to focus only on issues with the severity level of **High**.

Severity levels for rules in Amazon Inspector

Each Amazon Inspector rule has an assigned severity level. This reduces the need to prioritize one rule over another in your analysis. It can also help you determine your response when a rule highlights a potential problem.

High, **Medium**, and **Low** levels all indicate a security issue that can result in compromised information confidentiality, integrity, and availability within your assessment target. The levels are distinguished by how likely the issue is to result in a compromise and how urgent it is to fix the issue.

The **Informational** level simply highlights a security configuration detail of your assessment target.

Here are the recommended ways to respond to issues based on their severity:

- **High** – High severity issues are extremely urgent. Amazon Inspector recommends that you treat this security issue as an emergency and implement an immediate remediation.
- **Medium** – Medium severity issues are somewhat urgent. Amazon Inspector recommends that you fix this issue at the next possible opportunity, for example, during your next service update.
- **Low** – Low severity issues are less urgent. Amazon Inspector recommends that you fix this issue as part of one of your future service updates.
- **Informational** – These issues are purely informational. Based on your business and organization goals, you can either simply make note of this information or use it to improve the security of your assessment target.

Rules packages in Amazon Inspector

An Amazon Inspector assessment can use any combination of the following rules packages:

Network assessments:

Host assessments:

- Common vulnerabilities and exposures (p. 54)
- Center for Internet Security (CIS) Benchmarks (p. 55)
- Security best practices for Amazon Inspector (p. 57)

Network Reachability

The rules in the Network Reachability package analyze your network configurations to find security vulnerabilities of your EC2 instances. The findings that Amazon Inspector generates also provide guidance about restricting access that is not secure.

The Network Reachability rules package uses the latest technology from the AWS Provable Security initiative.

The findings generated by these rules show whether your ports are reachable from the internet through an internet gateway (including instances behind Application Load Balancers or Classic Load Balancers), a VPC peering connection, or a VPN through a virtual gateway. These findings also highlight network configurations that allow for potentially malicious access, such as mismanaged security groups, ACLs, IGWs, and so on.

These rules help automate the monitoring of your AWS networks and identify where network access to your EC2 instances might be misconfigured. By including this package in your assessment run, you can implement detailed network security checks without having to install scanners and send packets, which are complex and expensive to maintain, especially across VPC peering connections and VPNs.

> **Important**
> An Amazon Inspector agent is not required to assess your EC2 instances with this rules package. However, an installed agent can provide information about the presence of any processes listening on the ports. Do not install an agent on an operating system that Amazon Inspector does not support. If an agent is present on an instance that runs an unsupported operating system, then the Network Reachability rules package will not work on that instance.

> **Important**
> This rules package does not support Amazon EC2 Classic networks.

For more information, see Amazon Inspector rules packages for supported operating systems (p. 78).

Configurations analyzed

Network Reachability rules analyze the configuration of the following entities for vulnerabilities:

- Amazon EC2 instances
- Application Load Balancers
- Direct Connect
- Elastic Load Balancers
- Elastic Network Interfaces
- Internet Gateways (IGWs)
- Network Access Control Lists (ACLs)
- Route Tables
- Security Groups (SGs)
- Subnets
- Virtual Private Clouds (VPCs)
- Virtual Private Gateways (VGWs)

- VPC peering connections

Reachability routes

Network Reachability rules check for the following reachability routes, which correspond to the ways in which your ports can be accessed from outside of your VPC:

- **`Internet`** - Internet gateways (including Application Load Balancers and Classic Load Balancers)
- **`PeeredVPC`** - VPC peering connections
- **`VGW`** - Virtual private gateways

Findings types

An assessment that includes the Network Reachability rules package can return the following types of findings for each reachability route:

- `RecognizedPort` (p. 53)
- `UnrecognizedPortWithListener` (p. 54)
- `NetworkExposure` (p. 54)

`RecognizedPort`

A port that is typically used for a well-known service is reachable. If an agent is present on the target EC2 instance, the generated finding will also indicate whether there is an active listening process on the port. Findings of this type are given a severity based on the security impact of the well-known service:

- **`RecognizedPortWithListener`** – A recognized port is externally reachable from the public internet through a specific networking component, and a process is listening on the port.
- **`RecognizedPortNoListener`** – A port is externally reachable from the public internet through a specific networking component, and there are no processes listening on the port.
- **`RecognizedPortNoAgent`** – A port is externally reachable from the public internet through a specific networking component. The presence of a process listening on the port can't be determined without installing an agent on the target instance.

The following table shows a list of recognized ports:

Service	TCP Ports	UDP Ports
SMB	445	445
NetBIOS	137, 139	137, 138
LDAP	389	389
LDAP over TLS	636	
Global catalog LDAP	3268	
Global catalog LDAP over TLS	3269	
NFS	111, 2049, 4045, 1110	111, 2049, 4045, 1110

Service	TCP Ports	UDP Ports
Kerberos	88, 464, 543, 544, 749, 751	88, 464, 749, 750, 751, 752
RPC	111, 135, 530	111, 135, 530
WINS	1512, 42	1512, 42
DHCP	67, 68, 546, 547	67, 68, 546, 547
Syslog	601	514
Print services	515	
Telnet	23	23
FTP	21	21
SSH	22	22
RDP	3389	3389
MongoDB	27017, 27018, 27019, 28017	
SQL Server	1433	1434
MySQL	3306	
PostgreSQL	5432	
Oracle	1521, 1630	
Elasticsearch	9300, 9200	
HTTP	80	80
HTTPS	443	443

`UnrecogizedPortWithListener`

A port that is not listed in the preceding table is reachable and has an active listening process on it. Because findings of this type show information about listening processes, they can be generated only when an Amazon Inspector agent is installed on the target EC2 instance. Findings of this type are given **Low** severity.

`NetworkExposure`

Findings of this type show aggregate information on the ports that are reachable on your EC2 instance. For each combination of elastic network interfaces and security groups on an EC2 instance, these findings show the reachable set of TCP and UDP port ranges. Findings of this type have the severity of **Informational**.

Common vulnerabilities and exposures

The rules in this package help verify whether the EC2 instances in your assessment targets are exposed to common vulnerabilities and exposures (CVEs). Attacks can exploit unpatched vulnerabilities to

compromise the confidentiality, integrity, or availability of your service or data. The CVE system provides a reference method for publicly known information security vulnerabilities and exposures. For more information, see https://cve.mitre.org/.

If a particular CVE appears in a *finding* that is produced by an Amazon Inspector assessment, you can search https://cve.mitre.org/ for the ID of the CVE (for example, **CVE-2009-0021**). The search results can provide detailed information about this CVE, its severity, and how to mitigate it.

The rules included in this package help you assess whether your EC2 instances are exposed to the CVEs in the following regional lists:

- US East (N. Virginia)
- US East (Ohio)
- US West (N. California)
- US West (Oregon)
- EU (Ireland)
- EU (Frankfurt)
- EU (London)
- EU (Stockholm)
- Asia Pacific (Tokyo)
- Asia Pacific (Seoul)
- Asia Pacific (Mumbai)
- Asia Pacific (Sydney)
- AWS GovCloud West (US)
- AWS GovCloud East (US)

The CVE rules package is updated regularly; this list includes the CVEs that are included in assessments runs that occur at the same time that this list is retrieved.

For more information, see Amazon Inspector rules packages for supported operating systems (p. 78).

Center for Internet Security (CIS) Benchmarks

The CIS Security Benchmarks program provides well-defined, unbiased, consensus-based industry best practices to help organizations assess and improve their security. AWS is a CIS Security Benchmarks Member company. For a list of Amazon Inspector certifications, see the Amazon Web Services page on the CIS website.

Amazon Inspector currently provides the following CIS Certified rules packages to help establish secure configuration postures for the following operating systems:

Amazon Linux

- `CIS Benchmark for Amazon Linux 2 Benchmark v1.0.0 Level 1`
- `CIS Benchmark for Amazon Linux 2 Benchmark v1.0.0 Level 2`
- `CIS Benchmark for Amazon Linux Benchmark v2.1.0 Level 1`
- `CIS Benchmark for Amazon Linux Benchmark v2.1.0 Level 2`
- `CIS Benchmark for Amazon Linux 2014.09-2015.03 v1.1.0 Level 1`

CentOS Linux

- CIS Benchmark for CentOS Linux 7 Benchmark v2.2.0 Level 1 Server
- CIS Benchmark for CentOS Linux 7 Benchmark v2.2.0 Level 2 Server
- CIS Benchmark for CentOS Linux 7 Benchmark v2.2.0 Level 1 Workstation
- CIS Benchmark for CentOS Linux 7 Benchmark v2.2.0 Level 2 Workstation
- CIS Benchmark for CentOS Linux 6 Benchmark v2.0.2 Level 1 Server
- CIS Benchmark for CentOS Linux 6 Benchmark v2.0.2 Level 2 Server
- CIS Benchmark for CentOS Linux 6 Benchmark v2.0.2 Level 1 Workstation
- CIS Benchmark for CentOS Linux 6 Benchmark v2.0.2 Level 2 Workstation

Red Hat Enterprise Linux

- CIS Benchmark for Red Hat Enterprise Linux 7 Benchmark v2.1.1 Level 1 Server
- CIS Benchmark for Red Hat Enterprise Linux 7 Benchmark v2.1.1 Level 2 Server
- CIS Benchmark for Red Hat Enterprise Linux 7 Benchmark v2.1.1 Level 1 Workstation
- CIS Benchmark for Red Hat Enterprise Linux 7 Benchmark v2.1.1 Level 2 Workstation
- CIS Benchmark for Red Hat Enterprise Linux 6 Benchmark v2.0.2 Level 1 Server
- CIS Benchmark for Red Hat Enterprise Linux 6 Benchmark v2.0.2 Level 2 Server
- CIS Benchmark for Red Hat Enterprise Linux 6 Benchmark v2.0.2. Level 1 Workstation
- CIS Benchmark for Red Hat Enterprise Linux 6 Benchmark v2.0.2 Level 2 Workstation

Ubuntu

- CIS Benchmark for Ubuntu Linux 18.04 LTS Benchmark v1.0.0 Level 1 Server
- CIS Benchmark for Ubuntu Linux 18.04 LTS Benchmark v1.0.0 Level 2 Server
- CIS Benchmark for Ubuntu Linux 18.04 LTS Benchmark v1.0.0 Level 1 Workstation
- CIS Benchmark for Ubuntu Linux 18.04 LTS Benchmark v1.0.0 Level 2 Workstation
- CIS Benchmark for Ubuntu Linux 16.04 LTS Benchmark v1.1.0 Level 1 Server
- CIS Benchmark for Ubuntu Linux 16.04 LTS Benchmark v1.1.0 Level 2 Server
- CIS Benchmark for Ubuntu Linux 16.04 LTS Benchmark v1.1.0 Level 1 Workstation
- CIS Benchmark for Ubuntu Linux 16.04 LTS Benchmark v1.1.0 Level 2 Workstation
- CIS Benchmark for Ubuntu Linux 14.04 LTS Benchmark v2.0.0 Level 1 Server
- CIS Benchmark for Ubuntu Linux 14.04 LTS Benchmark v2.0.0 Level 2 Server
- CIS Benchmark for Ubuntu Linux 14.04 LTS Benchmark v2.0.0 Level 1 Workstation
- CIS Benchmark for Ubuntu Linux 14.04 LTS Benchmark v2.0.0 Level 2 Workstation

Windows

- Windows Server 2016 (CIS Benchmark for Microsoft Windows 2016 RTM (Release 1607), v1.1.0, Level 1 Member Server Profile)
- Windows Server 2016 (CIS Benchmark for Microsoft Windows 2016 RTM (Release 1607), v1.1.0, Level 2 Member Server Profile)
- Windows Server 2016 (CIS Benchmark for Microsoft Windows 2016 RTM (Release 1607), v1.1.0, Level 1 Domain Controller Profile)

- Windows Server 2016 (CIS Benchmark for Microsoft Windows 2016 RTM (Release 1607), v1.1.0, Level 2 Domain Controller Profile)
- Windows Server 2016 (CIS Benchmark for Microsoft Windows 2016 RTM (Release 1607), v1.1.0, Next Generation Windows Security Profile)
- Windows Server 2012 R2 (CIS Benchmark for Microsoft Windows 2012 R2, v2.2.0, Level 1 Domain Controller Profile)
- Windows Server 2012 R2 (CIS Benchmark for Microsoft Windows 2012 R2, v2.2.0, Level 2 Domain Controller Profile)
- Windows Server 2012 R2 (CIS Benchmark for Microsoft Windows 2012 R2, v2.2.0, Level 1 Member Server Profile)
- Windows Server 2012 R2 (CIS Benchmark for Microsoft Windows 2012 R2, v2.2.0, Level 2 Member Server Profile)
- Windows Server 2012 (CIS Benchmark for Microsoft Windows 2012 non-R2, v2.0.0, Level 1 Member Server Profile)
- Windows Server 2012 (CIS Benchmark for Microsoft Windows 2012 non-R2, v2.0.0, Level 2 Member Server Profile)
- Windows Server 2012 (CIS Benchmark for Microsoft Windows 2012 non-R2, v2.0.0, Level 1 Domain Controller Profile)
- Windows Server 2012 (CIS Benchmark for Microsoft Windows 2012 non-R2, v2.0.0, Level 2 Domain Controller Profile)
- Windows Server 2008 R2 (CIS Benchmark for Microsoft Windows 2008 R2, v3.0.0, Level 1 Domain Controller Profile)
- Windows Server 2008 R2 (CIS Benchmark for Microsoft Windows 2008 R2, v3.0.0, Level 1 Member Server Profile)

If a specific CIS benchmark appears in a finding that is produced by an Amazon Inspector assessment run, you can download a detailed PDF description of the benchmark from https://benchmarks.cisecurity.org/ (free registration required). The benchmark document provides detailed information about this CIS benchmark, its severity, and how to mitigate it.

For more information, see Amazon Inspector rules packages for supported operating systems (p. 78).

Security best practices for Amazon Inspector

Use Amazon Inspector rules to help determine whether your systems are configured securely.

> **Important**
> Currently, you can include in your assessment targets EC2 instances that are running either Linux-based or Windows-based operating systems.
> During an assessment run, the rules described in this section generate findings **only** for the EC2 instances that are running Linux-based operating systems. The rules do not generate findings for EC2 instances that are running Windows-based operating systems.
> For more information, see Amazon Inspector rules packages for supported operating systems (p. 78).

Topics

- Configure password complexity (p. 59)
- Enable ASLR (p. 60)
- Enable DEP (p. 60)
- Configure permissions for system directories (p. 61)

Disable root login over SSH

This rule helps determine whether the SSH daemon is configured to permit logging in to your EC2 instance as root .

Severity

Medium (p. 51)

Finding

There is an EC2 instance in your assessment target that is configured to allow users to log in with root credentials over SSH. This increases the likelihood of a successful brute-force attack.

Resolution

We recommend that you configure your EC2 instance to prevent root account logins over SSH. Instead, log in as a non-root user and use `sudo` to escalate privileges when necessary. To disable SSH root account logins, set `PermitRootLogin` to `no` in the `/etc/ssh/sshd_config` file, and then restart `sshd`.

Support SSH version 2 only

This rule helps determine whether your EC2 instances are configured to support SSH protocol version 1.

Severity

Medium (p. 51)

Finding

An EC2 instance in your assessment target is configured to support SSH-1, which contains inherent design flaws that greatly reduce its security.

Resolution

We recommend that you configure EC2 instances in your assessment target to support only SSH-2 and later. For OpenSSH, you can achieve this by setting `Protocol 2` in the `/etc/ssh/sshd_config` file. For more information, see `man sshd_config`.

Disable password authentication Over SSH

This rule helps determine whether your EC2 instances are configured to support password authentication over the SSH protocol.

Severity

Medium (p. 51)

Finding

An EC2 instance in your assessment target is configured to support password authentication over SSH. Password authentication is susceptible to brute-force attacks and should be disabled in favor of key-based authentication where possible.

Resolution

We recommend that you disable password authentication over SSH on your EC2 instances and enable support for key-based authentication instead. This significantly reduces the likelihood of a successful brute-force attack. For more information, see https://aws.amazon.com/articles/1233/. If password authentication is supported, it is important to restrict access to the SSH server to trusted IP addresses.

Configure password maximum age

This rule helps determine whether the maximum age for passwords is configured on your EC2 instances.

Severity

Medium (p. 51)

Finding

An EC2 instance in your assessment target is not configured for a maximum age for passwords.

Resolution

If you are using passwords, we recommend that you configure a maximum age for passwords on all EC2 instances in your assessment target. This requires users to regularly change their passwords and reduces the chances of a successful password guessing attack. To fix this issue for existing users, use the **chage** command. To configure a maximum age for passwords for all future users, edit the `PASS_MAX_DAYS` field in the `/etc/login.defs` file.

Configure password minimum length

This rule helps determine whether a minimum length for passwords is configured on your EC2 instances.

Severity

Medium (p. 51)

Finding

An EC2 instance in your assessment target is not configured for a minimum length for passwords.

Resolution

If you are using passwords, we recommend that you configure a minimum length for passwords on all EC2 instances in your assessment target. Enforcing a minimum password length reduces the risk of a successful password guessing attack. You can do this by using the following option in the `pwquality.conf` file: `minlen`. For more information, see see https://linux.die.net/man/5/pwquality.conf.

If `pwquality.conf` is not available on your instance, you can set the `minlen` option using the `pam_cracklib.so` module. For more information, see `man pam_cracklib`.

The `minlen` option should be set to 14 or greater.

Configure password complexity

This rule helps determine whether a password complexity mechanism is configured on your EC2 instances.

Severity

Medium (p. 51)

Finding

No password complexity mechanism or restrictions are configured on EC2 instances in your assessment target. This allows users to set simple passwords, which increases the chances of unauthorized users gaining access and misusing accounts.

Resolution

If you are using passwords, we recommend that you configure all EC2 instances in your assessment target to require a level of password complexity. You can do this by using the following options in the `pwquality.conf` file: `lcredit`, `ucredit`, `dcredit`, and `ocredit`. For more information, see https://linux.die.net/man/5/pwquality.conf .

If `pwquality.conf` is not available on your instance, you can set the `lcredit`, `ucredit`, `dcredit`, and `ocredit` options using the `pam_cracklib.so` module. For more information, see `man pam_cracklib`.

The expected value for each of these options is less than or equal to -1, as shown below:

```
lcredit <= -1, ucredit <= -1, dcredit<= -1, ocredit <= -1
```

Additionally, the `remember` option must be set to 12 or greater. For more information, see `man pam_unix`.

Enable ASLR

This rule helps determine whether address space layout randomization (ASLR) is enabled on the operating systems of the EC2 instances in your assessment target.

Severity

Medium (p. 51)

Finding

An EC2 instance in your assessment target does not have ASLR enabled.

Resolution

To improve the security of your assessment target, we recommend that you enable ASLR on the operating systems of all EC2 instances in your target by running **echo 2 | sudo tee /proc/sys/kernel/randomize_va_space**.

Enable DEP

This rule helps determine whether Data Execution Prevention (DEP) is enabled on the operating systems of the EC2 instances in your assessment target.

Note
This rule is not supported for EC2 instances with ARM processors.

Severity

Medium (p. 51)

Finding

An EC2 instance in your assessment target does not have DEP enabled.

Resolution

We recommend that you enable DEP on the operating systems of all EC2 instances in your assessment target. Enabling DEP protects your instances from security compromises using buffer-overflow techniques.

Configure permissions for system directories

This rule checks permissions on system directories that contain binaries and system configuration information. It checks that only the root user (a user who logs in by using root account credentials) has write permissions for these directories.

Severity

High (p. 51)

Finding

An EC2 instance in your assessment target contains a system directory that is writable by non-root users.

Resolution

To improve the security of your assessment target and to prevent privilege escalation by malicious local users, configure all system directories on all EC2 instances in your target to be writable only by users who log in by using root account credentials.

Amazon Inspector assessment templates and assessment runs

Amazon Inspector helps you discover potential security issues by using security rules to analyze your AWS resources. Amazon Inspector monitors and collects behavioral data (telemetry) about your resources. The data includes information about the use of secure channels, network traffic among running processes, and details of communication with AWS services. Next, Amazon Inspector analyzes and compares the data against a set of security rules packages. Finally, Amazon Inspector produces a list of *findings* that identify potential security issues of various levels of severity.

To get started, you create an *assessment target* (a collection of the AWS resources that you want Amazon Inspector to analyze). Next, you create an *assessment template* (a blueprint that you use to configure your assessment). You use the template to start an *assessment run*, which is the monitoring and analysis process that results in a set of findings.

Topics

Amazon Inspector assessment templates

An assessment template allows you to specify a configuration for your assessment runs, including the following:

- Rules packages that Amazon Inspector uses to evaluate your assessment target
- Duration of the assessment run – You can set the duration of an assessment run anywhere between 3 minutes to 24 hours. We recommend setting the duration of assessment runs to 1 hour.
- Amazon SNS topics that Amazon Inspector sends notifications to about your assessment run states and findings
- Amazon Inspector attributes (key-value pairs) that you can assign to findings that are generated by the assessment run that uses this assessment template

After Amazon Inspector creates the assessment template, you can tag it like any other AWS resource. For more information, see Tag Editor. Tagging assessment templates enables you to organize them and get better oversight of your security strategy. For example, Amazon Inspector offers a large number of rules that you can assess your assessment targets against. You might want to include various subsets of the available rules in your assessment templates to target specific areas of concern or to uncover specific security issues. Tagging assessment templates allows you to locate and run them quickly at any time in accordance with your security strategy and goals.

Important
After you create an assessment template, you can't modify it.

Amazon Inspector assessment templates limits

You can create up to 500 assessment templates for each AWS account.

For more information, see Amazon Inspector service limits (p. 3).

Creating an assessment template

To create an assessment template

1. Sign in to the AWS Management Console and open the Amazon Inspector console at https://console.aws.amazon.com/inspector/.
2. In the navigation pane, choose **Assessment Templates**, and then choose **Create**.
3. For **Name**, enter a name for your assessment template.
4. For **Target name**, choose an assessment target to analyze.

   Note
   When you create an assessment template, you can use the **Preview Target** button on the **Assessment Templates** page to review all EC2 instances included in the assessment target. For each EC2 instance, you can review the hostname, instance ID, IP address, and, if applicable, the status of the agent. The agent status can have the following values: **HEALTHY**, **UNHEALTHY**, and **UNKNOWN**. Amazon Inspector displays an **UNKNOWN** status when it can't determine whether there is an agent running on the EC2 instance.
   You can also use the **Preview Target** button on the **Assessment Templates** page to review EC2 instances that make up assessment targets included in your previously created templates.
5. For **Rules packages**, choose one or more rules packages to include in your assessment template.
6. For **Duration**, specify the duration for your assessment template.
7. (Optional) For **SNS topics**, specify an SNS topic that you want Amazon Inspector to send notifications to about assessment run states and findings. Amazon Inspector can send SNS notifications about the following events:

   - An assessment run has started
   - An assessment run has ended
   - An assessment run's status has changed
   - A finding was generated

   For more information about setting up an SNS topic, see Setting up an SNS topic for Amazon Inspector notifications (p. 66).
8. (Optional) For **Tag**, enter values for **Key** and **Value**. You can add multiple tags to the assessment template.
9. (Optional) For **Attributes added to findings**, enter values for **Key** and **Value**. Amazon Inspector applies the attributes to all findings that are generated by the assessment template. You can add multiple attributes to the assessment template. For more information about findings and tagging findings, see Amazon Inspector findings (p. 68).
10. (Optional) To set up a schedule for your assessment runs using this template, select the **Set up recurring assessment runs once every <number_of_days>, starting now** check box and specify the recurrence pattern (number of days) using the up and down arrows.

    Note
    When you use this check box, Amazon Inspector automatically creates an Amazon CloudWatch Events rule for the assessment runs schedule that you are

setting up. Amazon Inspector then also automatically creates an IAM role named `AWS_InspectorEvents_Invoke_Assessment_Template`. This role enables CloudWatch Events to make API calls against the Amazon Inspector resources. For more information, see What is Amazon CloudWatch Events? and Using Resource-Based Policies for CloudWatch Events.

Note
You can also set up automatic assessment runs through an AWS Lambda function. For more information, see Setting up automatic assessment runs through a Lambda function (p. 65).

11. Choose **Create and run** or **Create**.

Deleting an assessment template

To delete an assessment template, perform the following procedure.

To delete an assessment template

- On the **Assessment Templates** page, choose the template that you want to delete, and then choose **Delete**. When prompted for confirmation, choose **Yes**.

  Important
  When you delete an assessment template, all assessment runs, findings, and versions of the reports associated with this template are also deleted.

You can also delete an assessment template by using the `DeleteAssessmentTemplate` API.

Assessment runs

After you create an assessment template, you can use it to start assessment runs. You can start multiple runs using the same template as long as you stay within the runs limit for each AWS account. For more information, see Amazon Inspector assessment runs limits (p. 65).

If you use the Amazon Inspector console, you must start the first run of your new assessment template from the **Assessment templates** page. After you start the run, you can use the **Assessment runs** page to monitor the run's progress. Use the **Run**, **Cancel**, and **Delete** buttons to start, cancel, or delete a run. You can also view the run's details, including the ARN of the run, the rules packages selected for the run, the tags and attributes that you applied to the run, and more.

For subsequent runs of the assessment template, you can use the **Run**, **Cancel**, and **Delete** buttons on either the **Assessment templates** page or the **Assessment runs** page.

Deleting an assessment run

To delete an assessment run, perform the following procedure.

To delete a run

- On the **Assessment runs** page, choose the run that you want to delete, and then choose **Delete**. When prompted for confirmation, choose **Yes**.

  Important
  When you delete a run, all findings and all versions of the report from that run are also deleted.

You can also delete a run by using the `DeleteAssessmentRun` API.

Amazon Inspector assessment runs limits

You can create up to 50,000 assessment runs for each AWS account.

You can have multiple runs occurring at the same time as long as the targets used for the runs don't contain overlapping EC2 instances.

For more information, see Amazon Inspector service limits (p. 3).

Setting up automatic assessment runs through a Lambda function

If you want to set up a recurring schedule for your assessment, you can configure your assessment template to run automatically by creating a Lambda function using the AWS Lambda console. For more information, see Lambda Functions.

To set up automatic assessment runs using the AWS Lambda console, perform the following procedure.

To set up automatic runs through a Lambda function

1. Sign in to the AWS Management Console, and open the AWS Lambda console.
2. In the navigation pane, choose either **Dashboard** or **Functions**, and then choose **Create a Lambda Function**.
3. On the **Create function** page, choose **Browse serverless app repository**, then enter **`inspector`** in the search field.
4. Choose the **inspector-scheduled-run** blueprint.
5. On the **Review, configure, and deploy** page, set up a recurring schedule for automated runs by specifying a CloudWatch event that triggers your function. To do this, enter a rule name and description, and then choose a schedule expression. The schedule expression determines how often the run occurs, for example, every 15 minutes or once a day. For more information about CloudWatch events and concepts, see What is Amazon CloudWatch Events?

   If you select the **Enable trigger** check box, the run begins immediately after you finish creating your function. Subsequent automated runs follow the recurrence pattern that you specify in the **Schedule expression** field. If you don't select the **Enable trigger** check box while creating the function, you can edit the function later to enable this trigger.
6. On the **Configure function** page, specify the following:

   - For **Name**, enter a name for your function.
   - (Optional) For **Description**, enter a description that will help you identify your function later.
   - For **runtime**, keep the default value of **`Node.js 8.10`**. AWS Lambda supports the **inspector-scheduled-run** blueprint only for the **`Node.js 8.10`** runtime.
   - The assessment template that you want to run automatically using this function. You do this by providing the value for the environment variable called **assessmentTemplateArn**.
   - Keep the handler set to the default value of **`index.handler`**.
   - The permissions for your function using the **Role** field. For more information, see AWS Lambda Permissions Model.

To run this function, you need an IAM role that allows AWS Lambda to start the runs and write log messages about the runs, including any errors, to Amazon CloudWatch Logs. AWS Lambda assumes this role for every recurring automated run. For example, you can attach the following sample policy to this IAM role:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "inspector:StartAssessmentRun",
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": "*"
    }
  ]
}
```

7. Review your selections, and then choose **Create function**.

Setting up an SNS topic for Amazon Inspector notifications

Amazon Simple Notification Service (Amazon SNS) is a web service that sends messages to subscribing endpoints or clients. You can use Amazon SNS to set up notifications for Amazon Inspector.

To set up an SNS topic for notifications

1. Create an SNS topic. See Tutorial: Creating an Amazon SNS Topic. When you create the topic, expand the **Access policy - optional** section. Then do the following to permit the assessment to send messages to the topic:

   a. For **Choose method**, choose **Basic**.

   b. For **Define who can publish messages to the topic**, choose **Only the specified AWS accounts**, and then enter the ARN for the account in the Region that you're creating the topic in:

   - `US East (Ohio)` - *arn:aws:iam::646659390643:root*
   - `US East (N. Virginia)` - *arn:aws:iam::316112463485:root*
   - `US West (N. California)` - *arn:aws:iam::166987590008:root*
   - `US West (Oregon)` - *arn:aws:iam::758058086616:root*
   - `Asia Pacific (Mumbai)` - *arn:aws:iam::162588757376:root*
   - `Asia Pacific (Seoul)` - *arn:aws:iam::526946625049:root*
   - `Asia Pacific (Sydney)` - *arn:aws:iam::454640832652:root*
   - `Asia Pacific (Tokyo)` - *arn:aws:iam::406045910587:root*
   - `Europe (Frankfurt)` - *arn:aws:iam::537503971621:root*
   - `Europe (Ireland)` - *arn:aws:iam::357557129151:root*
   - `Europe (London)` - *arn:aws:iam::146838936955:root*
   - `Europe (Stockholm)` - *arn:aws:iam::453420244670:root*
   - `AWS GovCloud (US-East)` - *arn:aws-us-gov:iam::206278770380:root*

- `AWS GovCloud (US-West)` - *arn:aws-us-gov:iam::850862329162:root*

c. For **Define who can subscribe to this topic**, choose **Only the specified AWS accounts**, and then enter the ARN for the account in the Region in which you're creating the topic.

d. To protect yourself against Inspector being used as a confused deputy as detailed in Confused deputy problem in the *IAM User Guide*, do the following:

i. Choose **Advanced**. This will navigate you to the JSON editor.

ii. Add the following condition:

```
"Condition": {
   "StringEquals": {
     "aws:SourceAccount": <your account Id here>,
     "aws:SourceArn": "arn:aws:inspector:*:*:*"
   }
 }
```

e. (Optional) For additional information about aws:SourceAccount and aws:SourceArn, see Global condition context keys in the *IAM User Guide.*

f. Update other settings for the topic as needed, and then choose **Create topic**.

2. (Optional) To create and encrypted SNS topic, see Encryption at rest in the *SNS Developer Guide*.

3. To protect yourself against Inspector being used as a confused deputy for your KMS key, follow the additional steps below:

a. Go to your CMK in the KMS console.

b. Choose **Edit**.

c. Add the following condition:

```
"Condition": {
   "StringEquals": {
     "aws:SourceAccount": <your account Id here>,
     "aws:SourceArn": "arn:aws:sns:*:*:*"
   }
 }
```

4. Create a subscription to the topic that you created. For more information, see Tutorial: Subscribing an Endpoint to an Amazon SNS Topic.

5. To confirm that the subscription is configured correctly, publish a message to the topic. For more information, see Tutorial: Publishing a Message to an Amazon SNS Topic.

Amazon Inspector findings

Findings are potential security issues that Amazon Inspector discovers during an assessment of your assessment target. Findings are displayed on the Amazon Inspector console or through the API. Findings contain detailed descriptions of the security issues and recommendations for resolving them.

After Amazon Inspector generates the findings, you can track them by assigning Amazon Inspector attributes to them. These attributes consist of key-value pairs.

Tracking your findings with attributes can be useful for managing the workflow of your security strategy. For example, after you create and run an assessment, it generates a list of findings of various levels of severity, urgency, and interest to you, based on your security goals and approach. You might want to follow one finding's recommendation steps right away to resolve a potentially urgent security issue. Or you might want to postpone resolving another finding until your next upcoming service update. For example, to track a finding to resolve right away, you can create and assign to a finding an attribute with a key-value pair of **`Status`** / **`Urgent`**. You could also use attributes to distribute the workload of resolving potential security issues. For example, to give Bob (who is a security engineer on your team) the task of resolving a finding, you can assign to a finding an attribute with a key-value pair of **`Assigned Engineer`** / **`Bob`**.

Working with findings

Complete the following procedure on any of the generated Amazon Inspector findings.

To locate, analyze, and assign attributes to findings

1. Sign in to the AWS Management Console and open the Amazon Inspector console at https://console.aws.amazon.com/inspector/.
2. After you run an assessment, navigate to the **Findings** page in the Amazon Inspector console to view your findings.

   You can also see your findings in the **Notable Findings** section on the **Dashboard** page of the Amazon Inspector console.

   > **Note**
   > You can't view the findings that are generated by an assessment run while it is still in progress. However, you can view a subset of findings if you stop the assessment before it completes its duration. In a production environment, we recommend that you let every assessment run through its entire duration so that it can produce a full set of findings.
3. To view the details of a specific finding, choose the **Expand** widget next to that finding. The details of the finding include the following:

   - Name of the assessment target that includes the EC2 instance where this finding was registered.
   - Name of the assessment template that was used to produce this finding.
   - Assessment run start time.
   - Assessment run end time.
   - Assessment run status.
   - Name of the rules package that includes the rule that triggered this finding.
   - Name of the finding.
   - Severity of the finding.

- Native severity details from the Common Vulnerability Scoring System (CVSS). These include CVSS vector and CVSS score metrics (including CVSS version 2.0 and 3.0) for the findings triggered by the rules in the Common Vulnerabilities and Exposures rules package. For details about the CVSS, see https://www.first.org/cvss/.
- Native severity details from the Center of Internet Security (CIS). These include the CIS weight metric for the findings triggered by the rules in the CIS Benchmarks package. For more information about CIS weight metric, see https://www.cisecurity.org/.
- Description of the finding.
- Recommended steps that you can complete to fix the potential security issue described by the finding.

4. To assign attributes to a finding, choose a finding, and then choose **Add/Edit Attributes**.

   You can also assign attributes to findings as you create an assessment template. To do that, you configure the new template to automatically assign attributes to all findings that are generated by the assessment run. You can use the **Key** and **Value** fields from the **Tags for findings from this assessment** field. For more information, see Amazon Inspector assessment templates and assessment runs (p. 62).
5. To export findings to a spreadsheet, choose the down arrow in the upper-right corner of the **Findings** page. In the dialog box, choose **Export all columns** or **Export visible columns**.

   Note that in the exported content, all datetime values are epoch timestamps.
6. To filter your current findings enter a single string you want to filter on, such as an instance ID or CVE number, in the filter bar above the findings table. To show or hide additional information columns, choose the settings icon in the upper-right corner of the **Findings** page.
7. To delete findings, navigate to the **Assessment runs** page and choose the run that resulted in the findings that you want to delete. Then choose **Delete**. When prompted for confirmation, choose **Yes**.

   Important
   You can't delete individual findings in Amazon Inspector. When you delete an assessment run, all findings and all versions of the report from that run are also deleted.

   You can also delete an assessment run by using the DeleteAssessmentRun API.

Assessment reports

An Amazon Inspector *assessment report* is a document that details what is tested in the assessment run and the results of the assessment. You can store the reports, share them with your team for remediation actions, or use them to augment your compliance audit data. You can generate a report for an assessment run after the run has successfully completed.

> **Note**
> You can generate reports only for assessment runs that occur after April 25, 2017, which is when assessment reports in Amazon Inspector became available.

You can view the following types of assessment reports:

- **Findings report** – this report contains the following information:
  - Summary of the assessment
  - EC2 instances evaluated during the assessment run
  - Rules packages included in the assessment run
  - Detailed information about each finding, including all EC2 instances that had the finding
- **Full report** – this report contains all the information that is included in a findings report, and additionally provides the list of rules that were checked against the instances in the assessment target.

To generate an assessment report

1. On the **Assessment runs** page, locate the assessment run that you want to generate a report for. Make sure that its status is set to **Analysis complete**.
2. Under the **Reports** column for this assessment run, choose the reports icon.

   > **Important**
   > The reports icon is present in the **Reports** column only for those assessment runs that took place or will take place after April 25, 2017. That is when assessment reports in Amazon Inspector became available.
3. In the **Assessment report** dialog box, choose the type of report that you want to view (either a **Findings** or a **Full** report) and the report format (HTML or PDF). Then choose **Generate report**.

You can also generate assessment reports through the GetAssessmentReport API.

To delete an assessment report, perform the following procedure.

To delete a report

- On the **Assessment runs** page, choose the run that the report that you want to delete is based on, and then choose **Delete**. When prompted for confirmation, choose **Yes**.

  > **Important**
  > In Amazon Inspector, you can't delete individual reports. When you delete an assessment run, all versions of the report from that run and all findings are also deleted.

  You can also delete an assessment run by using the DeleteAssessmentRun API.

Exclusions in Amazon Inspector

Exclusions are an output of Amazon Inspector assessment runs. Exclusions show which of your security checks can't be completed and how to resolve the issues. For example, issues can be caused by the absence of an agent on the specified target's EC2 instances, the use of an unsupported operating system, or unexpected errors.

You can view exclusions on the **Assessment runs** page on the console. For more information, see Viewing post-assessment exclusions (p. 77).

To avoid incurring unnecessary AWS fees, Amazon Inspector allows you to preview exclusions before running an assessment. You can find the previews on the **Assessment templates** page on the console. For more information, see Previewing exclusions (p. 76).

> **Note**
> You can generate post-assessment exclusions only for runs that occur after June 25, 2018. That's when exclusions in Amazon Inspector became available. However, exclusion previews are available for all assessment templates regardless of date.

Topics

Exclusion types

Amazon Inspector can produce the following exclusion types.

Exclus Type	Description	Recommenda								
No instances in target	There are no EC2 instances with the tags specified in the assessment target.	Check that the tags in your assessment target match the tags of your target EC2 instance.								
Agent is already running	An assessment run is already in progress on the target EC2 instance.	Wait until the current assessment run on the target EC2 instance has completed.								
Agent not found	An Amazon Inspector agent was	Install or reinstall an Amazon								

Exclus Type	Description	Recommenda								
	not found on the target EC2 instance.	Inspector agent on the target EC2 instance. For more information, see Installing Amazon Inspector agents (p. 36).								
Agent is unhealthy	The Amazon Inspector agent on the target EC2 instance is in an unhealthy state.	Check the status of the Amazon Inspector agent on this instance and take necessary action. For more information, see Inspector Agents.								
Unsupported OS version	The operating system of the target EC2 instance is not supported for Amazon Inspector assessments.	Remove the target EC2 instance from the assessment target, or create a target that doesn't include this instance. For a list of supported operating systems, see Amazon Inspector Supported Operating Systems and Regions.								

Exclus Type	Description	Recommenda								
Deprecated rules package	The assessment template includes a deprecated rules package.	Create an assessment template without the deprecated rules package, and use it for future assessment runs.								
Rules package not supported by OS	The operating system of the target EC2 instance is not supported by a rules package included in the assessment template.	Create an assessment template without the conflicting rules packages or remove the target EC2 instance from the assessment template. For a list of rules package support by operating system, see Rules Package Availability Across Supported Operating Systems.								
Rules evaluation error for single instance	An internal error has caused the rules evaluation to fail for this instance.	Attempt to run your assessment again. Contact support if the exclusion persists when you rerun the assessment.								

Exclusion Type	Description	Recommendation								
Rules evaluation error	An internal error has caused the rules evaluation to fail for your assessment.	Attempt to run the assessment again. Contact support if the exclusion persists when you rerun the assessment.								
Network Reachability error – internet	An internal error has caused a Network Reachability evaluation to fail on checks for ports reachable from the internet. You might get findings for other Network Reachability types.	Attempt to run the assessment again. Contact support if the exclusion persists when you rerun the assessment.								
Network Reachability error – internet through an Application Load Balancer	An internal error has caused a Network Reachability evaluation to fail on checks for ports reachable from the internet through an Application Load Balancer. You might get findings for other Network Reachability types.	Attempt to run the assessment again. Contact support if the exclusion persists when you rerun the assessment.								

Exclus Type	Description	Recommenda								
Network Reachability error – internet through an Elastic Load Balancing load balancer	An internal error has caused a Network Reachability evaluation to fail on checks for ports reachable from the internet though an Elastic Load Balancing load balancer. You might get findings for other Network Reachability types.	Attempt to run the assessment again. Contact support if the exclusion persists when you rerun the assessment.								
Network Reachability error –VPN	An internal error has caused a Network Reachability evaluation to fail on checks for ports reachable from VPN. You might get findings for other Network Reachability types.	Attempt to run the assessment again. Contact support if the exclusion persists when you rerun the assessment.								

Exclus Type	Description	Recommenda								
Network Reachability error – AWS Direct Connect	An internal error has caused a Network Reachability evaluation to fail on checks for ports reachable through AWS Direct Connect. You might get findings for other Network Reachability types.	Attempt to run the assessment again. Contact support if the exclusion persists when you rerun the assessment.								
Network Reachability error – VPC peering	An internal error has caused a Network Reachability evaluation to fail on checks for ports reachable from a peered VPC. You might get findings for other Network Reachability types.	Attempt to run the assessment again. Contact support if the exclusion persists when you rerun the assessment.								

Previewing exclusions

Amazon Inspector allows you to preview potential exclusions before running an assessment.

To preview assessment exclusions

1. Sign in to the AWS Management Console and open the Amazon Inspector console at https://console.aws.amazon.com/inspector/.
2. In the navigation pane, choose **Assessment templates**.
3. Expand a template, and in the **Assessment templates** section, choose **Preview exclusions**.
4. Review the descriptions of all detected exclusions and the recommendations for addressing them.

You can also list and describe exclusions by using the `ListExclusions` and `DescribeExclusions` operations.

Viewing post-assessment exclusions

After an assessment run, you can view details about any exclusions.

To view details about exclusions

1. Sign in to the AWS Management Console and open the Amazon Inspector console at https://console.aws.amazon.com/inspector/.
2. In the navigation pane, choose **Assessment runs**.
3. In the **Exclusions** column, choose the active link that is associated with an assessment run.
4. Review the descriptions of all detected exclusions and the recommendations for addressing them.

   You can also list and describe exclusions by using the `ListExclusions` and `DescribeExclusions` operations.

Amazon Inspector rules packages for supported operating systems

You can run Amazon Inspector rules packages on the EC2 instances that are included in your assessment targets. The following table shows the availability of rules packages for supported operating systems.

Important
You can run an agentless assessment with the Network Reachability (p. 52) rules package on any EC2 instance regardless of operating system.

Note
For more information about supported operating systems, see Amazon Inspector supported operating systems and Regions (p. 4).

Suppo Opera Syste	Common Vulnerabiliti and Exposures	CIS Benchmarks	Network Reachability	Security Best Practices	Runtime Behavior Analysis
Amazon Linux 2	Supported	Supported	Supported	Supported	Deprecated
Amazon Linux 2018.03	Supported	Supported	Supported	Supported	Deprecated
Amazon Linux 2017.09	Supported	Supported	Supported	Supported	Deprecated
Amazon Linux 2017.03	Supported	Supported	Supported	Supported	Deprecated
Amazon Linux 2016.09	Supported	Supported	Supported	Supported	Deprecated
Amazon Linux 2016.03	Supported	Supported	Supported	Supported	Deprecated
Amazon Linux 2015.09	Supported	Supported	Supported	Supported	Deprecated
Amazon Linux 2015.03	Supported	Supported	Supported	Supported	Deprecated
Amazon Linux 2014.09	Supported		Supported	Supported	

Suppo Opera System	Common Vulnerabiliti and Exposures	CIS Benchmarks	Network Reachability	Security Best Practices	Runtime Behavior Analysis
Amazon Linux 2014.03	Supported		Supported	Supported	
Amazon Linux 2013.09	Supported		Supported	Supported	
Amazon Linux 2013.03	Supported		Supported	Supported	
Amazon Linux 2012.09	Supported		Supported	Supported	
Amazon Linux 2012.03	Supported		Supported	Supported	
Ubuntu 20.04 LTS	Supported		Supported	Supported	
Ubuntu 18.04 LTS	Supported	Supported	Supported	Supported	Deprecated
Ubuntu 16.04 LTS	Supported	Supported	Supported	Supported	Deprecated
Ubuntu 14.04 LTS	Supported	Supported	Supported	Supported	Deprecated
Debian 10.x, 9.0 - 9.5, 8.0 - 8.7	Supported		Supported	Supported	
RHEL 8.x	Supported		Supported	Supported	
RHEL 7.6 - 7.x	Supported	Supported	Supported	Supported	

Suppo Opera System	Common Vulnerabiliti and Exposures	CIS Benchmarks	Network Reachability	Security Best Practices	Runtime Behavior Analysis
RHEL 6.2 - 6.9, 7.2 - 7.5	Supported	Supported	Supported	Supported	Deprecated
CentOS 7.6 - 7.X	Supported	Supported	Supported	Supported	
CentOS 6.2 - 6.9, 7.2 - 7.5	Supported	Supported	Supported	Supported	Deprecated
Windows Server 2019 Base	Supported		Supported		
Windows Server 2016 Base	Supported	Supported	Supported		Deprecated
Windows Server 2012 R2	Supported	Supported	Supported		Deprecated
Windows Server 2012	Supported	Supported	Supported		Deprecated
Windows Server 2008 R2	Supported	Supported	Supported		Deprecated

Logging Amazon Inspector API calls with AWS CloudTrail

Amazon Inspector is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Amazon Inspector. CloudTrail captures all API calls for Amazon Inspector as events, including calls from the Amazon Inspector console and code calls to the Amazon Inspector API operations. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Amazon Inspector. If you don't configure a trail, you can still view the most recent events on the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to Amazon Inspector, the IP address the request was made from, who made the request, when it was made, and more.

To learn more about CloudTrail, see the AWS CloudTrail User Guide. For a full list of Amazon Inspector API operations, see Actions in the *Amazon Inspector API Reference*.

Amazon Inspector information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in Amazon Inspector, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see Viewing Events with CloudTrail Event History.

For an ongoing record of events in your AWS account, including events for Amazon Inspector, create a *trail*. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail on the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- Overview for Creating a Trail
- CloudTrail Supported Services and Integrations
- Configuring Amazon SNS Notifications for CloudTrail
- Receiving CloudTrail Log Files from Multiple Regions and Receiving CloudTrail Log Files from Multiple Accounts

CloudTrail logs all Amazon Inspector operations, including read-only operations, such as `ListAssessmentRuns` and `DescribeAssessmentTargets`, and management operations, such as `AddAttributesToFindings` and `CreateAssessmentTemplate`.

> **Note**
> CloudTrail logs only the request information of Amazon Inspector read-only operations. Both request and response information is logged for all other Amazon Inspector operations.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or AWS Identity and Access Management (IAM) user credentials

- Whether the request was made with temporary security credentials for a role or federated user
- Whether the request was made by another AWS service

For more information, see CloudTrail userIdentity Element.

Understanding Amazon Inspector log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, and other request parameters. CloudTrail log files aren't an ordered stack trace of the public API calls, so they don't appear in any specific order.

The following example shows a CloudTrail log entry that demonstrates the Amazon Inspector `CreateResourceGroup` operation:


```
{
    "eventVersion": "1.03",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::444455556666:user/Alice",
        "accountId": "444455556666",
        "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2016-04-14T17:05:54Z"
            },
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::444455556666:user/Alice",
                "accountId": "444455556666",
                "userName": "Alice"
            }
        }
    },
    "eventTime": "2016-04-14T17:12:34Z",
    "eventSource": "inspector.amazonaws.com",
    "eventName": "CreateResourceGroup",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.251.233.179",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
        "resourceGroupTags": [
            {
                "key": "Name",
                "value": "ExampleEC2Instance"
            }
        ]
    },
    "responseElements": {
        "resourceGroupArn": "arn:aws:inspector:us-west-2:444455556666:resourcegroup/0-
oclRMp8B"
    },
    "requestID": "148256d2-0264-11e6-a9b5-b98a7d3b840f",
    "eventID": "e5ea533e-eede-46cc-94f6-0d08e6306ff0",
    "eventType": "AwsApiCall",
    "apiVersion": "v20160216",
```

```
    "recipientAccountId": "444455556666"
}
```

Monitoring Amazon Inspector using Amazon CloudWatch

You can monitor Amazon Inspector using Amazon CloudWatch, which collects and processes raw data into readable, near real-time metrics. By default, Amazon Inspector sends metric data to CloudWatch in 5-minute periods. You can use the AWS Management Console, the AWS CLI, or an API to view the metrics that Amazon Inspector sends to CloudWatch.

For more information about Amazon CloudWatch, see the Amazon CloudWatch User Guide.

Amazon Inspector CloudWatch metrics

The Amazon Inspector namespace includes the following metrics.

`AssessmentTargetARN` metrics:

Metric	Description			
`TotalMatchingAgents`	Number of agents that match this target			
`TotalHealthyAgents`	Number of agents that match this target that are healthy			
`TotalAssessmentRuns`	Number of assessment runs for this target			
`TotalAssessmentRunFindings`	Number of findings for this target			

`AssessmentTemplateARN` metrics:

Metric	Description			
`TotalMatchingAgents`	Number of agents that match this template			
`TotalHealthyAgents`	Number of agents that match this template that are healthy			
`TotalAssessmentRuns`	Number of assessment runs for this template			
`TotalAssessmentRunFindings`	Number of findings for this template			

Aggregate metrics

Metric	Description			
TotalAssessmentRuns	Number of assessment runs in this AWS account			

Configuring Amazon Inspector using AWS CloudFormation

For reference information about Amazon Inspector resources that are supported by AWS CloudFormation, see the following topics:

- `AWS::Inspector::AssessmentTarget`
- `AWS::Inspector::AssessmentTemplate`
- `AWS::Inspector::ResourceGroup`

Important
For lists of the ARNs of Amazon Inspector rules packages in supported AWS Regions, see Amazon Inspector ARNS for rules packages (p. 91).

Integration with AWS Security Hub

AWS Security Hub provides you with a comprehensive view of your security state in AWS and helps you to check your environment against security industry standards and best practices. Security Hub collects security data from across AWS accounts, services, and supported third-party partner products and helps you to analyze your security trends and identify the highest priority security issues.

The Amazon Inspector integration with Security Hub enables you to send findings from Amazon Inspector to Security Hub. Security Hub can then include those findings in its analysis of your security posture.

Contents

How Amazon Inspector sends findings to Security Hub

In Security Hub, security issues are tracked as findings. Some findings come from issues that are detected by other AWS services or by third-party partners. Security Hub also has a set of rules that it uses to detect security issues and generate findings.

Security Hub provides tools to manage findings from across all of these sources. You can view and filter lists of findings and view details for a finding. See Viewing findings in the *AWS Security Hub User Guide*. You can also track the status of an investigation into a finding. See Taking action on findings in the *AWS Security Hub User Guide*.

All findings in Security Hub use a standard JSON format called the AWS Security Finding Format (ASFF). The ASFF includes details about the source of the issue, the affected resources, and the current status of the finding. See AWS Security Finding Format (ASFF) in the *AWS Security Hub User Guide*.

Amazon Inspector is one of the AWS services that sends findings to Security Hub.

Types of findings that Amazon Inspector sends

Amazon Inspector sends all of the findings it generates to Security Hub.

Amazon Inspector sends the findings to Security Hub using the AWS Security Finding Format (ASFF). In ASFF, the `Types` field provides the finding type. Findings from Amazon Inspector can have the following values for `Types`.

- Software and Configuration Checks/Vulnerabilities/CVE

- Software and Configuration Checks/AWS Security Best Practices/Network Reachability
- Software and Configuration Checks/Industry and Regulatory Standards/CIS Host Hardening Benchmarks

Latency for sending findings

When Amazon Inspector creates a new finding, it is usually sent to Security Hub within five minutes.

Retrying when Security Hub is not available

If Security Hub is not available, Amazon Inspector retries sending the findings until they are received.

Updating existing findings in Security Hub

After it sends a finding to Security Hub, Amazon Inspector updates the finding to reflect additional observations of the finding activity. This will result in fewer Amazon Inspector findings in Security Hub than in Amazon Inspector.

Typical finding from Amazon Inspector

Amazon Inspector sends findings to Security Hub using the AWS Security Finding Format (ASFF).

Here is an example of a typical finding from Amazon Inspector.


```
{
  "SchemaVersion": "2018-10-08",
  "Id": "inspector/us-east-1/111122223333/629ff13fbbb44c872f7bba3e7f79f60cb6d443d8",
  "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/inspector",
  "GeneratorId": "arn:aws:inspector:us-east-1:316112463485:rulespackage/0-PmNV0Tcd",
  "AwsAccountId": "111122223333",
  "Types": [
    "Software and Configuration Checks/AWS Security Best Practices/Network Reachability -
 Recognized port reachable from internet"
  ],
  "CreatedAt": "2020-08-19T17:36:22.169Z",
  "UpdatedAt": "2020-11-04T16:36:06.064Z",
  "Severity": {
    "Label": "MEDIUM",
    "Normalized": 40,
    "Original": "6.0"
  },
  "Confidence": 10,
  "Title": "On instance i-0c10c2c7863d1a356, TCP port 22 which is associated with 'SSH' is
 reachable from the internet",
  "Description": "On this instance, TCP port 22, which is associated with SSH, is
 reachable from the internet. You can install the Inspector agent on this instance
 and re-run the assessment to check for any process listening on this port. The
 instance i-0c10c2c7863d1a356 is located in VPC vpc-a0c2d7c7 and has an attached ENI
 eni-078eac9d6ad9b20d1 which uses network ACL acl-154b8273. The port is reachable from the
 internet through Security Group sg-0af64c8a5eb30ca75 and IGW igw-e209d785",
  "Remediation": {
    "Recommendation": {
      "Text": "You can edit the Security Group sg-0af64c8a5eb30ca75 to remove access from
 the internet on port 22"
    }
```

```
  },
  "ProductFields": {
    "attributes/VPC": "vpc-a0c2d7c7",
    "aws/inspector/id": "Recognized port reachable from internet",
    "serviceAttributes/schemaVersion": "1",
    "aws/inspector/arn": "arn:aws:inspector:us-east-1:111122223333:target/0-8zh1cWkg/
template/0-rqtRV0u0/run/0-Ck2F6tY9/finding/0-B458MQWe",
    "attributes/ACL": "acl-154b8273",
    "serviceAttributes/assessmentRunArn": "arn:aws:inspector:us-
east-1:111122223333:target/0-8zh1cWkg/template/0-rqtRV0u0/run/0-Ck2F6tY9",
    "attributes/PROTOCOL": "TCP",
    "attributes/RULE_TYPE": "RecognizedPortNoAgent",
    "aws/inspector/RulesPackageName": "Network Reachability",
    "attributes/INSTANCE_ID": "i-0c10c2c7863d1a356",
    "attributes/PORT_GROUP_NAME": "SSH",
    "attributes/IGW": "igw-e209d785",
    "serviceAttributes/rulesPackageArn": "arn:aws:inspector:us-
east-1:111122223333:rulespackage/0-PmNV0Tcd",
    "attributes/SECURITY_GROUP": "sg-0af64c8a5eb30ca75",
    "attributes/ENI": "eni-078eac9d6ad9b20d1",
    "attributes/REACHABILITY_TYPE": "Internet",
    "attributes/PORT": "22",
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-east-1::product/aws/inspector/
inspector/us-east-1/111122223333/629ff13fbbb44c872f7bba3e7f79f60cb6d443d8",
    "aws/securityhub/ProductName": "Inspector",
    "aws/securityhub/CompanyName": "Amazon"
  },
  "Resources": [
    {
      "Type": "AwsEc2Instance",
      "Id": "arn:aws:ec2:us-east-1:193043430472:instance/i-0c10c2c7863d1a356",
      "Partition": "aws",
      "Region": "us-east-1",
      "Tags": {
        "Name": "kubectl"
      },
      "Details": {
        "AwsEc2Instance": {
          "ImageId": "ami-02354e95b39ca8dec",
          "IpV4Addresses": [
            "172.31.43.6"
          ],
          "VpcId": "vpc-a0c2d7c7",
          "SubnetId": "subnet-4975b475"
        }
      }
    }
  ],
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NEW"
  },
  "RecordState": "ACTIVE"
}
```


Enabling and configuring the integration

To use the integration with Security Hub, you must enable Security Hub. For information on how to enable Security Hub, see Setting up Security Hub in the *AWS Security Hub User Guide*.

When you enable both Amazon Inspector and Security Hub, the integration is enabled automatically. Amazon Inspector begins to send findings to Security Hub.

How to stop sending findings

To stop sending findings to Security Hub, you can use either the Security Hub console or the API.

See Disabling and enabling the flow of findings from an integration (console) or Disabling the flow of findings from an integration (Security Hub API, AWS CLI) in the *AWS Security Hub User Guide*.

Amazon Inspector ARNs

Each resource type and rules package in Amazon Inspector has a unique Amazon Resource Name (ARN) associated with it.

Contents

ARNs for Amazon Inspector resources

In Amazon Inspector, the primary resources are resource groups, assessment targets, assessment templates, assessment runs, and findings. These resources have unique Amazon Resource Names (ARNs) associated with them, as shown in the following table.

Resource Type	ARN Format
Resource group	`arn:aws:inspector:region:account-id:resourcegroup/ID`
Assessment target	`arn:aws:inspector:region:account-id:target/ID`
Assessment template	`arn:aws:inspector:region:account-id:target/ID:template:ID`
Assessment run	`arn:aws:inspector:region:account-id:target/ID/template/ID/run/ID`
Finding	`arn:aws:inspector:region:account-id:target/ID/template/ID/run/ID/finding/ID`

Amazon Inspector ARNS for rules packages

The following tables show the ARNs for Amazon Inspector rules packages in all supported Regions.

Topics

US East (Ohio)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	`arn:aws:inspector:us-east-2:646659390643:rulespackage/0-JnA8Zp85`
CIS Operating System Security Configuration Benchmarks	`arn:aws:inspector:us-east-2:646659390643:rulespackage/0-m8r61nnh`
Network Reachability	`arn:aws:inspector:us-east-2:646659390643:rulespackage/0-cE4kTR30`
Security Best Practices	`arn:aws:inspector:us-east-2:646659390643:rulespackage/0-AxKmMHPX`

US East (N. Virginia)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	`arn:aws:inspector:us-east-1:316112463485:rulespackage/0-gEjTy7T7`
CIS Operating System Security Configuration Benchmarks	`arn:aws:inspector:us-east-1:316112463485:rulespackage/0-rExsr2X8`
Network Reachability	`arn:aws:inspector:us-east-1:316112463485:rulespackage/0-PmNV0Tcd`

Rules Package Name	ARN
Security Best Practices	arn:aws:inspector:us-east-1:316112463485:rulespackage/0-R01qwB5Q

US West (N. California)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws:inspector:us-west-1:166987590008:rulespackage/0-TKgzoVOa
CIS Operating System Security Configuration Benchmarks	arn:aws:inspector:us-west-1:166987590008:rulespackage/0-xUY8iRqX
Network Reachability	arn:aws:inspector:us-west-1:166987590008:rulespackage/0-TxmXimXF
Security Best Practices	arn:aws:inspector:us-west-1:166987590008:rulespackage/0-byoQRFYm

US West (Oregon)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9
CIS Operating System Security Configuration Benchmarks	arn:aws:inspector:us-west-2:758058086616:rulespackage/0-H5hpSawc
Network Reachability	arn:aws:inspector:us-west-2:758058086616:rulespackage/0-rD1z6dpl
Security Best Practices	arn:aws:inspector:us-west-2:758058086616:rulespackage/0-JJOtZiqQ

Asia Pacific (Mumbai)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws:inspector:ap-south-1:162588757376:rulespackage/0-LqnJE9dO
CIS Operating System Security Configuration Benchmarks	arn:aws:inspector:ap-south-1:162588757376:rulespackage/0-PSUlX14m
Network Reachability	arn:aws:inspector:ap-south-1:162588757376:rulespackage/0-YxKfjFu1
Security Best Practices	arn:aws:inspector:ap-south-1:162588757376:rulespackage/0-fs0IZZBj

Asia Pacific (Seoul)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws:inspector:ap-northeast-2:526946625049:rulespackag PoGHMznc
CIS Operating System Security Configuration Benchmarks	arn:aws:inspector:ap-northeast-2:526946625049:rulespackag T9srhg1z
Network Reachability	arn:aws:inspector:ap-northeast-2:526946625049:rulespackag s3OmLzhL
Security Best Practices	arn:aws:inspector:ap-northeast-2:526946625049:rulespackag

Asia Pacific (Sydney)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws:inspector:ap-southeast-2:454640832652:rulespackag D5TGAxiR
CIS Operating System Security Configuration Benchmarks	arn:aws:inspector:ap-southeast-2:454640832652:rulespackag Vkd2Vxjq

Rules Package Name	ARN
Network Reachability	arn:aws:inspector:ap-southeast-2:454640832652:rulespackag FLcuV4Gz
Security Best Practices	arn:aws:inspector:ap-southeast-2:454640832652:rulespackag asL6HRgN

Asia Pacific (Tokyo)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws:inspector:ap-northeast-1:406045910587:rulespackag gHP9oWNT
CIS Operating System Security Configuration Benchmarks	arn:aws:inspector:ap-northeast-1:406045910587:rulespackag
Network Reachability	arn:aws:inspector:ap-northeast-1:406045910587:rulespackag YI95DVd7
Security Best Practices	arn:aws:inspector:ap-northeast-1:406045910587:rulespackag bBUQnxMq

Europe (Frankfurt)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws:inspector:eu-central-1:537503971621:rulespackage/ wNqHa8M9
CIS Operating System Security Configuration Benchmarks	arn:aws:inspector:eu-central-1:537503971621:rulespackage/ nZrAVuv8
Network Reachability	arn:aws:inspector:eu-central-1:537503971621:rulespackage/
Security Best Practices	arn:aws:inspector:eu-central-1:537503971621:rulespackage/ ZujVHEPB

Europe (Ireland)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws:inspector:eu-west-1:357557129151:rulespackage/0-ubA5XvBh
CIS Operating System Security Configuration Benchmarks	arn:aws:inspector:eu-west-1:357557129151:rulespackage/0-sJBhCr0F
Network Reachability	arn:aws:inspector:eu-west-1:357557129151:rulespackage/0-SPzU33xe
Security Best Practices	arn:aws:inspector:eu-west-1:357557129151:rulespackage/0-SnojL3Z6

Europe (London)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws:inspector:eu-west-2:146838936955:rulespackage/0-kZGCqcE1
CIS Operating System Security Configuration Benchmarks	arn:aws:inspector:eu-west-2:146838936955:rulespackage/0-IeCjwf1W
Network Reachability	arn:aws:inspector:eu-west-2:146838936955:rulespackage/0-AizSYyNq
Security Best Practices	arn:aws:inspector:eu-west-2:146838936955:rulespackage/0-XApUiSaP

Europe (Stockholm)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws:inspector:eu-north-1:453420244670:rulespackage/0-IgdgIewd
CIS Operating System Security Configuration Benchmarks	arn:aws:inspector:eu-north-1:453420244670:rulespackage/0-Yn8jlX7f

Rules Package Name	ARN
Network Reachability	arn:aws:inspector:eu-north-1:453420244670:rulespackage/0-
Security Best Practices	arn:aws:inspector:eu-north-1:453420244670:rulespackage/0-HfBQSbSf

AWS GovCloud (US-East)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws-us-gov:inspector:us-gov-east-1:206278770380:rulespackage/0-3
CIS Operating System Security Configuration Benchmarks	arn:aws-us-gov:inspector:us-gov-east-1:206278770380:rulespackage/0-pTLCdIww
Security Best Practices	arn:aws-us-gov:inspector:us-gov-east-1:206278770380:rulespackage/0-vlgEGcVD

AWS GovCloud (US-West)

Rules Package Name	ARN
Common Vulnerabilities and Exposures	arn:aws-us-gov:inspector:us-gov-west-1:850862329162:rulespackage/0-4
CIS Operating System Security Configuration Benchmarks	arn:aws-us-gov:inspector:us-gov-west-1:850862329162:rulespackage/0-Ac4CFOuc
Security Best Practices	arn:aws-us-gov:inspector:us-gov-west-1:850862329162:rulespackage/0-rOTGqe5G

Document history

The following table describes the documentation release history of Amazon Inspector after May 2018.

update-history-change	update-history-description	update-history-date
Updated security best practices for passwords (p. 98)	The Amazon Inspector security best practice requirements for EC2 instance password length and password complexity have been updated. See Configure password minimum length and Configure password complexity	March 8, 2021
Added support for newer operating system versions	Amazon Inspector now supports the following operating system versions: Ubunti 20.4 LTS, Debian 10.x, RHEL 8.x, and Windows Server 2019 Base.	October 15, 2020
Security information consolidated into a new security chapter (p. 98)	Security information for Amazon Inspector, including information on managing identity and access management, is consolidated into a security chapter. See Security in Amazon Inspector.	April 7, 2020
Updated documentation to remove support for the Runtime Behavior Analysis rules package. (p. 98)	Multiple topics were updated to remove information about the Runtime Behavior Analysis rules package, which is no longer supported.	September 5, 2019
Added OS Support (p. 98)	Added Amazon Inspector support for CentOS 7.6. For more information, see Amazon Inspector Supported Operating Systems and Regions and Rules Packages Availability Across Supported Operating Systems.	December 3, 2018
New content (p. 98)	Added the Amazon Inspector Network Reachability rules package, which allows users to run agentless assessments that analyze network configuration for security vulnerabilities. For more information, see Network Reachability.	November 9, 2018
Added OS Support (p. 98)	Added Amazon Inspector support for RHEL 7.6. For more information, see Amazon Inspector Supported Operating	October 30, 2018

	Systems and Regions and Rules Packages Availability Across Supported Operating Systems.	
Added OS support (p. 98)	Added support for various operating systems to the CIS Benchmark rules package. For more information, see Center for Internet Security (CIS) Benchmarks and Rules Packages Availability Across Supported Operating Systems.	August 13, 2018
Added Region support (p. 98)	Added Region support for `AWS GovCloud (US)`.	June 13, 2018

The following table describes the documentation release history of Amazon Inspector before June 2018.

Change	Description	Date
New content	Added the ability to target all Amazon EC2 instances in an account. For more information, see Amazon Inspector assessment targets (p. 48).	May 24, 2018
Added OS support	Added Amazon Inspector support for Amazon Linux 2018.03 and Ubuntu 18.04.	May 15, 2018
New content	Added ability to set up recurring Amazon Inspector assessments.	April 30, 2018
New content	Added ability to install an Amazon Inspector agent through the console.	April 30, 2018
Added OS support	Added Amazon Inspector support for Amazon Linux 2.	March 13, 2018
Added OS support	Added Amazon Inspector assessment support for Windows Server 2016 Base.	February 20, 2018
Added Region support	Added Amazon Inspector support for the `US East (Ohio)` Region.	February 7, 2018
New content	Amazon Inspector assessments can now run when the kernel module is unavailable.	January 11, 2018
Added Region support	Added Amazon Inspector support for the `EU (Frankfurt)` Region.	December 19, 2017
New content	Added ability to check Amazon Inspector agent health with	December 15, 2017

Change	Description	Date
	the Amazon Inspector API and console.	
New content	Added the following features: • Service-linked role usage • Amazon Inspector agent AMI available in the AWS Marketplace • Amazon Inspector AWS CloudFormation templates	December 5, 2017
Added OS support	Added Amazon Inspector assessment support for CentOS 7.4.	November 9, 2017
Added OS support	Added Amazon Inspector assessment support for Amazon Linux 2017.09.	October 11, 2017
Added OS support	Added Amazon Inspector assessment support for RHEL 7.4.	February 20, 2018
Added HIPAA eligibility	Amazon Inspector is now HIPAA eligible.	July 31, 2017
New content	Added ability to automatically trigger Amazon Inspector security assessment with Amazon CloudWatch Events.	July 27, 2017
Added Region support	Added Amazon Inspector support for the `US West (N. California)` Region.	June 6, 2018
Added OS support	Added Amazon Inspector assessment support for RHEL 6.2-6.9, RHEL 7.2-7.3, CentOS 6.9, and CentOS 7.2-7.3.	May 23, 2017
Added OS support	Added Amazon Inspector assessment support for Amazon Linux 2017.03.	April 25, 2017
New content and added OS support	Added: • Amazon Inspector support for Ubuntu 16.04. • Availability of Lambda blueprint for automating Amazon Inspector operations.	January 5, 2017
New OS support	Added Amazon Inspector support for Microsoft Windows.	August 26, 2016

Change	Description	Date
Added Region support	Added Amazon Inspector support for the `Asia Pacific (Seoul)` Region.	August 26, 2016
Added Region support	Added Amazon Inspector support for the `Asia Pacific (Mumbai)` Region.	April 25, 2016
Added Region support	Added Amazon Inspector support for the `Asia Pacific (Sydney)` Region.	April 25, 2016
Service launch	Amazon Inspector serviced launched.	Oct 7, 2015

AWS glossary

For the latest AWS terminology, see the AWS glossary in the *AWS General Reference*.