Table of Contents

What is AWS Security Hub? ........................................................................................................... 1
  Benefits of Security Hub ........................................................................................................... 1
  Getting started with Security Hub .......................................................................................... 1
  Security Hub free trial, usage, and pricing ............................................................................. 2
    Viewing usage details and estimated cost ........................................................................... 2
    Viewing pricing details ......................................................................................................... 3
Terminology and concepts ........................................................................................................... 4
  Quotas ................................................................................................................................... 7
Supported Regions ...................................................................................................................... 8
Setting up Security Hub ............................................................................................................ 9
  Attaching the required IAM policy to the IAM identity ......................................................... 9
  Enabling Security Hub .......................................................................................................... 9
    Enabling Security Hub from the console ............................................................................ 9
    Enabling Security Hub using the API ............................................................................... 10
  Service-linked role assigned to Security Hub ..................................................................... 10
  Enabling AWS Config to support security checks ............................................................... 10
Security .................................................................................................................................... 12
  Data protection .................................................................................................................... 12
  Identity and access management .......................................................................................... 13
    Audience .......................................................................................................................... 13
    Authenticating with identities ......................................................................................... 14
    AWS Account Root User ................................................................................................. 14
    IAM users and groups .................................................................................................... 14
    IAM roles ....................................................................................................................... 14
    Managing access using policies ....................................................................................... 15
    How AWS Security Hub works with IAM .................................................................... 17
Compliance validation .............................................................................................................. 22
Infrastructure security .............................................................................................................. 22
Managing access to Security Hub ............................................................................................ 23
  Using IAM policies to delegate Security Hub access to IAM identities ................................. 23
    AWS managed (predefined) policies for Security Hub .................................................... 23
    Resources defined by Security Hub ................................................................................ 23
    Condition keys defined by Security Hub ........................................................................ 24
    Using service-linked roles ............................................................................................... 24
      Service-linked role permissions for Security Hub ........................................................ 25
      Creating a service-linked role for Security Hub ............................................................. 26
      Editing a service-linked role for Security Hub ............................................................... 26
      Deleting a service-linked role for Security Hub ............................................................ 26
Master and member accounts .................................................................................................. 27
  Designating master and member accounts on the Security Hub console .......................... 27
  Designating master and member accounts through Security Hub API operations ............. 29
  Accounts and data retention in Security Hub .................................................................... 29
Insights ..................................................................................................................................... 31
  Viewing insight results and findings ..................................................................................... 31
    Viewing and taking action on insight results (Console) ................................................ 31
    Viewing insight results (API) ......................................................................................... 32
    Viewing findings for an insight result (Console) ................................................................ 32
Managed insights .................................................................................................................... 32
  Creating a new custom insight (Console) .......................................................................... 38
  Creating a new custom insight (API) .................................................................................. 38
  Modifying a custom insight (Console) .............................................................................. 38
  Modifying a custom insight (API) ...................................................................................... 39
  Creating a new custom insight from a managed insight (Console) .................................... 39
  Viewing findings for a custom insight (Console) ............................................................... 39
  Viewing findings for a custom insight (API) ..................................................................... 39
What is AWS Security Hub?

AWS Security Hub provides you with a comprehensive view of your security state in AWS and helps you 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 analyze your security trends and identify the highest priority security issues.

Topics
- Benefits of AWS Security Hub (p. 1)
- Getting started with Security Hub (p. 1)
- AWS Security Hub free trial, usage, and pricing (p. 2)

Benefits of AWS Security Hub

Reduced effort to collect and prioritize findings

Security Hub reduces the effort to collect and prioritize security findings across accounts from integrated AWS services and AWS partner products.

Security Hub processes finding data using a standard finding format, which eliminates the need to manage findings data from multiple formats.

Security Hub then correlates findings across providers to prioritize the most important ones.

Automatic security checks against best practices and standards

Security Hub automatically runs continuous, account-level configuration and security checks based on AWS best practices and industry standards.

Security Hub provides the result of these checks as a readiness score, and identifies specific accounts and resources that require attention.

Consolidated view of findings across accounts and providers

Security Hub consolidates your security findings across accounts and provider products and displays results on the Security Hub console.

This allows you to view your overall current security status to spot trends, identify potential issues, and take the necessary remediation steps.

Ability to automate remediation of findings

Security Hub supports integration with Amazon CloudWatch Events. To automate remediation of specific findings, you can define custom actions to take when a finding is received.

For example, you can configure custom actions to send findings to a ticketing system or to an automated remediation system.

Getting started with Security Hub

You can use Security Hub in the following ways:
Security Hub console

Sign in to the AWS Management Console and open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.

Security Hub API

To access Security Hub programmatically, use the Security Hub API, which allows you to issue HTTPS requests directly to the service. For more information, see the AWS Security Hub API Reference.

When you enable Security Hub, it begins to consume, aggregate, organize, and prioritize findings from AWS services that you have enabled, such as Amazon GuardDuty, Amazon Inspector, and Amazon Macie. You can also enable integrations with AWS partner security products. Those partner products can then also send findings to Security Hub. See Product integrations (p. 123).

Security Hub also generates its own findings by running continuous, automated security checks based on AWS best practices and supported industry standards. See Security standards (p. 134).

Security Hub then correlates and consolidates findings across providers to help you to prioritize the most significant findings. For information on viewing and managing findings, see the section called “Viewing and taking action on findings” (p. 43).

You can also create insights in Security Hub. An insight is a collection of findings that are grouped together when you apply a Group by filter. Insights help you identify common security issues that may require remediation action. Security Hub includes several managed insights, or you can create your own custom insights. See Insights (p. 31).

Important

Security Hub only detects and consolidates findings that are generated after you enable Security Hub. It does not retroactively detect and consolidate security findings that were generated before you enabled Security Hub.

Security Hub only receives and processes findings from the Region where you enabled Security Hub in your account.

For full compliance with CIS AWS Foundations Benchmark security checks, you must enable Security Hub in all AWS Regions.

AWS Security Hub free trial, usage, and pricing

When you enable Security Hub for the first time, your AWS account is automatically enrolled in a 30-day Security Hub free trial.

When you use Security Hub during the free trial, you are charged for usage of other services that Security Hub interacts with, such as AWS Config items. You are not charged for AWS Config rules that are enabled by Security Hub security standards.

You are not charged for using Security Hub until your free trial ends.

Viewing usage details and estimated cost

Security Hub provides usage information, including an estimated 30-day cost for using Security Hub. The usage details include the time remaining for the free trial. During the free trial, the usage information can help you to understand what the Security Hub cost will be after the free trial ends.

To display the usage information

2. In the navigation pane, choose Settings.
3. On the **Settings** page, choose **Usage**.

The estimated monthly cost is based on your Security Hub usage for findings and security checks projected over a 30-day period.

If you are using Security Hub from a master account, the estimated monthly cost includes the costs associated with all of the member accounts.

If you are using Security Hub from a member account, the estimated monthly cost is only for the member account.

The estimated monthly cost is for only the current Region. It is not for all Regions in which Security Hub is enabled.

**Viewing pricing details**

For more information about how Security Hub charges for ingested findings and security checks, see Security Hub pricing.
Terminology and concepts

This topic describes the key concepts in AWS Security Hub to help you get started.

**Account**

A standard Amazon Web Services (AWS) account that contains your AWS resources. You can sign in to AWS with your account and enable Security Hub.

You can also invite other accounts to enable Security Hub and become associated with your account in Security Hub. If your invitations are accepted, your account is designated as the Security Hub master account, and the added accounts are member accounts. As the master account, you can view findings in your member accounts.

An account cannot be both a Security Hub master account and a member account at the same time. An account can accept only one membership invitation. Accepting a membership invitation is optional.

For more information, see Master and member accounts in AWS Security Hub (p. 27).

**Archived finding**

A finding that has a RecordState set to ARCHIVED.

Finding providers can use the BatchImportFindings operation of the Security Hub API to archive findings that they created. Security Hub automatically archives findings for controls if the associated resource is deleted, based on one of the following criteria.

- The finding was not updated in three days.
- The associated AWS Config evaluation returned NOT_APPLICABLE.

By default, archived findings are excluded from findings lists in the Security Hub console. You can update the filter to include archived findings.

The GetFindings operation of the Security Hub API returns both active and archived findings. You can include a filter for the record state.

```json
"RecordState": [ 
  { 
    "Comparison": "EQUALS",
    "Value": "ARCHIVED"
  } 
],
```

**AWS Security Finding Format**

A standardized format for the contents of findings that Security Hub aggregates or generates. The AWS Security Finding Format enables you to use Security Hub to view and analyze findings that are generated by AWS security services, third-party solutions, or Security Hub itself from running security checks. For more information, see AWS Security Finding Format (ASFF) (p. 46).

**Control**

A safeguard or countermeasure prescribed for an information system or an organization designed to protect the confidentiality, integrity, and availability of its information and to meet a set of defined security requirements. A security standard consists of controls.
Custom action

A Security Hub mechanism for sending selected findings to CloudWatch Events. A custom action is created in Security Hub. It is then linked to a CloudWatch Events rule. The rule defines a specific action to take when a finding is received that is associated with the custom action ID. Custom actions can be used, for example, to send a specific finding, or a small set of findings, to a response or remediation workflow. For more information, see the section called "Creating a custom action and associating it with a CloudWatch Events rule" (p. 249).

Finding

The observable record of a security check or security-related detection.

For more information about findings in Security Hub, see Findings (p. 41).

Note

Findings are deleted 90 days after the most recent update or 90 days after the creation date if no update occurs. To store findings for longer than 90 days, you can configure a rule in CloudWatch Events that routes findings to your Amazon S3 bucket.

Insight

A collection of related findings defined by an aggregation statement and optional filters. An insight identifies a security area that requires attention and intervention. Security Hub offers several managed (default) insights that you can't modify. You can also create custom Security Hub insights to track security issues that are unique to your AWS environment and usage. For more information, see Insights (p. 31).

Related requirements

A set of industry or regulatory requirements that are mapped to a control.

Rule

A set of automated criteria that is used to assess whether a control is being adhered to. When a rule is evaluated, it can pass or fail. If the evaluation cannot determine whether rule passes or fails, then the rule is in a warning state. If the rule cannot be evaluated, then it is in a not available state.

Security check

A specific point-in-time evaluation of a rule against a single resource resulting in a passed, failed, warning, or not available state. Running a security check produces a finding.

Security standard

A published statement on a topic specifying the characteristics, usually measurable and in the form of controls, that must be satisfied or achieved for compliance. Security standards can be based on regulatory frameworks, best practices, or internal company policies. To learn more about security standards in Security Hub, see Security standards (p. 134).

Workflow status

The status of an investigation into a finding. Tracked using the Workflow.Status attribute.

The workflow status is initially NEW. If you notified the resource owner to take action on the finding, you can set the workflow status to NOTIFIED. If the finding is not an issue, and does not require any action, set the workflow status to SUPPRESSED. After you review and remediate a finding, set the workflow status to RESOLVED.

By default, most finding lists only include findings with a workflow status of NEW or NOTIFIED. Finding lists for controls also include RESOLVED findings.

For the GetFindings operation, you can include a filter for the workflow status.

"WorkflowStatus": [
The Security Hub console provides an option to set the workflow status for findings. Customers (or SIEM, ticketing, incident management, or SOAR tools working on behalf of a customer to update findings from finding providers) can also use BatchUpdateFindings to update the workflow status.
Quotas

The following AWS Security Hub quotas are per AWS account per Region.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Quota</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Security Hub member accounts</td>
<td>1000</td>
<td>The maximum number of Security Hub member accounts that can be added per account (Security Hub master account) per Region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This is a hard quota. You cannot request an increase to the allowed number of Security Hub member accounts.</td>
</tr>
<tr>
<td>Number of Security Hub outstanding invitations</td>
<td>1000</td>
<td>The maximum number of outstanding Security Hub member account invitations that can be sent per account (Security Hub master account) per Region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This is a hard quota. You cannot request an increase to the allowed number of Security Hub outstanding invitations.</td>
</tr>
<tr>
<td>Number of Security Hub custom insights</td>
<td>100</td>
<td>The maximum number of user-defined custom Security Hub insights that can be created per account per Region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This is a hard quota. You cannot request an increase to the allowed number of Security Hub custom insights.</td>
</tr>
<tr>
<td>Number of insight results</td>
<td>100</td>
<td>The maximum number of aggregated results returned for the GetInsightsResults API operation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This is a hard quota. You cannot request an increase to the number of insight results.</td>
</tr>
</tbody>
</table>
Supported Regions

To view the Regions that AWS Security Hub is available in, see Security Hub Service Endpoints.
Setting up AWS Security Hub

You must have an AWS account to enable AWS Security Hub. If you don't have an account, use the following procedure to create one.

To sign up for AWS
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.

Attaching the required IAM policy to the IAM identity

The IAM identity (user, role, or group) that you use to enable Security Hub must have the required permissions.

To grant the permissions required to enable Security Hub, attach the following policy to an IAM user, group, or role.

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": "securityhub:*",
         "Resource": "*"
      },
      {
         "Effect": "Allow",
         "Action": "iam:CreateServiceLinkedRole",
         "Resource": "*",
         "Condition": {
            "StringLike": {
               "iam:AWSServiceName": "securityhub.amazonaws.com"
            }
         }
      }
   ]
}
```

Enabling Security Hub

After you attach the required policy to the IAM identity, you use that identity to enable Security Hub.

You can enable Security Hub from the console or the API.

Enabling Security Hub from the console

When you enable Security Hub from the console, you are also given the option to enable the supported security standards.
To enable Security Hub

1. Use the credentials of the IAM identity to sign in to the Security Hub console.
2. When you open the Security Hub console for the first time, choose Get Started.
   To enable a standard, select its check box.
   To disable a standard, clear its check box.
   You can enable or disable a standard or its individual controls at any time. For information about the security standards and how to manage them, see Security standards (p. 134).

Enabling Security Hub using the API

To enable Security Hub from the API, use the EnableSecurityHub operation.

- When you enable Security Hub from the API, it automatically enables these security standards.
  - CIS AWS Foundations Standard
  - AWS Foundational Security Best Practices Standard
  If you do not want to enable these standards, then set EnableDefaultStandards to false.

After you enable Security Hub, to enable standards, use the BatchEnableStandards operation.
To disable standards, use the BatchDisableStandards operation.

Service-linked role assigned to Security Hub

When you enable Security Hub, it is assigned a service-linked role named AWSServiceRoleForSecurityHub. This service-linked role includes the permissions and trust policy that Security Hub requires to do the following:

- Detect and aggregate findings from Amazon GuardDuty, Amazon Inspector, and Amazon Macie
- Configure the requisite AWS Config infrastructure to run security checks for the supported standards (in this release, CIS AWS Foundations)

To view the details of AWSServiceRoleForSecurityHub, on the Enable Security Hub page, choose View service role permissions. For more information, see Using service-linked roles for AWS Security Hub (p. 24).

For more information about service-linked roles, see Using service-linked roles in the IAM User Guide.

Enabling AWS Config to support security checks

When you enable Security Hub from the console, you also have the option to enable the supported security standards. When you enable Security Hub from the API, then the CIS AWS Foundations standard is enabled automatically.
Many of the controls for the security standards rely on AWS Config service-level rules.

If you have any of the security standards enabled, you must enable AWS Config in the account where you enabled Security Hub. For a Security Hub master account, you must enable AWS Config in each of this account's Security Hub member accounts.

Security Hub does not manage AWS Config for you. If you already have AWS Config enabled, you can continue configuring its settings through the AWS Config console or APIs.

If you do not have AWS Config enabled, you can enable it manually or you can use the AWS CloudFormation "Enable AWS Config" template in AWS CloudFormation StackSets Sample Templates. If you use the Security Hub multi-account, multi-region enablement script, it also enables AWS Config for you.

**Important**

When you turn on the AWS Config recorder, choose to record all resources supported in a given Region, including global resources.

For more information, see Getting started with AWS Config in the *AWS Config Developer Guide*. 
Security in AWS Security Hub

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is 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 AWS Security Hub, 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 Security Hub. The following topics show you how to configure Security Hub to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Security Hub resources.

**Topics**
- Data protection in AWS Security Hub (p. 12)
- Identity and access management for AWS Security Hub (p. 13)
- Compliance validation for AWS Security Hub (p. 22)
- Infrastructure security in AWS Security Hub (p. 22)

**Data protection in AWS Security Hub**

Security Hub is a multi-tenant service offering. To ensure data protection, Security Hub encrypts data at rest and data in transit between component services.

AWS Security Hub conforms to the AWS shared responsibility model, which includes regulations and guidelines for data protection. AWS is responsible for protecting the global infrastructure that runs all the AWS services. AWS maintains control over data hosted on this infrastructure, including the security configuration controls for handling customer content and personal data. AWS customers and APN partners, acting either as data controllers or data processors, are responsible for any personal data that they put in the AWS Cloud.

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), so that 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.
- 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.

We strongly recommend that you never put sensitive identifying information, such as your customers' account numbers, into free-form fields such as a **Name** field. This includes when you work with Security Hub or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into Security Hub or other services might get picked up for inclusion in diagnostic logs. When you provide a URL to an external server, don't include credentials information in the URL to validate your request to that server.

For more information about data protection, see the [AWS Shared Responsibility Model and GDPR blog post on the AWS Security Blog](https://aws.amazon.com/security/shared-responsibility-model/).

**Identity and access management for AWS Security Hub**

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 Security Hub resources. IAM is an AWS service that you can use with no additional charge.

**Topics**
- Audience (p. 13)
- Authenticating with identities (p. 14)
- AWS Account Root User (p. 14)
- IAM users and groups (p. 14)
- IAM roles (p. 14)
- Managing access using policies (p. 15)
- How AWS Security Hub works with IAM (p. 17)

**Audience**

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

**Service user** – If you use the Security Hub service to do your job, then your administrator provides you with the credentials and permissions that you need. As you use more Security Hub 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 Security Hub, see Troubleshooting AWS Security Hub identity and access (p. 20).

**Service administrator** – If you're in charge of Security Hub resources at your company, you probably have full access to Security Hub. It's your job to determine which Security Hub 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 Security Hub, see How AWS Security Hub works with IAM (p. 17).

**IAM administrator** – If you're an IAM administrator, you might want to learn details about how you can write policies to manage access to Security Hub. To view example Security Hub identity-based policies that you can use in IAM, see AWS Security Hub identity-based policy examples (p. 19).
**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 The IAM Console and Sign-in Page 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 or your IAM user name. You can access AWS programmatically using your root user or IAM user 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.

- **AWS service access** – A service role is an IAM role that a service assumes to perform actions in your account on your behalf. When you set up some AWS service environments, you must define a role for the service to assume. This service role must include all the permissions that are required for the service to access the AWS resources that it needs. Service roles vary from service to service, but many allow you to choose your permissions as long as you meet the documented requirements for that service. Service roles provide access only within your account and cannot be used to grant access to services in other accounts. You can create, modify, and delete a service role from within IAM. For example, you can create a role that allows Amazon Redshift to access an Amazon S3 bucket on your behalf and then load data from that bucket into an Amazon Redshift cluster. For more information, see Creating a Role to Delegate Permissions to an AWS Service in the IAM User Guide.

- **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, 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. AWS evaluates these policies when an entity (root user, IAM user, or IAM role) makes a request. 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.

An IAM administrator can use policies to specify who has access to AWS resources, and what actions they can perform on those resources. 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, role, or group. These policies control what actions that identity 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 such as an Amazon S3 bucket. Service administrators can use these policies to define what actions a specified principal (account member, user, or role) can perform on that resource and under what conditions. Resource-based policies are inline policies. There are no managed resource-based policies.

### 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 AWS Security Hub works with IAM

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

Topics
- Security Hub identity-based policies (p. 17)
- Security Hub resource-based policies (Not supported) (p. 18)
- Authorization based on Security Hub tags (p. 18)
- Security Hub IAM roles (p. 18)
- Service-linked roles (p. 18)
- Service roles (p. 19)
- AWS Security Hub identity-based policy examples (p. 19)

Security Hub 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. Security Hub 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

The Action element of an IAM identity-based policy describes the specific action or actions that will be allowed or denied by the policy. Policy actions usually have the same name as the associated AWS API operation. The action is used in a policy to grant permissions to perform the associated operation.

Policy actions in Security Hub use the following prefix before the action: securityhub:. For example, to grant a user permission to enable Security Hub using the EnableSecurityHub API operation, you include the securityhub:EnableSecurityHub action in the policy assigned to that user. Policy statements must include either an Action or NotAction element. Security Hub 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": [
    "securityhub:action1",
    "securityhub:action2"
]
```

You can specify multiple actions using wildcards (*). For example, to specify all actions that begin with the word Get, include the following line in your policy:

```
"Action": "securityhub:Get***"
```

To see a list of Security Hub actions, see Actions Defined by AWS Security Hub in the IAM User Guide.

Resources

The Resource element specifies the object or objects to which the action applies. Statements must include either a Resource or a NotResource element. You specify a resource using an ARN or using the wildcard (*) to indicate that the statement applies to all resources.
For more information about the format of ARNs, see Amazon Resource Names (ARNs) and AWS Service Namespaces.

To see a list of Security Hub resource types and their ARNs, see Resources Defined by AWS Security Hub in the IAM User Guide. To learn with which actions you can specify the ARN of each resource, see Actions Defined by AWS Security Hub.

**Condition keys**

The Condition element (or Condition block) lets you specify conditions in which a statement is in effect. The Condition element is optional. You can build conditional expressions that use condition operators, such as equals or less than, to match the condition in the policy with values in the request.

If you specify multiple Condition elements in a statement, or multiple keys in a single Condition element, AWS evaluates them using a logical AND operation. If you specify multiple values for a single condition key, AWS evaluates the condition using a logical OR operation. All of the conditions must be met before the statement's permissions are granted.

You can also use placeholder variables when you specify conditions. For example, you can grant an IAM user permission to access a resource only if it is tagged with their IAM user name. For more information, see IAM Policy Elements: Variables and Tags in the IAM User Guide.

Security Hub defines its own set of condition keys and also supports using some global condition keys. To see all AWS global condition keys, see AWS Global Condition Context Keys in the IAM User Guide.

Security Hub actions support the securityhub:TargetAccount condition key.

To see a list of Security Hub condition keys, see Condition Keys for AWS Security Hub in the IAM User Guide. To learn with which actions and resources you can use a condition key, see Actions Defined by AWS Security Hub.

**Security Hub resource-based policies (Not supported)**

Security Hub does not support resource-based policies.

**Authorization based on Security Hub tags**

You can add tags to Security Hub resources or pass tags in a request to Security Hub. To control access based on tags, you provide tag information in the condition element of a policy using the securityhub:ResourceTag/key-name, aws:RequestTag/key-name, or aws:TagKeys condition keys.

**Security Hub IAM roles**

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

**Using temporary credentials with Security Hub**

You can use temporary credentials to sign in with federation, 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.

Security Hub 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.
Security Hub supports service-linked roles.

**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.

Security Hub supports service roles.

**AWS Security Hub identity-based policy examples**

By default, IAM users and roles don't have permission to create or modify Security Hub 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 (p. 19)
- Using the Security Hub console (p. 19)
- Troubleshooting AWS Security Hub identity and access (p. 20)

**Policy best practices**

Identity-based policies are very powerful. They determine whether someone can create, access, or delete Security Hub 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 Security Hub 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 Security Hub console**

To access the AWS Security Hub console, you must have a minimum set of permissions. These permissions must allow you to list and view details about the Security Hub 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 Security Hub console, also attach the following AWS managed policy to the entities. For more information, see Adding permissions to a user in the IAM User Guide:

```json
{
   "Version": "2012-10-17",
   "Statement": [
   {
      "Effect": "Allow",
      "Action": "securityhub:*",
      "Resource": "*"
   },
   {
      "Effect": "Allow",
      "Action": "iam:CreateServiceLinkedRole",
      "Resource": "*",
      "Condition": {
         "StringLike": {
            "iam:AWSServiceName": "securityhub.amazonaws.com"
         }
      }
   }
   ]
}
```

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.

**Troubleshooting AWS Security Hub identity and access**

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

**Topics**

- I am not authorized to perform an action in Security Hub (p. 20)
- I am not authorized to perform iam:PassRole (p. 21)
- I want to view my access keys (p. 21)
- I'm an administrator and want to allow others to access Security Hub (p. 21)
- I want to allow people outside My AWS account to access my Security Hub resources (p. 21)

**I am not authorized to perform an action in Security Hub**

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 view details about a widget but does not have securityhub:GetWidget permissions.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform: securityhub:GetWidget on resource: my-example-widget
```

In this case, Mateo asks his administrator to update his policies to allow him to access the `my-example-widget` resource using the securityhub:GetWidget 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 Security Hub.

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 Security Hub. 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, `wJalrXUttnFEMI/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 Security Hub

To allow others to access Security Hub, 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 Security Hub.

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 My AWS account to access my Security Hub 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 Security Hub supports these features, see How AWS Security Hub works with IAM (p. 17).
- 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*.
Compliance validation

Third-party auditors assess the security and compliance of AWS Security Hub as part of multiple AWS compliance programs. Security Hub is SOC, ISO, PCI, and HIPAA certified.

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 Security Hub 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.
- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- AWS Config – This AWS 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.

Infrastructure security in AWS Security Hub

As a managed service, AWS Security Hub 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 Security Hub 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.
Managing access to Security Hub

You use AWS Identity and Access Management to manage access to Security Hub and Security Hub resources.

Security Hub has AWS managed policies to provide full and read-only access. You can limit access to specific resources.

Security Hub also uses service-linked roles to get access to resources in your AWS account.

Topics

- Using IAM policies to delegate Security Hub access to IAM identities (p. 23)
- Using service-linked roles for AWS Security Hub (p. 24)

Using IAM policies to delegate Security Hub access to IAM identities

By default, access to the Security Hub resources is restricted to the owner of the account that the resources were created in.

If you're the owner, you can choose to grant full or limited access to Security Hub to the various IAM identities in your account. For more information about creating IAM access policies, see Controlling access using policies.

AWS managed (predefined) policies for Security Hub

AWS addresses many common use cases by providing standalone IAM policies that AWS creates and administers. These managed policies grant necessary permissions for common use cases so that you don't have to investigate which permissions are needed. For more information, see AWS managed policies in the IAM User Guide.

The following AWS managed policies, which you can attach to users in your account, are specific to Security Hub:

- AWSSecurityHubFullAccess – Provides access to all Security Hub functionality
- AWSSecurityHubReadOnlyAccess – Provides read-only access to Security Hub

Resources defined by Security Hub

The following resource types are defined by this service and can be used in the Resource element of IAM permission policy statements.

<table>
<thead>
<tr>
<th>Resource Types</th>
<th>ARN</th>
</tr>
</thead>
<tbody>
<tr>
<td>action-target</td>
<td>arn:${Partition}:securityhub:${Region}:${Account}:action/custom/${Id}</td>
</tr>
<tr>
<td>hub</td>
<td>arn:${Partition}:securityhub:${Region}:${Account}:hub/default</td>
</tr>
</tbody>
</table>
### Condition keys defined by Security Hub

Security Hub defines the following condition keys that you can use in the `Condition` element of an IAM policy.

You can use these keys to further refine the conditions under which the policy statement applies.

<table>
<thead>
<tr>
<th>Condition Keys</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>securityhub:TargetAccount</td>
<td>The ID of the AWS account associated with a finding. In the AWS Security Finding Format, this field is called AwsAccountId.</td>
<td>String</td>
</tr>
</tbody>
</table>

### Using service-linked roles for AWS Security Hub

AWS Security Hub 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 Security Hub. Service-linked roles are predefined by Security Hub and include all the permissions that Security Hub requires to call other AWS services on your behalf.

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

Security Hub supports using service-linked roles in all of the Regions where Security Hub is available. For more information, see Supported Regions (p. 8).

You can delete the Security Hub service-linked role only after first disabling Security Hub in all Regions where it's enabled. This protects your Security Hub resources because you can't inadvertently remove permissions to access them.
For information about other services that support service-linked roles, see AWS services that work with IAM in the IAM User Guide and locate 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 Security Hub

Security Hub uses the service-linked role named AWSServiceRoleForSecurityHub. It's a service-linked role required for AWS Security Hub to access your resources.

The AWSServiceRoleForSecurityHub service-linked role trusts the following services to assume the role:

- securityhub.amazonaws.com

The role permissions policy allows Security Hub to complete the following actions on the specified resources:

- Action: cloudtrail:DescribeTrails
- Action: cloudtrail:GetTrailStatus
- Action: cloudtrail:GetEventSelectors
- Action: cloudwatch:DescribeAlarms
- Action: logs:DescribeMetricFilters
- Action: sns:ListSubscriptionsByTopic
- Action: config:DescribeConfigurationRecorders
- Action: config:DescribeConfigurationRecorderStatus
- Action: config:DescribeConfigRules
- Action: config:BatchGetResourceConfig
- Resources: *

And:

- Action: config:PutConfigRule
- Action: config:DeleteConfigRule
- Action: config:GetComplianceDetailsByConfigRule
- Action: config:DescribeConfigRuleEvaluationStatus
- Resources: arn:aws:config:*:*:config-rule/aws-service-rule/*securityhub*

You must configure permissions to allow an IAM entity (such as a user, group, or role) to create, edit, or delete a service-linked role. For the AWSServiceRoleForSecurityHub service-linked role to be successfully created, the IAM identity that you use Security Hub with must have the required permissions. To grant the required permissions, attach the following policy to this IAM user, group, or role.

```json
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": "securityhub:*",
         "Resource": "*"
      },
      {
         "Effect": "Allow",
         "Action": "iam:CreateServiceLinkedRole",
         "Resource": "*"
      }
   ]
}
```
Creating a service-linked role for Security Hub

The AWSServiceRoleForSecurityHub service-linked role is automatically created when you enable Security Hub for the first time or enable Security Hub in a supported Region where you previously didn't have it enabled. You can also create the AWSServiceRoleForSecurityHub service-linked role manually using the IAM console, the IAM CLI, or the IAM API.

Important
The service-linked role that is created for the Security Hub master account doesn't apply to the Security Hub member accounts.

For more information about creating the role manually, see Creating a service-linked role in the IAM User Guide.

Editing a service-linked role for Security Hub

Security Hub doesn't allow you to edit the AWSServiceRoleForSecurityHub 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 by using IAM. For more information, see Editing a service-linked role in the IAM User Guide.

Deleting a service-linked role for Security Hub

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 isn't actively monitored or maintained.

Important
To delete the AWSServiceRoleForSecurityHub service-linked role, you must first disable Security Hub in all Regions where it's enabled. If Security Hub isn't disabled when you try to delete the service-linked role, the deletion fails. For more information, see Disabling AWS Security Hub (p. 253).

When you disable Security Hub, the AWSServiceRoleForSecurityHub service-linked role is not automatically deleted. If you enable Security Hub again, it starts using the existing AWSServiceRoleForSecurityHub service-linked role.

To manually delete the service-linked role using IAM

Use the IAM console, the IAM CLI, or the IAM API to delete the AWSServiceRoleForSecurityHub service-linked role. For more information, see Deleting a service-linked role in the IAM User Guide.
Master and member accounts in AWS Security Hub

You can invite other AWS accounts to enable AWS Security Hub and become associated with your AWS account. If the owner of the account that you invite enables Security Hub and then accepts the invitation, your account is designated as the master Security Hub account, and the invited accounts become associated as member accounts. When the invited account accepts the invitation, permission is granted to the master account to view the findings from the member account. The master account can also perform actions on findings in a member account.

Security Hub supports up to 1000 member account per master account per Region. The master-member account association is created in only the one Region that the invitation was sent from. You must enable Security Hub in each Region that you want to use in, and then invite each account to associate as a member account in each Region.

Security Hub aggregates findings from Amazon GuardDuty, Amazon Inspector, and Amazon Macie. However, the master-member relationships that you set up for your accounts in which GuardDuty, Amazon Inspector, or Amazon Macie are enabled don't automatically apply to Security Hub.

For example, suppose that as a user from a GuardDuty master account A you can see the findings of accounts B and C (GuardDuty member accounts) on the GuardDuty console. If you then enable Security Hub in account A, as a user from account A, you do not automatically see the findings generated by GuardDuty for accounts B and C in Security Hub. You need to create a master-member relationship between these accounts in Security Hub as well. You must first enable Security Hub in all three accounts (A, B, and C). Then make account A the Security Hub master account and then invite accounts B and C to become member accounts in Security Hub.

An account can't be a Security Hub master account and member account at the same time. An account can accept only one Security Hub membership invitation. Accepting a membership invitation is optional.

Designating master and member accounts on the Security Hub console

In Security Hub, your account becomes the master account when the account that you invite accepts your invitation. When you accept an invitation from another account, your account becomes a member account. If your account is the master account, you can't accept an invitation to become a member account.

Use the following procedures to add an account, invite an account, or accept an invitation from another account.

- Procedure 1: Adding an account
- Procedure 2: Inviting an account
- Procedure 3: Accepting an invitation

Procedure 1: Adding an account

2. In the left pane, choose **Settings**.
3. On the **Settings page** choose **Accounts**, choose **Add accounts**, and then do one of the following:
4. Under **Enter accounts**, enter the **Account ID** and the **Email address** of the account to add, then choose **Add**.

To add more accounts, enter the account ID and email address for an account and then choose **Add** for each account.

You can add multiple accounts at the same time by using a comma-separated values (CSV) file. Add the account ID and email for each account to add, and then choose **Upload list (.csv)** to bulk-add accounts.

**Important**
In your .csv list, accounts must appear one per line. The first line of the .csv file must contain the following header, as shown in the following example: **Account ID,Email**. Each subsequent line must contain a valid account ID and email address for the account to add. Separate the account ID and email address with a comma.

```
Account ID,Email
111111111111,user@example.com
```

5. After you finish adding accounts, choose **Add**. Then in the **Accounts to be added** section, choose **Next**.

**Procedure 2: Inviting an account**

2. In the navigation pane, under **Settings**, choose **Accounts**.
3. For the account to invite, choose **Invite** in the **Status** column.
4. In the **Invitation to Security Hub** dialog box, choose **Invite**.

The value in the **Status** column for the invited account changes to **Invited**.

**Procedure 3: Accepting an invitation**

2. Do one of the following:
   
   After Security Hub is enabled, choose **Settings**, then choose **Accounts**. Locate the invitation to accept. Use the **Accept** widget and the **Accept invitation** button to accept the membership invitation.

   **Important**
   You must enable Security Hub before you can accept a membership invitation.

   - If Security Hub is already enabled, use the **Accept** widget and the **Accept invitation** button to accept the membership invitation.

After you accept the invitation, your account becomes a Security Hub member account. The account used to send the invitation becomes the Security Hub master account. The master account user can now view Security Hub aggregated findings for your member account.
Designating master and member accounts through Security Hub API operations

You can also designate Security Hub master and member accounts with operations in the Security Hub API. Use the following Security Hub API operations in the order listed to create master and member accounts.

Use these operations to designate a master account and then send an invitation to become a member account.

1. Run `CreateMembers` using the credentials of the account that has Security Hub enabled. This is the account that you want to be the master Security Hub account.
2. Run `InviteMembers` using the master account.

Use these operations to enable Security Hub and then accept an invitation. Use the credentials for the account you invited to become the member account.

1. Run `EnableSecurityHub` for each account that you invited. Security Hub must be enabled in the account before the account owner can accept the invitation.
2. Run `AcceptInvitation` for each account you invited to accept your invitation.

Accounts and data retention in Security Hub

When you disable Security Hub for an account, either master or member, it is disabled only for that account in the AWS Region that is selected when you disable it. You must disable Security Hub separately in each Region where you enabled it.

When you disable Security Hub for a master account, the default company and product settings are removed. Integrations with Macie, GuardDuty, and Amazon Inspector are removed. Enabled security standards are disabled. Other Security Hub data and settings, including member account associations, custom actions, insights, and subscriptions to third-party products are not removed. No new findings are generated for the master account while Security Hub isn't enabled, and existing findings are deleted after 90 days. If you enable Security Hub again later, the default company and product settings, security standards that you had enabled, and integrations with AWS services are restored. This lets you use Security Hub as you did before you disabled it without having to reconfigure it.

When you disable Security Hub for a member account, no new findings are generated for the member account in the Region, but the master account can still view existing findings in the member account. Findings are deleted 90 days after the last update, or 90 days after they are created if no update is made. The relationship of master and member account is maintained. You can enable Security Hub in the member account and use it as you did before you disabled it, except that there are no findings for the period of time when Security Hub was not enabled.

When a member account is disassociated from the master account, the master account loses permission to view findings in the member account. Security Hub continues to run in both accounts. Custom settings or integrations defined for the master account are not applied to findings from the former member account. For example, a custom action in the master account used as the event pattern in a CloudWatch Events rule can't be used in the member account after the accounts are disassociated.

When your AWS account is deleted or suspended, all Security Hub–related data for that account is deleted after 90 days. The data can't be retrieved after it's deleted. To retain findings for more than 90
days, you can archive them or use a custom action with a CloudWatch Events rule to store findings in your Amazon S3 bucket.
Insights in AWS Security Hub

An AWS Security Hub insight is a collection of related findings. It identifies a security area that requires attention and intervention. For example, an insight might point out EC2 instances that are the subject of findings that detect poor security practices. An insight brings together findings from across finding providers.

Each insight is defined by a group by statement and optional filters. The group by statement indicates how to group the matching findings, and identifies the type of item that the insight applies to. For example, if an insight is grouped by resource identifier, then the insight produces a list of resource identifiers. The optional filters narrow down the matching findings for the insight. For example, you might want to only see findings from specific providers or findings associated with specific types of resources.

Security Hub offers several built-in managed (default) insights. You cannot modify or delete managed insights.

To track security issues that are unique to your AWS environment and usage, you can create custom insights.

An insight only returns results if you have enabled product integrations or security standards that produce matching findings. For example, the managed insight 29. Top resources by counts of failed CIS checks only returns results if you enable the CIS AWS Foundations security standard.

Topics
- Viewing and taking action on insight results and findings (p. 31)
- Managed insights (p. 32)
- Managing custom insights (p. 37)

Viewing and taking action on insight results and findings

For each insight, AWS Security Hub first determines the findings that match the filter criteria, then uses the grouping attribute to group the matching findings.

From the Insights page, you can view and take action on the results and findings.

Viewing and taking action on insight results (Console)

The insight results consist of a grouped list of the results for the insight. For example, if the insight is grouped by resource ID, then the insight results are the list of resource IDs. Each item in the results list indicates the number of matching findings for that item.

The results list is sorted from most to fewest matching findings.

In addition to the results list, the insight results display a set of charts summarizing the number of matching findings for the following attributes.

- Severity label - Number of findings for each severity label
- Account ID - Top 5 account IDs for the matching findings
- Resource type - Top 5 resource types for the matching findings
- Resource ID - Top 5 resource IDs for the matching findings
- Product Name - Top 5 finding providers for the matching findings
If you have configured custom actions, then you can send selected results to a custom action. The action must be associated with a CloudWatch rule for the Security Hub Insight Results event type. See *Security Hub with CloudWatch Events* (p. 247).

**To display and take action on the list of insight results**

2. In the navigation pane, choose **Insights**.
3. To display the list of insight results, choose the insight name.
4. To take action on an insight result or results:
   a. Select the check box for each result to send to the custom action.
   b. From the **Actions** menu, choose the custom action.

**Viewing insight results (API)**

To retrieve the list of insight results using the API, use the **GetInsightResults** operation.

**Viewing findings for an insight result (Console)**

From the insight results list, you can display the list of findings for each result.

**To display and take action on insight findings**

2. In the navigation pane, choose **Insights**.
3. To display the list of insight results, choose the insight name.
4. To display the list of findings for an insight result, choose the item from the results list.

The findings list shows the active findings for the selected insight result that have a workflow status of **NEW** or **NOTIFIED**.

From the findings list, you can perform the following actions.

- Change the filters and grouping for the list (p. 44)
- View details for individual findings (p. 45)
- Update the workflow status of findings (p. 45)
- Send findings to custom actions (p. 46)

**Managed insights**

AWS Security Hub provides several managed insights.

You cannot edit or delete Security Hub managed insights. You can view and take action on the insight results and findings (p. 31). You can also use a managed insight as the basis for a new custom insight (p. 39).

As with all insights, a managed insight only returns results if you have enabled product integrations or security standards that can produce matching findings.

In the current release, Security Hub offers the following managed insights:
<table>
<thead>
<tr>
<th>Insight name</th>
<th>Grouped by</th>
<th>Finding filters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. AWS resources with the most findings</td>
<td>Resource identifier</td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>2. S3 buckets with public write or read permissions</td>
<td>Resource identifier</td>
<td>Type starts with Effects/Data Exposure</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resource type is AwsS3Bucket</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>3. AMIs that are generating the most findings</td>
<td>EC2 instance image ID</td>
<td>Resource type is AwsEc2Instance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>4. EC2 instances involved in known Tactics, Techniques, and Procedures (TTPs)</td>
<td>Resource ID</td>
<td>Type starts with TTPs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resource type is AwsEc2Instance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>5. AWS users with the most suspicious activity</td>
<td>IAM access key user name</td>
<td>Resource type is AwsIamAccessKey</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>6. AWS resources instances that don't meet security standards / best practices</td>
<td>Resource ID</td>
<td>Type is Software and Configuration Checks/Industry and Regulatory Standards/AWS Security Best Practices</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>7. AWS resources associated with potential data exfiltration</td>
<td>Resource ID</td>
<td>Type starts with Effects/Data Exfiltration/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>8. AWS resources associated with unauthorized resource consumption</td>
<td>Resource ID</td>
<td>Type starts with Effects/Resource Consumption</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>9. S3 buckets that don't meet security standards / best practice</td>
<td>Resource ID</td>
<td>Resource type is AwsS3Bucket</td>
</tr>
<tr>
<td>Insight name</td>
<td>Grouped by</td>
<td>Finding filters</td>
</tr>
<tr>
<td>----------------------------------------------------------------------------</td>
<td>--------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| 10. S3 buckets with sensitive data                                         | Resource ID  | Resource type is AwsS3Bucket  
Type starts with Sensitive Data Identifications/  
Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
| 11. Credentials that may have leaked                                      | Resource ID  | Type starts with Sensitive Data Identifications/Passwords/  
Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
| 12. EC2 instances that have missing security patches for important vulnerabilities | Resource ID  | Type starts with Software and Configuration Checks/Vulnerabilities/CVE  
Resource type is AwsEc2Instance  
Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
| 13. EC2 instances with general unusual behavior                            | Resource ID  | Type starts with Unusual Behaviors  
Resource type is AwsEc2Instance  
Record state is ACTIVE  
Workflow status is NEW or NOTIFIED. |
| 14. EC2 instances that have ports accessible from the Internet             | Resource ID  | Type starts with Software and Configuration Checks/AWS Security Best Practices/Network Reachability  
Resource type is AwsEc2Instance  
Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
<table>
<thead>
<tr>
<th>Insight name</th>
<th>Grouped by</th>
<th>Finding filters</th>
</tr>
</thead>
<tbody>
<tr>
<td>15. EC2 instances that don't meet security standards / best practices</td>
<td>Resource ID</td>
<td>Type starts with one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Software and Configuration Checks/Industry and Regulatory Standards/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Software and Configuration Checks/AWS Security Best Practices</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resource type is AwsEc2Instance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>16. EC2 instances that are open to the Internet</td>
<td>Resource ID</td>
<td>Type starts with Software and Configuration Checks/AWS Security Best Practices/</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Network Reachability</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resource type is AwsEc2Instance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>17. EC2 instances associated with adversary reconnaissance</td>
<td>Resource ID</td>
<td>Type starts with TTPs/Discovery/Recon</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resource type is AwsEc2Instance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>18. AWS resources that are associated with malware</td>
<td>Resource ID</td>
<td>Type starts with one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Effects/Data Exfiltration/Trojan</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• TTPs/Initial Access/Trojan</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• TTPs/Command and Control/Backdoor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• TTPs/Command and Control/Trojan</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Software and Configuration Checks/Backdoor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Unusual Behaviors/VM/Backdoor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record state is ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>Insight name</td>
<td>Grouped by</td>
<td>Finding filters</td>
</tr>
<tr>
<td>--------------</td>
<td>------------</td>
<td>-----------------</td>
</tr>
</tbody>
</table>
| 19. AWS resources associated with cryptocurrency issues | Resource ID | Type starts with one of the following:  
- Effects/Resource Consumption/Cryptocurrency  
- TTPs/Command and Control/CryptoCurrency  
Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
| 20. AWS resources with unauthorized access attempts | Resource ID | Type starts with one of the following:  
- TTPs/Command and Control/UnauthorizedAccess  
- TTPs/Initial Access/UnauthorizedAccess  
- Effects/Data Exfiltration/UnauthorizedAccess  
- Unusual Behaviors/User/UnauthorizedAccess  
- Effects/Resource Consumption/UnauthorizedAccess  
Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
| 21. Threat Intel indicators with the most hits in the last week |  | Created within the last 7 days |
| 22. Top accounts by counts of findings | AWS account ID | Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
| 23. Top products by counts of findings | Product ARN | Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
| 24. Severity by counts of findings | Severity label | Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
| 25. Top S3 buckets by counts of findings | Resource ID | Resource type is AwsS3Bucket  
Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
| 26. Top EC2 instances by counts of findings | Resource ID | Resource type is AwsEc2Instance  
Record state is ACTIVE  
Workflow status is NEW or NOTIFIED |
Managing custom insights

In addition to the AWS Security Hub managed insights, you can create custom insights to track issues and resources that are specific to your environment.

You can create completely new custom insights, or start from an existing custom or managed insight.

Each insight is configured with the following options.

- The grouping attribute. The grouping attribute determines the items that are displayed in the insight results list. For example, if the grouping attribute is Product name, then the insight results display the number of findings associated with each finding provider.

- Optional filters. The filters narrow down the matching findings for the insight.

When querying your findings, Security Hub applies AND logic to the set of filters. In other words, a finding only matches if it matches all of the provided filters. For example, if the filters are "Product name is GuardDuty" and "Resource type is AwsS3Bucket,", then matching findings must match both of these criteria.

<table>
<thead>
<tr>
<th>Insight name</th>
<th>Grouped by</th>
<th>Finding filters</th>
</tr>
</thead>
<tbody>
<tr>
<td>27. Top AMIs by counts of findings</td>
<td>EC2 instance image ID</td>
<td>Resource type is AwsEc2Instance&lt;br&gt;Record state is ACTIVE&lt;br&gt;Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>28. Top IAM users by counts of findings</td>
<td>IAM access key user name</td>
<td>Resource type is AwsIamAccessKey&lt;br&gt;Record state is ACTIVE&lt;br&gt;Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>29. Top resources by counts of failed CIS checks</td>
<td>Resource ID</td>
<td>Generator ID starts with&lt;br&gt;arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/v/1.2.0/rule&lt;br&gt;Updated in the last day&lt;br&gt;Compliance status is FAILED&lt;br&gt;Record state is ACTIVE&lt;br&gt;Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>30. Top integrations by counts of findings</td>
<td>Product ARN</td>
<td>Record state is ACTIVE&lt;br&gt;Workflow status is NEW or NOTIFIED</td>
</tr>
<tr>
<td>31. Resources with the most failed security checks</td>
<td>Resource ID</td>
<td>Updated in the last day&lt;br&gt;Compliance status is FAILED&lt;br&gt;Record state is ACTIVE&lt;br&gt;Workflow status is NEW or NOTIFIED</td>
</tr>
</tbody>
</table>
However, Security Hub applies OR logic to filters that use the same attribute but different values. For example, if the filters are "Product name is GuardDuty" and "Product name is Amazon Inspector", then a finding matches if it was generated by either GuardDuty or Amazon Inspector.

Creating a new custom insight (Console)

From the console, you can create a completely new insight.

**To create an insight**

2. In the navigation pane, choose Insights.
3. Choose Create insight.
4. To select the grouping attribute for the insight:
   a. Choose the Add filter field.
   b. Choose Group by.
   c. Select the attribute to use to group the findings associated with this insight.
   d. Choose Apply.
5. (Optional) Choose any additional filters to use for this insight. For each filter, define the filter criteria, then choose Apply.
6. Choose Create insight.
7. Enter an Insight name, then choose Create insight.

Creating a new custom insight (API)

To create a new custom insight from the Security Hub API, use the CreateInsight operation.

Modifying a custom insight (Console)

You can modify an existing custom insight to change the grouping value and filters. After you make the changes, you can save the updates to the original insight, or save the updated version as a new insight.

**To modify an insight**

2. In the navigation pane, choose Insights.
3. Choose the custom insight to modify.
4. To change the attribute used to group findings in the insight:
   a. Remove the existing grouping. To do this, choose the circled X next to Group by.
   b. Choose Add filter.
   c. Select the attribute to use for grouping.
   d. Choose Apply.
5. To remove a filter from the insight, choose the circled X next to the filter.
6. To add a filter to the insight:
   a. Choose Add filter.
   b. Select the attribute and value to use as a filter.
c. Choose Apply.
7. When you complete the updates, choose Save insight.
8. When prompted, do one of the following:
   - To update the existing insight to reflect your changes, choose Update <Insight_Name> and then choose Save insight.
   - To create a new insight with the updates, choose Save new insight. Enter an Insight name, and then choose Save insight.

Modifying a custom insight (API)

To update the configuration of a custom insight, use the UpdateInsight operation.

Creating a new custom insight from a managed insight (Console)

You cannot save changes to or delete a managed insight. You can use a managed insight as the basis for a new custom insight.

To create a new custom insight from a managed insight
2. In the navigation pane, choose Insights.
3. Choose the managed insight to work from.
4. To change the attribute used to group findings in the insight:
   a. Remove the existing grouping. To do this, choose the circled X next to Group by.
   b. Choose Add filter.
   c. Select the attribute to use for grouping.
   d. Choose Apply.
5. To remove a filter from the insight, choose the circled X next to the filter.
6. To add a filter to the insight:
   a. Choose Add filter.
   b. Select the attribute to use as a filter.
   c. Choose Apply.
7. When your updates are complete, choose Create insight.
8. When prompted, enter an Insight name, then choose Create insight.

Deleting a custom insight (Console)

When you no longer want a custom insight, you can delete it. You cannot delete managed insights.

To delete a custom insight
2. In the navigation pane, choose Insights.
3. Locate the custom insight to delete.
4. For that insight, choose the more options icon (the three dots in the top-left corner of the card).
5. Choose **Delete**.

**Deleting a custom insight (API)**

To delete a custom insight from the Security Hub API, use the `DeleteInsight` operation.
Findings in AWS Security Hub

AWS Security Hub eliminates the complexity of addressing large volumes of findings from multiple providers. It reduces the effort required to manage and improve the security of all of your AWS accounts, resources, and workloads.

Security Hub receives findings from the following sources.

- Integrations with AWS security services that you enable. See the section called “Available AWS service integrations” (p. 125).
- Integrations with third-party products that you enable. See the section called “Available third-party partner product integrations” (p. 126).
- Custom integrations that you configure. See the section called “Using custom product integrations” (p. 131).
- Security Hub checks against enabled controls. See the section called “Results of security checks” (p. 136).

Security Hub consumes findings using a standard findings format called the AWS Security Finding Format. For more information about the finding format, see the section called “Finding format” (p. 46).

Security Hub correlates the findings across integrated products to prioritize the most important ones.

Finding providers can update findings to reflect additional instances of the finding. You can update findings to provide details about your investigation and its results.

Topics
- Types of finding updates in Security Hub (p. 41)
- Viewing and taking action on findings (p. 43)
- AWS Security Finding Format (ASFF) (p. 46)

Types of finding updates in Security Hub

In AWS Security Hub, a finding can originate from one of the following types of finding providers.

- An enabled integration with another AWS service
- An enabled integration with a third-party provider
- An enabled security standard and control in Security Hub

After a finding is created, it can be updated by the finding provider or by the customer.

- The finding provider uses the BatchImportFindings API operation to update the general information about a finding. Finding providers can only update findings that they created.
- The customer uses BatchUpdateFindings API operation to reflect the status of the investigation into a finding. BatchUpdateFindings can also be used by a ticketing, incident management, orchestration, remediation, or SIEM tool on behalf of the customer.
From the Security Hub console, they can view finding details, manage the workflow status of findings, and send findings to custom actions. See the section called “Viewing and taking action on findings” (p. 43).

Security Hub also automatically updates and deletes findings.

Topics

• Using BatchImportFindings to create and update findings (p. 42)
• Using BatchUpdateFindings to update a finding (p. 43)

Using BatchImportFindings to create and update findings

Finding providers use the BatchImportFindings API operation to create new findings and to update information about the findings they created. They cannot update findings that they did not create.

Customers, SIEMs, ticketing tools, and SOAR tools use BatchUpdateFindings to make updates related to their processing of findings from finding providers. See the section called “Using BatchUpdateFindings to update a finding” (p. 43).

AWS Security Hub can only accept finding updates for accounts that have Security Hub enabled. The finding provider also must be enabled. If Security Hub is disabled, or the finding provider integration is not enabled, then the findings are returned in the FailedFindings list, with an InvalidAccess error.

The payload for a BatchImportFindings call cannot be larger than 6MB.

Determining whether to create or update a finding

To determine whether to create or update a finding, Security Hub checks the ID field. If the value of ID does not match an existing finding, then a new finding is created.

If ID does match an existing finding, then Security Hub checks the UpdatedAt field for the update.

• If UpdatedAt on the update occurs before UpdatedAt on the existing finding, then the update is ignored.
• If UpdatedAt on the update occurs after UpdatedAt on the existing finding, then the existing finding is updated.

Restricted fields for BatchImportFindings

For an existing finding, finding providers cannot use BatchImportFindings to update the following fields and objects. These fields can only be updated using BatchUpdateFindings.

• Confidence
• Criticality
• Note
• RelatedFindings
• Severity
• Types
• UserDefinedFields
• VerificationState
Using BatchUpdateFindings to update a finding

Customers (or SIEM, ticketing, incident management, or SOAR tools working on behalf of a customer to update findings from finding providers) use BatchUpdateFindings to update information related to their processing of existing findings. BatchUpdateFindings cannot be used to create new findings. It can be used to update up to 100 findings at a time.

BatchUpdateFindings does not change the UpdatedAt field for the finding. UpdatedAt only reflects the most recent update from the finding provider.

Master accounts can use BatchUpdateFindings to update findings for their account or for their member accounts. For those accounts, they can only use BatchUpdateFindings to update the following fields and objects.

- Confidence
- Criticality
- Note
- RelatedFindings
- Severity
- Types
- UserDefinedFields
- VerificationState
- Workflow

Member accounts can only use BatchUpdateFindings to update the Note object on their findings.

Viewing and taking action on findings

In the AWS Security Hub navigation pane, Findings displays a list of findings from all of the enabled product integrations and controls.

From Security standards, you can display a list of findings generated from a selected control. See the section called “Viewing details and findings for a control” (p. 140).

From Integrations, you can display a list of findings generated by an enabled integration. See the section called “Viewing the findings from an integration” (p. 125).

From Insights, you can display a list of findings for a matching insight result. See the section called “Viewing insight results and findings” (p. 31).

You can also use the GetFindings API operation to retrieve a filtered list of findings.

All findings are automatically deleted if they were not updated in the past 90 days.

Topics
- Filtering and grouping findings (p. 44)
- Viewing finding details (p. 45)
- Setting the workflow status for a finding (p. 45)
- Sending findings to a custom action (p. 46)
Filtering and grouping findings

When you first display a list of findings, the list is always filtered based on the record state and workflow status. This is in addition to the filters for an insight, integration, or control.

The record state indicates whether the finding is active or archived. A finding can be archived by the finding provider. AWS Security Hub also automatically archives findings for controls if the associated resource is deleted. By default, a finding list only shows active findings.

The workflow status indicates the status of the investigation into the finding. The workflow status can only be updated by the Security Hub customer or a system operating on the customer's behalf. By default, a finding list only shows findings with a workflow status of NEW or NOTIFIED. The default finding list for a control also includes RESOLVED findings.

When you display a list of findings for a control, you can only filter based on text in the finding list. You cannot add or remove filters, or group the findings.

For other lists of findings, you can change the filters and group the findings.

Adding filters

To change the scope of the list, you can add filters to it.

When filtering the finding list, Security Hub applies AND logic to the set of filters. In other words, a finding only matches if it matches all of the provided filters. For example, if the filters are "Product name is GuardDuty" and "Resource type is AwsS3Bucket," then matching findings must match both of these criteria.

However, Security Hub applies OR logic to filters that use the same attribute but different values. For example, if the filters are "Product name is GuardDuty" and "Product name is Amazon Inspector", then a finding matches if it was generated by either GuardDuty or Amazon Inspector.

To add a filter to the finding list

1. Choose the Filter field.
2. In the menu, under Filters, choose the field to use for the filter.
3. Choose the filter match type.
   - For a string field, you can choose whether the field value matches (EQUALS) or starts with (PREFIX) the filter value.
   - For a numeric field, you can choose whether to provide a single number (Simple) or a range of numbers (Range).
   - For a datetime field, you can choose whether to provide a length of time from the current date time (Rolling window) or a date range (Fixed range).
4. Specify the filter value.
   - Note that for string fields, the filter value is case sensitive.
   - For example, for findings from Security Hub, Product name isSecurity Hub. If you use the EQUALS operator to see findings from Security Hub, you must enter Security Hub as the filter value. If you enter security hub, no findings are displayed.
   - Similarly, if you use the PREFIX operator, and enter Sec, Security Hub findings are displayed. If you enter sec, no Security Hub findings are displayed.
5. Choose Apply.
Grouping findings

In addition to changing the filters, you can group the findings based on a grouping field.

When you add a grouping field, the list of findings is replaced with a list of values for that field in the matching findings. For each field value, the list displays the number of findings that match the other filter criteria.

For example, if you group the findings by AWS account ID, you see a list of account identifiers, with the number of matching findings for each account.

When you choose a field value, the list of matching findings for that field value is displayed.

To group the findings in a findings list

1. Choose the Filter field.
2. In the menu, under Grouping, choose Group by.
3. In the list, choose the field to use for the grouping.
4. Choose Apply.

Changing a filter value or grouping field

For an existing filter, you can change the filter value. You can also change the grouping field.

For example, you can change the Record state filter to look for ARCHIVED findings instead of ACTIVE findings.

To edit a filter or grouping field

1. Choose the filter or grouping field.
2. For the grouping field, choose the new field, then choose Apply.
3. For a filter, choose the new value, then choose Apply.

Deleting a filter or grouping field

To delete a filter or grouping field, choose the x icon.

The list updates automatically to reflect the change. When you remove the grouping field, the list changes from the list of field values back to a list of findings.

Viewing finding details

From a finding list, to view the finding details pane, choose the finding title.

To add a field value to the finding list filter, choose + next to the field.

To display the complete JSON for the finding, choose the finding title. From Finding JSON, you can download the finding JSON to a file.

For findings that are based on AWS Config rules, to display a list of the applicable rules, choose Rules.

Setting the workflow status for a finding

For findings, the workflow status tracks the progress of your investigation into a finding.

The workflow status values are as follows.
NEW

The initial state of a finding, before you review it.

NOTIFIED

Indicates that you notified the resource owner about the security issue. You can use this status when you are not the resource owner, and you need intervention from the resource owner in order to resolve a security issue.

SUPPRESSED

The finding will not be reviewed again and will not be acted upon.

RESOLVED

The finding was reviewed and remediated and is now considered resolved.

Setting the workflow status (Console)

To set the workflow status from the finding details for a finding, from Workflow status, choose the status.

You can also set the workflow status for multiple selected findings in the findings list.

To set the workflow status for multiple findings

1. In the findings list, select the check box for each finding that you want to update.
2. From Change workflow status, choose the status.

Setting the workflow status (API)

To set the workflow status for findings from the Security Hub API, use the BatchUpdateFindings API operation.

Sending findings to a custom action

You can create AWS Security Hub custom actions to automate Security Hub with Amazon CloudWatch Events. For custom actions, the event type is Security Hub Findings - Custom Action.

For more information and detailed steps on creating custom actions, see Security Hub with CloudWatch Events (p. 247).

After you set up a custom action, you can send findings to it.

To send findings to a custom action

1. In the findings list, select the check box for each finding to send to the custom action.
   You can send up to 20 findings at a time.
2. From Actions, choose the custom action.

AWS Security Finding Format (ASFF)

AWS Security Hub consumes, aggregates, organizes, and prioritizes findings from AWS security services and from the third-party product integrations. Security Hub processes these findings using a standard
findings format called the AWS Security Finding Format (ASFF), which eliminates the need for time-consuming data conversion efforts. Then it correlates ingested findings across products to prioritize the most important ones.

Topics

- ASFF syntax (p. 47)
- ASFF attributes (p. 54)
- Types taxonomy for ASFF (p. 120)

ASFF syntax

The following is the syntax of the complete finding JSON in the ASFF.

```json
"Findings": [
  {
    "AwsAccountId": "string",
    "Compliance": {
      "Status": "string",
      "RelatedRequirements": ["string"]
    },
    "Confidence": number,
    "CreatedAt": "string",
    "Criticality": number,
    "Description": "string",
    "FirstObservedAt": "string",
    "GeneratorId": "string",
    "Id": "string",
    "LastObservedAt": "string",
    "Malware": [
      {
        "Name": "string",
        "Path": "string",
        "State": "string",
        "Type": "string"
      }
    ],
    "Network": {
      "DestinationDomain": "string",
      "DestinationIpV4": "string",
      "DestinationIpV6": "string",
      "DestinationPort": number,
      "Direction": "string",
      "Protocol": "string",
      "SourceDomain": "string",
      "SourceIpV4": "string",
      "SourceIpV6": "string",
      "SourceMac": "string",
      "SourcePort": number
    },
    "Note": {
      "Text": "string",
      "UpdatedAt": "string",
      "UpdatedBy": "string"
    },
    "Process": {
      "LaunchedAt": "string",
      "Name": "string",
      "ParentPid": number,
      "Path": "string",
      "Pid": number,
      "TerminatedAt": "string"
    }
  }
]
"ProductArn": "string",
"ProductFields": {
   "string": "string"
},
"RecordState": "string",
"RelatedFindings": [
   {
      "Id": "string",
      "ProductArn": "string"
   }
],
"Remediation": {
   "Recommendation": {
      "Text": "string",
      "Url": "string"
   }
},
"Resources": [
   {
      "Details": {
         "AwsCloudFrontDistribution": {
            "DomainName": "string",
            "Etag": "string",
            "LastModifiedTime": "string",
            "Logging": {
               "Bucket": "string",
               "Enabled": boolean,
               "IncludeCookies": boolean,
               "Prefix": "string"
            },
            "Origins": {
               "Items": {
                  "OriginPath": "string",
                  "Id": "string",
                  "DomainName": "string"
               }
            },
            "Status": "string",
            "WebAclId": "string"
         }
      },
      "AwsCodeBuildProject": {
         "EncryptionKey": "string",
         "Environment": {
            "Type": "string",
            "Certificate": "string",
            "ImagePullCredentialsType": "string",
            "RegistryCredential": {
               "Credential": "string",
               "CredentialProvider": "string"
            }
         },
         "Name": "string",
         "ServiceRole": "string",
         "Source": {
            "Type": "string",
            "Location": "string",
            "GitCloneDepth": integer
         },
         "VpcConfig": {
            "VpcId": "string",
            "Subnets": ["string"],
            "SecurityGroupIds": ["string"]
         }
      }
   }
}
"IamInstanceProfileArn": "string",
"ImageId": "string",
"IpV4Addresses": [ "string" ],
"IpV6Addresses": [ "string" ],
"KeyName": "string",
"LaunchedAt": "string",
"SubnetId": "string",
"Type": "string",
"VpcId": "string"
},
"AwsEc2NetworkInterface": {
  "Attachment": { 
    "AttachmentId": "string",
    "AttachTime": "string",
    "DeleteOnTermination": true,
    "DeviceIndex": number,
    "InstanceId": "string"
  }
},
"SecurityGroups": [ 
  { 
    "GroupId": "string",
    "GroupName": "string"
  }
],
"NetworkInterfaceId": "string",
"SourceDestCheck": false
},
"AwsEc2SecurityGroup": {
  "GroupId": "string",
  "GroupName": "string",
  "IpPermissions": [ 
    { 
      "FromPort": number,
      "IpProtocol": "string",
      "IpRanges": [ 
        { 
          "CidrIp": "string"
        }
      ],
      "PrefixListIds": [ 
        { "PrefixListId": "string" }
      ]
    },
    "ToPort": number
  ],
  "UserIdGroupPairs": [ 
    { 
      "UserId": "string",
      "GroupId": "string"
    }
  ]
],
"IpPermissionsEgress": [ 
  { 
    "FromPort": number,
    "IpProtocol": "string",
    "IpRanges": [ 
      { 
        "CidrIp": "string"
      }
    ],
    "PrefixListIds": [ 
      { "PrefixListId": "string" }
    ]
  },
  "ToPort": number
]
"UserIdGroupPairs": [  
  {  
    "UserId": "string",  
    "GroupId": "string"  
  }  
],  
"OwnerId": "string",  
"VpcId": "string"},  
"AwsElasticSearchDomain": {  
  "AccessPolicies": "string",  
  "DomainStatus": {  
    "DomainId": "string",  
    "DomainName": "string",  
    "Endpoint": "string",  
    "Endpoints": {  
      "string": "string"  
    }  
  },  
  "DomainEndpointOptions": {  
    "EnforceHTTPS": boolean,  
    "TLSSecurityPolicy": "string"  
  },  
  "ElasticsearchVersion": "string",  
  "EncryptionAtRestOptions": {  
    "Enabled": boolean,  
    "KmsKeyId": "string"  
  },  
  "NodeToNodeEncryptionOptions": {  
    "Enabled": boolean  
  },  
  "VPCOptions": {  
    "AvailabilityZones": [  
      "string"  
    ],  
    "SecurityGroupIds": [  
      "string"  
    ],  
    "SubnetIds": [  
      "string"  
    ],  
    "VPCId": "string"  
  }  
},  
"AwsElbv2LoadBalancer": {  
  "AvailabilityZones": {  
    "SubnetId": "string",  
    "ZoneName": "string"  
  },  
  "CanonicalHostedZoneId": "string",  
  "CreatedTime": "string",  
  "DNSName": "string",  
  "IPAddressType": "string",  
  "Scheme": "string",  
  "SecurityGroups": [  
    "string"  
  ],  
  "State": {  
    "Code": "string",  
    "Reason": "string"  
  },  
  "Type": "string",  
  "VpcId": "string"  
},  
"AwsIamAccessKey": {  
  "CreatedAt": "string",  
  "CreatedAt": "string"}
"PrincipalId": "string",
"PrincipalName": "string",
"PrincipalType": "string",
"Status": "string"
},
"AwsIamRole": {
  "AssumeRolePolicyDocument": "string",
  "CreateDate": "string",
  "MaxSessionDuration": number,
  "Path": "string",
  "RoleId": "string",
  "RoleName": "string"
},
"AwsKmsKey": {
  "AWSAccountId": "string",
  "CreationDate": "string",
  "KeyId": "string",
  "KeyManager": "string",
  "KeyState": "string",
  "Origin": "string"
},
"AwsLambdaFunction": {
  "Code": {
    "S3Bucket": "string",
    "S3Key": "string",
    "S3ObjectVersion": "string",
    "ZipFile": "string"
  },
  "CodeSha256": "string",
  "DeadLetterConfig": {
    "TargetArn": "string",
  },
  "Environment": {
    "Variables": {
      "string": "string"
    },
  },
  "Error": {
    "ErrorCode": "string",
    "Message": "string"
  },
  "FunctionName": "string",
  "Handler": "string",
  "KmsKeyArn": "string",
  "LastModified": "string",
  "Layers": {
    "Arn": "string",
    "CodeSize": number
  },
  "RevisionId": "string",
  "Role": "string",
  "Runtime": "string",
  "Timeout": "integer",
  "TracingConfig": {
    "TracingConfig.Mode": "string"
  },
  "Version": "string",
  "VpcConfig": {
    "SecurityGroupIds": [ "string" ],
    "SubnetIds": [ "string" ]
  },
  "MasterArn": "string",
  "MemorySize": number
},
"AwsLambdaLayerVersion": {
  "CompatibleRuntimes": ["string"]}
"string",
"CreatedDate": "string",
"Version": number
},
"AwsRdsDbInstance": {
  "AssociatedRoles": [
  {
    "RoleArn": "string",
    "FeatureName": "string",
    "Status": "string"
  }
  ],
  "CACertificateIdentifier": "string",
  "DBClusterIdentifier": "string",
  "DBInstanceClass": "string",
  "DBInstanceIdentifier": "string",
  "DBInstancePort": number,
  "DBInstanceResourceId": "string",
  "DBName": "string",
  "DeletionProtection": boolean,
  "Endpoint": {
    "Address": "string",
    "Port": number,
    "HostedZoneId": "string"
  },
  "Engine": "string",
  "EngineVersion": "string",
  "IAMDatabaseAuthenticationEnabled": boolean,
  "InstanceCreateTime": "string",
  "KmsKeyId": "string",
  "PubliclyAccessible": boolean,
  "TdeCredentialArn": "string",
  "StorageEncrypted": boolean,
  "VpcSecurityGroups": [
  {
    "VpcSecurityGroupId": "string",
    "Status": "string"
  }
  ],
},
"AwsS3Bucket": {
  "CreatedAt": "string",
  "OwnerId": "string",
  "OwnerName": "string",
  "ServerSideEncryptionConfiguration": {
    "Rules": [
    {
      "ApplyServerSideEncryptionByDefault": {
        "KMSMasterKeyId": "string",
        "SSEAlgorithm": "string"
      }
    }
    ]
  }
},
"AwsS3Object": {
  "ContentType": "string",
  "ETag": "string",
  "LastModified": "string",
  "ServerSideEncryption": "string",
  "SSEKMSKeyId": "string",
  "VersionId": "string"
},
"AwsSnsTopic": {
  "KmsKeyId": "string",
  "TopicArn": "string",
  "Policy": "string"
}
"Owner": "string",
"Subscription": {
  "Endpoint": "string",
  "Protocol": "string"
},
"TopicName": "string"
},
"AwsSqsQueue": {
  "DeadLetterTargetArn": "string",
  "KmsDataKeyReusePeriodSeconds": number,
  "KmsMasterKeyId": "string",
  "QueueName": "string"
},
"AwsWafWebAcl": {
  "DefaultValue": "string",
  "Name": "string",
  "Rules": [
    {
      "Action": {
        "Type": "string"
      },
      "ExcludedRules": [
        {
          "RuleId": "string"
        }
      ],
      "OverrideAction": {
        "Type": "string"
      },
      "Priority": number,
      "RuleId": "string",
      "Type": "string"
    }
  ],
  "WebAclId": "string"
},
"Container": {
  "ImageId": "string",
  "ImageName": "string",
  "LaunchedAt": "string",
  "Name": "string"
},
"Other": {
  "string": "string"
}
},
"Id": "string",
"Partition": "string",
"Region": "string",
"Tags": {
  "string": "string"
},
"Type": "string"
],
"SchemaVersion": "string",
"Severity": {
  "Label": "string",
  "Normalized": number,
  "Product": number
},
"SourceUrl": "string",
"ThreatIntelIndicators": [
  {
    "Category": "string",
    "LastObservedAt": "string",
..."string": "string",
  }
]
ASFF attributes

The following table lists the top-level attributes and objects for the ASFF. For objects, to see the details for the object attributes and subfields, choose the object name.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AwsAccountId</td>
<td>Yes</td>
<td>The AWS account ID that the finding applies to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (12 digits max)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;AwsAccountId&quot;: &quot;111111111111&quot;</td>
</tr>
<tr>
<td>Compliance (p. 70)</td>
<td>No</td>
<td>Exclusive to findings that are generated as the result of a check run</td>
</tr>
<tr>
<td></td>
<td></td>
<td>against a specific rule in a supported standard (for example, CIS AWS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Compliance&quot;: {</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Status&quot;: &quot;PASSED&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;RelatedRequirements&quot;: [&quot;Req1&quot;, &quot;Req2&quot;]</td>
</tr>
<tr>
<td>Confidence</td>
<td>No</td>
<td>A finding's confidence. Confidence is defined as the likelihood that a</td>
</tr>
<tr>
<td></td>
<td></td>
<td>finding accurately identifies the behavior or issue that it was</td>
</tr>
<tr>
<td></td>
<td></td>
<td>intended to identify.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A finding provider can provide an initial value for this attribute, but</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cannot update it after that. This attribute can only be updated using</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BatchUpdateFindings. It can only be updated by a master account. It cannot</td>
</tr>
<tr>
<td></td>
<td></td>
<td>be updated by a member account.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: integer (range 0–100)</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>Confidence</td>
<td>No</td>
<td>Confidence is scored on a 0–100 basis using a ratio scale, where 0 means zero-percent confidence and 100 means 100-percent confidence. However, a data exfiltration detection based on a statistical deviation of network traffic has a much lower confidence because an actual exfiltration hasn't been verified. Example: &quot;Confidence&quot;: 42</td>
</tr>
<tr>
<td>CreatedAt</td>
<td>Yes</td>
<td>An ISO8601-formatted timestamp (as defined in <a href="https://tools.ietf.org/html/rfc3339">RFC-3339 Date and Time on the Internet: Timestamps</a>) that indicates when the potential security issue captured by a finding was created. Because the CreatedAt timestamp reflects the time when the finding record was created, it can differ from the FirstObservedAt timestamp, which reflects the time when the event or vulnerability was first observed. This timestamp must be provided on the first generation of the finding and can't be changed upon subsequent updates to the finding. Type: timestamp Example: &quot;CreatedAt&quot;: &quot;2017-03-22T13:22:13.933Z&quot;</td>
</tr>
</tbody>
</table>

**Note**  
Findings are deleted 90 days after the most recent update or 90 days after the creation date if no update occurs. To store findings for longer than 90 days, you can configure a rule in CloudWatch Events that routes findings to your Amazon S3 bucket.
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
</table>
| Criticality | No       | The level of importance that is assigned to the resources associated with the finding. A score of 0 means that the underlying resources have no criticality, and a score of 100 is reserved for the most critical resources. A finding provider can provide an initial value for this attribute, but cannot update it after that. This attribute can only be updated using `BatchUpdateFindings`. It can only be updated by a master account. It cannot be updated by a member account. Type: integer (range 0–100) Criticality is scored on a 0–100 basis, using a ratio scale that supports only full integers. This means that you should assess not only which findings impact resources that are more critical than others but also how much more critical those resources are compared to other resources. A score of 0 means that the underlying resources have no criticality, and a score of 100 is reserved for the most critical resources. When assessing criticality of a finding, consider the following:  
  - Does the impacted resource contain sensitive data (for example, an S3 bucket with PII)?  
  - Does the impacted resource enable an adversary to deepen their access or extend their capabilities to carry out additional malicious activity (for example, a compromised sysadmin account)?  
  - Is the resource a business-critical asset (for example, a key business system that if compromised could have significant revenue impact)? You can use the following guidelines:  
  - A resource powering mission-critical systems or containing highly sensitive data can be scored in the 75–100 range  
  - A resource powering important (but not critical systems) or containing moderately important data can be scored in the 25–75 range  
  - A resource powering non-important systems or containing non-sensitive data should be scored in the 0–24 range Example:  

  "Criticality": 99
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Yes</td>
<td>A finding's description. This field can be nonspecific boilerplate text or details that are specific to the instance of the finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (1,024 characters max)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Description&quot;: &quot;The version of openssl found on instance i-abcd1234 is known to contain a vulnerability.&quot;</td>
</tr>
<tr>
<td>FirstObservedAt</td>
<td>No</td>
<td>An ISO8601-formatted timestamp (as defined in RFC-3339 Date and Time on the Internet: Timestamps) that indicates when the potential security issue captured by a finding was first observed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: timestamp</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Because this timestamp reflects the time of when the event or vulnerability was first observed, it can differ from the CreatedAt timestamp, which reflects the time this finding record was created.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This timestamp should be immutable between updates of the finding record, but can be updated if a more accurate timestamp has been determined.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;FirstObservedAt&quot;: &quot;2017-03-22T13:22:13.933Z&quot;</td>
</tr>
<tr>
<td>GeneratorId</td>
<td>Yes</td>
<td>The identifier for the solution-specific component (a discrete unit of logic) that generated a finding. In various solutions from security findings products, this generator can be called a rule, a check, a detector, a plug-in, and so on.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (512 characters max) or Amazon Resource Name (ARN)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;GeneratorId&quot;: &quot;acme-vuln-9ab348&quot;</td>
</tr>
</tbody>
</table>
## ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Yes</td>
<td>The product-specific identifier for a finding. Type: string (512 characters max) or ARN. The finding ID must comply with the following constraints:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The ID must be globally unique within the product. To enforce uniqueness, you can incorporate the public AWS Region name and account ID in the identifier.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• You cannot recycle identifiers regardless of whether the previous finding no longer exists.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The ID must only contain characters from the unreserved characters set defined in section 2.3 of RFC-3986 Uniform Resource Identifier (URI): Generic Syntax.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For non-AWS services, the ID cannot be prefixed with the literal string &quot;arn:&quot;.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For AWS services, the ID must be the ARN of the finding if one is available. Otherwise, you can use any other unique identifier.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>These constraints are expected to hold within a findings product, but are not required to hold across findings products.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Id&quot;: &quot;us-west-2/111111111111/98aebb2207407c87f51e89943f12b1ef&quot;</td>
</tr>
<tr>
<td>LastObservedAt</td>
<td>No</td>
<td>An ISO8601-formatted timestamp (as defined in RFC-3339 Date and Time on the Internet: Timestamps) that indicates when the potential security issue captured by a finding was most recently observed by the security findings product.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: timestamp</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Because this timestamp reflects the time of when the event or vulnerability was last or most recently observed, it can differ from the UpdatedAt timestamp, which reflects the time this finding record was last or most recently updated.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can provide this timestamp, but it isn't required upon the first observation. If you provide the field in this case, the timestamp should be the same as the FirstObservedAt timestamp. You should update this field to reflect the last or most recently observed timestamp each time a finding is observed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;LastObservedAt&quot;: &quot;2017-03-23T13:22:13.933Z&quot;</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Malware (p. 71)</strong></td>
<td>No</td>
<td>A list of malware related to a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: array of up to five malware objects</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Malware&quot;: [</td>
</tr>
<tr>
<td></td>
<td></td>
<td>{</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Name&quot;: &quot;Stringler&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Type&quot;: &quot;COIN_MINER&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Path&quot;: &quot;/usr/sbin/stringler&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;State&quot;: &quot;OBSERVED&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>}</td>
</tr>
<tr>
<td><strong>Network (p. 72)</strong></td>
<td>No</td>
<td>The details of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Network&quot;: {</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Direction&quot;: &quot;IN&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Protocol&quot;: &quot;TCP&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;SourceIpV4&quot;: &quot;1.2.3.4&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;SourceIpV6&quot;: &quot;FE80:CD00:0000:0CDE:1257:0000:211E:729C&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;SourcePort&quot;: &quot;42&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;SourceDomain&quot;: &quot;here.com&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;SourceMac&quot;: &quot;00:0d:83:b1:c0:8e&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;DestinationIpV4&quot;: &quot;2.3.4.5&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;DestinationIpV6&quot;: &quot;FE80:CD00:0000:0CDE:1257:0000:211E:729C&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;DestinationPort&quot;: &quot;80&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;DestinationDomain&quot;: &quot;there.com&quot;</td>
</tr>
<tr>
<td><strong>Note (p. 74)</strong></td>
<td>No</td>
<td>A user-defined note that is added to a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A finding provider can provide an initial note for a finding, but</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cannot add notes after that.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A note can only be updated using BatchUpdateFindings. Notes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>can be added by both master accounts and member accounts.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Note&quot;: {</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Text&quot;: &quot;Don't forget to check under the mat.&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;UpdatedBy&quot;: &quot;jsmith&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;UpdatedAt&quot;: &quot;2018-08-31T00:15:09Z&quot;</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Process (p. 75)</td>
<td>No</td>
<td>The details of process-related information about a finding. Type: object</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Process&quot;: { &quot;Name&quot;: &quot;syslogd&quot;, &quot;Path&quot;: &quot;/usr/sbin/syslogd&quot;, &quot;Pid&quot;: 12345, &quot;ParentPid&quot;: 56789,</td>
</tr>
<tr>
<td>ProductArn</td>
<td>Yes</td>
<td>The ARN generated by Security Hub that uniquely identifies a third-party findings product after the product is registered with Security Hub. Type: ARN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The format of this field is \texttt{arn:partition:securityhub:region:account-id:product/company-id/product-id}.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For AWS services that are integrated with Security Hub, the company-id must be &quot;aws&quot;, and the product-id must be the AWS public service name. Because AWS products and services aren't associated with an account, the account-id section of the ARN is empty. AWS services that are not yet integrated with Security Hub are considered third-party products.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For public products, the company-id and product-id must be the ID values specified at the time of registration.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For private products, the company-id must be the account ID. The product-id must be the reserved word &quot;default&quot; or the ID that was specified at the time of registration.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>// Private ARN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;ProductArn&quot;: &quot;arn:aws:securityhub:us-east-1:111111111111:product/111111111111/default&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>// Public ARN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;ProductArn&quot;: &quot;arn:aws:securityhub:us-west-2::product/aws/guardduty&quot;</td>
</tr>
</tbody>
</table>
## ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProductFields</td>
<td>No</td>
<td>A data type where security findings products can include additional solution-specific details that aren't part of the defined AWS Security Finding Format.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: map of up to 50 key/value pairs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This field should not contain redundant data and must not contain data that conflicts with AWS Security Finding Format fields.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The &quot;aws/&quot; prefix represents a reserved namespace for AWS products and services only and must not be submitted with findings from partner products.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Although not required, products should format field names as company-id/product-id/field-name, where the company-id and product-id match those supplied in the ProductArn of the finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Field names can include alphanumeric characters, white space, and the following symbols: _ . / = + \ - @</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;ProductFields&quot;: {</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;generico/secure-pro/Count&quot;: &quot;6&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;generico/secure-pro/Action.Type&quot;, &quot;AWS_API_CALL&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;API&quot;, &quot;DeleteTrail&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Service_Name&quot;: &quot;cloudtrail.amazonaws.com&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;aws/inspector/AssessmentTemplateName&quot;: &quot;My daily CVE assessment&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;aws/inspector/AssessmentTargetName&quot;: &quot;My prod env&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;aws/inspector/RulesPackageName&quot;: &quot;Common Vulnerabilities and Exposures&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>}</td>
</tr>
<tr>
<td>RecordState</td>
<td>No</td>
<td>The record state of a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>By default, when initially generated by a service, findings are considered ACTIVE.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The ARCHIVED state indicates that a finding should be hidden from view. Archived findings are not immediately deleted. You can search, review, and report against them.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Finding providers can update the record state. Security Hub also automatically archives control-based findings if the associated resource is deleted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: enum</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;RecordState&quot;: &quot;ACTIVE&quot;</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RelatedFindings (p. 76)</td>
<td>No</td>
<td>A list of related findings. A finding provider can provide an initial list of related findings, but cannot update the list after that. The list of related findings can only be updated using BatchUpdateFindings. It can only be updated by a master account. It cannot be updated by a member account. Type: array of up to 10 RelatedFinding objects</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;RelatedFindings&quot;: [</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Id&quot;: &quot;123e4567-e89b-12d3-a456-426655440000&quot; },</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Id&quot;: &quot;AcmeNerfHerder-111111111111-x189dx7824&quot; }</td>
</tr>
<tr>
<td>Remediation (p. 77)</td>
<td>No</td>
<td>The remediation options for a finding. Type: object</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Remediation&quot;: {</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Recommendation&quot;: {</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Text&quot;: &quot;Run sudo yum update and cross your fingers and toes.&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Url&quot;: &quot;<a href="http://myfp.com/recommendations/dangerous_things_and_how_to_fix_them.html">http://myfp.com/recommendations/dangerous_things_and_how_to_fix_them.html</a>&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>}</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| **Resources**   | Yes      | A set of resource data types that describe the resources that the finding refers to.  
Type: array of up to 32 resource objects  
Example:  
```
"Resources": [
  {
    "Type": "AwsEc2Instance",
    "Id": "i-cafebabe",
    "Partition": "aws",
    "Region": "us-west-2",
    "Tags": {
      "billingCode": "Lotus-1-2-3",
      "needsPatching": "true"
    },
    "Details": {
      "AwsEc2Instance": {
        "Type": "i3.xlarge",
        "ImageId": "ami-abcd1234",
        "IpV4Addresses": [ "54.194.252.215",
                          "192.168.1.88"
                        ],
        "IpV6Addresses": [ "2001:db8:1234:1a2b::123" ],
        "KeyName": "my_keypair",
        "IamInstanceProfileArn": "arn:aws:iam::1111111111:instance-profile/AdminRole",
        "VpcId": "vpc-11112222",
        "SubnetId": "subnet-56f5f633",
        "LaunchedAt": "2018-05-08T16:46:19.000Z"
      }
    }
  }
]
```
| **SchemaVersion** | Yes | The schema version that a finding is formatted for. The value of this field must be one of the officially published versions identified by AWS.  
In the current release, the AWS Security Finding Format schema version is 2018-10-08.  
Type: string (10 characters max, conforms to YYYY-MM-DD)  
Example:  
```
"SchemaVersion": "2018-10-08"
```
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Severity (p. 116)</strong></td>
<td>Yes</td>
<td>A finding's severity. The finding must have either Label or Normalized populated. Label is the preferred attribute. If neither attribute is populated, then the finding is invalid. A finding provider can provide initial severity information for a finding, but cannot update it after that. The severity information can only be updated using <code>BatchUpdateFindings</code>. It can only be updated by a master account. It cannot be updated by a member account. Type: object Example:</td>
</tr>
<tr>
<td>SourceUrl</td>
<td>No</td>
<td>A URL that links to a page about the current finding in the finding product. Type: URL</td>
</tr>
<tr>
<td><strong>ThreatIntelIndicators (p. 118)</strong></td>
<td>No</td>
<td>Threat intelligence details that are related to a finding. Type: array of up to five threat intelligence indicator objects Example:</td>
</tr>
<tr>
<td>Title</td>
<td>Yes</td>
<td>A finding's title. This field can contain nonspecific boilerplate text or details that are specific to this instance of the finding. Type: string (256 characters max)</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>Types</td>
<td>Yes</td>
<td>One or more finding types in the format of <code>namespace/category/classifier</code> that classify a finding. Type: array of 50 strings max</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <em>namespace</em> must be a value from the predefined set of namespace values</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: Software and Configuration Checks</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• *category might be any value, but it is recommended that finding products use categories from the finding type taxonomy in Types taxonomy for ASFF (p. 120)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• *classifier might be any value, but it is recommended that finding providers use the identifier verbatim defined by published standards whenever possible</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Namespaces are required for all finding types, but categories and classifiers are optional. If you specify a classifier, you must also specify a category.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The '/' character is reserved and must not be used in a category or classifier. Escaping the '/' character is not supported.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Types&quot;: [</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Software and Configuration Checks/Vulnerabilities/CVE&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>]</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UpdatedAt</td>
<td>Yes</td>
<td>An ISO8601-formatted timestamp (as defined in <a href="https://tools.ietf.org/html/rfc3339">RFC-3339 Date and Time on the Internet: Timestamps</a>) that indicates when the findings product last updated the finding record. Because this timestamp reflects the time when the finding record was last or most recently updated, it can differ from the LastObservedAt timestamp, which reflects when the event or vulnerability was last or most recently observed. When you update the finding record, you must update this timestamp to the current timestamp. Upon creation of a finding record, the CreatedAt and UpdatedAt timestamps must be the same timestamp. After an update to the finding record, the value of this field must be greater than all of the previous values that it contained. Note that UpdatedAt is not updated by changes from BatchUpdateFindings. It is only updated by BatchImportFindings. Type: timestamp Findings are deleted 90 days after the most recent update or 90 days after the creation date if no update occurs. To store findings for longer than 90 days, you can configure a rule in CloudWatch Events that routes findings to your Amazon S3 bucket.</td>
</tr>
<tr>
<td>UserDefinedFields</td>
<td>No</td>
<td>A list of name/value string pairs that are associated with the finding. These are custom, user-defined fields that are added to a finding. These fields can be generated automatically via your specific configuration. Findings products must not use this field for data that the product generates. Instead, findings products can use the ProductFields field for data that doesn't map to any standard AWS Security Finding Format field. These fields can only be updated using BatchUpdateFindings. They can only be updated by a master account. They cannot be updated by a member account. Type: map of up to 50 key/value pairs Example:</td>
</tr>
</tbody>
</table>

```
"UserDefinedFields": {  
   "reviewedByCio": "true",  
   "comeBackToLater": "Check this again on Monday"
}
```
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
</table>
| VerificationState  | No       | The veracity of a finding. Findings products can provide the value of **UNKNOWN** for this field. A findings product should provide this value if there is a meaningful analog in the findings product's system. This field is typically populated by a user determination or action after they investigate a finding.  
A finding provider can provide an initial value for this attribute, but cannot update it after that. This attribute can only be updated using `BatchUpdateFindings`. It can only be updated by a master account. It cannot be updated by a member account.  
Type: enum  
Valid values:  
• **UNKNOWN** – The default disposition of a security finding unless a user changes it  
• **TRUE_POSITIVE** – A user sets this value if the security finding has been confirmed  
• **FALSE_POSITIVE** – A user sets this value if the security finding has been determined to be a false alarm  
• **BENIGN_POSITIVE** – A user sets this value as a special case of **TRUE_POSITIVE** where the finding doesn't pose any threat, is expected, or both |
| Workflow (p. 119)  | No       | Provides information about the status of the investigation into a finding.  
The workflow status is not intended for finding providers. The workflow status can only be updated using `BatchUpdateFindings`. Customers can also update it from the console. See the section called "Setting the workflow status for a finding" (p. 45). The workflow status can only be updated by a master account. It cannot be updated by a member account.  
Type: object  
Example:  

```json  
Workflow: {  
  "Status": "NEW"  
}  
```
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
</table>
| WorkflowState (deprecated) | No       | This field is being deprecated in favor of the Status field of the Workflow object. The workflow state of a finding. Findings products can provide the value of NEW for this field. A findings product can provide a value for this field if there is a meaningful analog in the findings product's system. Type: enum Valid values: 
- **NEW** – This can be associated with findings in the Active record state. This is the default workflow state for any new finding.
- **ASSIGNED** – This can be associated with findings in the Active record state. The finding has been acknowledged and given to someone to review or address.
- **IN_PROGRESS** – This can be associated with findings in the Active record state. Team members are actively working on the finding.
- **RESOLVED** – This can be associated with findings in the Archived record state. This differs from DEFERRED findings in that if the finding were to occur again (be updated by the native service) or any new finding matching this, the finding appears to customers as an active, new finding.
- **DEFERRED** – This can be associated with findings in the Archived record state, and it means that any additional findings that match this finding aren't shown for a set amount of time or indefinitely. Either the customer doesn't consider the finding to be applicable, or it's a known issue that they don't want to include in the active dataset.
- **DUPLICATE** – This can be associated with findings in the Archived record state. It means that the finding is a duplicate of another finding.

Example:

"WorkflowState": "NEW"

---

Details for AWS Security Finding Format objects

- Compliance (p. 70)
- Malware (p. 71)
- Network (p. 72)
- Note (p. 74)
- Process (p. 75)
- RelatedFindings (p. 76)
- Remediation (p. 77)
  - Recommendation (p. 78)
• Resources (p. 78)
  • AwsCloudFrontDistribution (p. 83)
    • Logging (p. 84)
    • Origins (p. 84)
  • AwsCodeBuildProject (p. 85)
    • Environment (p. 86)
    • Source (p. 87)
    • VpcConfig (p. 88)
  • AwsEc2Instance (p. 89)
  • AwsEc2NetworkInterface (p. 90)
    • Attachment (p. 91)
    • SecurityGroups (p. 91)
  • AwsEc2SecurityGroup (p. 91)
    • IP permission object (p. 93)
  • AwsElasticSearchDomain (p. 94)
    • DomainEndpointOptions (p. 95)
    • DomainStatus (p. 95)
    • EncryptionAtRestOptions (p. 96)
    • NodeToNodeEncryptionOptions (p. 96)
    • VpcOptions (p. 97)
  • AwsElbv2LoadBalancer (p. 97)
    • AvailabilityZones (p. 98)
    • State (p. 98)
  • AwsIamAccessKey (p. 99)
  • AwsIamRole (p. 99)
  • AwsKmsKey (p. 100)
  • AwsLambdaFunction (p. 101)
    • Code (p. 102)
    • DeadLetterConfig (p. 103)
    • Environment (p. 103)
    • Layers (p. 104)
    • TracingConfig (p. 104)
    • VpcConfig (p. 104)
  • AwsLambdaLayerVersion (p. 104)
  • AwsRdsDbInstance (p. 105)
    • AssociatedRoles (p. 107)
    • Endpoint (p. 108)
    • VpcSecurityGroups (p. 108)
  • AwsS3Bucket (p. 109)
    • ApplyServerSideEncryptionByDefault (p. 109)
  • AwsS3Object (p. 110)
  • AwsSnsTopic (p. 111)
    • Subscription (p. 111)
  • AwsSqsQueue (p. 111)
  • AwsWafWebAcl (p. 112)
    • Rule object (p. 113)
• Container (p. 115)
• Other (p. 115)
• Severity (p. 116)
  • How Security Hub maps Label and Normalized values (p. 117)
  • Guidance for Assigning the Normalized Severity (AWS Services and Partners) (p. 117)
• ThreatIntelIndicators (p. 118)
• Workflow (p. 119)

Compliance

Exclusive to findings that are generated as the result of a check run against a specific rule in a supported standard (for example, CIS AWS Foundations). Contains standards-related finding details.

Example:

```
"Compliance": {
  "Status": "PASSED",
  "RelatedRequirements": ["Req1", "Req2"]
}
```

The Compliance object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>No</td>
<td>The result of a security check.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: enum</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PASSED – Security check passed for all evaluated resources.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• WARNING – Some information is missing or this check is not supported given your configuration.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FAILED – Security check failed for at least one evaluated resource.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NOTAVAILABLE – Check could not be performed due to a service outage or API error. The NOTAVAILABLE status can also indicate that the result of the AWS Config evaluation was NOTAPPLICABLE. In that case, after 3 days, Security Hub automatically archives the finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Status&quot;: &quot;PASSED&quot;</td>
</tr>
<tr>
<td>RelatedRequirements</td>
<td>No</td>
<td>For a Security Hub control, the related requirements within an industry or regulatory framework. The check for that control is aligned with those requirements.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can provide up to 32 related requirements.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To identify a requirement, use its identifier.</td>
</tr>
</tbody>
</table>
## Malware

The **Malware** object provides a list of malware related to a finding. It is an array that can contain up to 5 malware objects.

Example:

```json
"Malware": [
  {
    "Name": "Stringler",
    "Type": "COIN_MINER",
    "Path": "/usr/sbin/stringler",
    "State": "OBSERVED"
  }
]
```

Each malware object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Yes</td>
<td>The name of the malware that was observed. Type: string (64 characters max) Example: &quot;Name&quot;: &quot;Stringler&quot;</td>
</tr>
<tr>
<td>Path</td>
<td>No</td>
<td>The filesystem path of the malware that was observed. Type: string (512 characters max) Example: &quot;Path&quot;: &quot;/usr/sbin/stringler&quot;</td>
</tr>
<tr>
<td>State</td>
<td>No</td>
<td>The state of the malware that was observed. Type: enum Valid values: OBSERVED</td>
</tr>
<tr>
<td>Type</td>
<td>No</td>
<td>The type of the malware that was observed. Type: enum Valid values: ADWARE</td>
</tr>
</tbody>
</table>
### Attribute

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POTENTIALLY_UNWANTED</td>
<td></td>
<td>Spyware, Ransomware, Remote Access, Rootkit, Trojan, Virus, Worm</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td>&quot;Type&quot;: &quot;COIN_MINER&quot;</td>
</tr>
</tbody>
</table>

### Network

The details of network-related information about a finding.

**Type:** object

**Example:**

```json
"Network": {
    "Direction": "IN",
    "Protocol": "TCP",
    "SourceIpV4": "1.2.3.4",
    "SourceIpV6": "FE80:CD00:0000:0CDE:1257:0000:211E:729C",
    "SourcePort": "42",
    "SourceDomain": "here.com",
    "SourceMac": "00:0d:83:b1:c0:8e",
    "DestinationIpV4": "2.3.4.5",
    "DestinationIpV6": "FE80:CD00:0000:0CDE:1257:0000:211E:729C",
    "DestinationPort": "80",
    "DestinationDomain": "there.com"
}
```

The **Network** object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DestinationDomain</td>
<td>No</td>
<td>The destination domain of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (128 characters max)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;DestinationDomain&quot;: &quot;there.com&quot;</td>
</tr>
<tr>
<td>DestinationIpV4</td>
<td>No</td>
<td>The destination IPv4 address of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: IPv4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;DestinationIpV4&quot;: &quot;2.3.4.5&quot;</td>
</tr>
<tr>
<td>DestinationIpV6</td>
<td>No</td>
<td>The destination IPv6 address of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: IPv6</td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;DestinationIpV6&quot;: &quot;FE80:CD00:0000:0CDE:1257:0000:211E:729C&quot;</td>
</tr>
<tr>
<td><strong>DestinationPort</strong></td>
<td><strong>No</strong></td>
<td>The destination port of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: number (range of 0–65535)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;DestinationPort&quot;: &quot;80&quot;</td>
</tr>
<tr>
<td><strong>Direction</strong></td>
<td><strong>No</strong></td>
<td>The direction of network traffic associated with a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: enum</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: IN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Direction&quot;: &quot;IN&quot;</td>
</tr>
<tr>
<td><strong>Protocol</strong></td>
<td><strong>No</strong></td>
<td>The protocol of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (16 characters max)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The name should be the IANA registered name for the associated port except in the case where the finding product can determine a more accurate protocol.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Protocol&quot;: &quot;TCP&quot;</td>
</tr>
<tr>
<td><strong>SourceDomain</strong></td>
<td><strong>No</strong></td>
<td>The source domain of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (128 characters max)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;SourceDomain&quot;: &quot;here.com&quot;</td>
</tr>
<tr>
<td><strong>SourceIpV4</strong></td>
<td><strong>No</strong></td>
<td>The source IPv4 address of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: IPv4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;SourceIpV4&quot;: &quot;1.2.3.4&quot;</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SourceIpV6</td>
<td>No</td>
<td>The source IPv6 address of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: IPv6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;SourceIpV6&quot;: &quot;FE80:CD00:0000:0CDE:1257:0000:211E:729C&quot;</td>
</tr>
<tr>
<td>SourceMac</td>
<td>No</td>
<td>The source media access control (MAC) address of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (must match MM:MM:MM:SS:SS:SS)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;SourceMac&quot;: &quot;00:0d:83:b1:c0:8e&quot;</td>
</tr>
<tr>
<td>SourcePort</td>
<td>No</td>
<td>The source port of network-related information about a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: number (range of 0–65535)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;SourcePort&quot;: &quot;80&quot;</td>
</tr>
</tbody>
</table>

### Note

The `Note` object adds a user-defined note to the finding.

A finding provider can provide an initial note for a finding, but cannot add notes after that. A note can only be updated using `BatchUpdateFindings`. Notes can be added by both master accounts and member accounts.

Example:

```
"Note": {
    "Text": "Don't forget to check under the mat.",
    "UpdatedBy": "jsmith",
    "UpdatedAt": "2018-08-31T00:15:09Z"
}
```

The `Note` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>Yes</td>
<td>The text of a finding note.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (512 characters max)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
</tbody>
</table>
### Attribute

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td></td>
<td>&quot;Example text.&quot;</td>
</tr>
<tr>
<td>UpdatedAt</td>
<td>Yes</td>
<td>The timestamp of when the note was updated. Type: timestamp Example: &quot;UpdatedAt&quot;: &quot;2018-08-31T00:15:09Z&quot;</td>
</tr>
<tr>
<td>UpdatedBy</td>
<td>Yes</td>
<td>The principal that created a note. Type: string (512 characters max) or ARN Example: &quot;UpdatedBy&quot;: &quot;jsmith&quot;</td>
</tr>
</tbody>
</table>

### Process

The Process object provides process-related details about the finding.

Example:

```json
"Process": {  
    "Name": "syslogd",  
    "Path": "/usr/sbin/syslogd",  
    "Pid": 12345,  
    "ParentPid": 56789,  
    "LaunchedAt": "2018-09-27T22:37:31Z",  
    "TerminatedAt": "2018-09-27T23:37:31Z"  
}
```

The Process object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LaunchedAt</td>
<td>No</td>
<td>The timestamp for the date and time when the process was launched. Type: timestamp Example: &quot;LaunchedAt&quot;: &quot;2018-09-27T22:37:31Z&quot;</td>
</tr>
<tr>
<td>Name</td>
<td>No</td>
<td>The name of the process. Type: string (64 characters max) Example:</td>
</tr>
</tbody>
</table>

75
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Name&quot;</td>
<td></td>
<td>&quot;syslogd&quot;</td>
</tr>
<tr>
<td>ParentPid</td>
<td>No</td>
<td>The parent process ID. Type: number Example: &quot;ParentPid&quot;: 56789</td>
</tr>
<tr>
<td>Path</td>
<td>No</td>
<td>The path to the process executable. Type: string (512 characters max) Example: &quot;Path&quot;: &quot;/usr/sbin/syslogd&quot;</td>
</tr>
<tr>
<td>Pid</td>
<td>No</td>
<td>The process ID. Type: number Example: &quot;Pid&quot;: 12345</td>
</tr>
<tr>
<td>TerminatedAt</td>
<td>No</td>
<td>The timestamp for the date and time when the process was terminated. Type: timestamp Example: &quot;TerminatedAt&quot;: &quot;2018-09-27T23:37:31Z&quot;</td>
</tr>
</tbody>
</table>

**RelatedFindings**

The **RelatedFindings** object provides a list of findings that are related to the current finding.

A finding provider can provide an initial list of related findings, but cannot update it after that. **RelatedFindings** can only be updated using the **BatchUpdateFindings**. It can only be updated by a master account. It cannot be updated by a member account.

**Example:**

```json
]```

Each related finding object can have the following attributes.
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Yes</td>
<td>The product-generated identifier for a related finding. Type: string (512 characters max) or ARN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Id&quot;: &quot;123e4567-e89b-12d3-a456-426655440000&quot;</td>
</tr>
<tr>
<td>ProductArn</td>
<td>Yes</td>
<td>The ARN of the product that generated a related finding. Type: ARN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;ProductArn&quot;: &quot;arn:aws:securityhub:us-west-2::product/aws/guardduty&quot;</td>
</tr>
</tbody>
</table>

### Remediation

The Remediation object provides information about recommended remediation steps to address the finding.

Example:

```
"Remediation": {
    "Recommendation": {
        "Text": "Run sudo yum update and cross your fingers and toes.",
        "Url": "http://myfp.com/recommendations/dangerous_things_and_how_to_fix_them.html"
    }
}
```

The Remediation object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommendation</td>
<td>No</td>
<td>A recommendation on how to remediate the issue identified within a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The Recommendation field is meant to facilitate manual instructions or details to resolve a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the recommendation object is present, then either the Text or Url field must be present and populated. Both fields can be present and populated.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
</tbody>
</table>
|                      |          | "Recommendation": {
|                      |          |    "Text": "Example text.",
|                      |          |    "Url": "http://myfp.com/recommendations/dangerous_things_and_how_to_fix_them.html"|
```
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
</table>
| Text      | No       | A free-form string that is the recommendation of what to do about the finding when presented to a user. This field can contain nonspecific boilerplate text or details that are specific to this instance of the finding.  
Type: string (512 characters max)  
Example:  
"Text": "Example text." |
| Url       | No       | A URL to link to general remediation information for the finding type of a finding.  
This URL must not require credentials to access. It must be accessible from the public internet and must not expect any context or session.  
Type: URL  
Example:  
"Url": "http://myfp.com/recommendations/example_domain.html" |

**Resources**

The Resources object provides information about the resources involved in a finding.

Type: array of up to 10 resource objects

Example:

"Resources": [
{
  "Type": "AwsEc2Instance",
  "Id": "i-cafebabe",
  "Partition": "aws",
  "Region": "us-west-2",
  "Tags": {
    "billingCode": "Lotus-1-2-3",
    "needsPatching": "true"
  },
  "Details": {
    "AwsEc2Instance": {
      "Type": "i3.xlarge",
      "...
AWS Security Hub User Guide
ASFF attributes

```
"ImageId": "ami-abcd1234",
"IpV4Addresses": [ "54.194.252.215", "192.168.1.88" ],
"IpV6Addresses": [ "2001:db8:1234:1a2b::123" ],
"KeyName": "my_keypair",
"IamInstanceProfileArn": "arn:aws:iam::111111111111:instance-profile/AdminRole",
"VpcId": "vpc-11112222",
"SubnetId": "subnet-56f5f633",
"LaunchedAt": "2018-05-08T16:46:19.000Z"
}
```

Each resource object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>No</td>
<td>This field provides additional details about a single resource using the appropriate subfields.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Each resource must be provided in a separate resource object in the Resources field.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Security Hub provides a set of available subfields for its supported resource types. These subfields correspond to values of the resource</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type. Use the provided types and subfields whenever possible.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsCloudFrontDistribution</code> (p. 83)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsCodeBuildProject</code> (p. 85)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsEc2Instance</code> (p. 89)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsEc2NetworkInterface</code> (p. 90)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsEc2SecurityGroup</code> (p. 91)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsElasticsearchDomain</code> (p. 94)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsElbv2LoadBalancer</code> (p. 97)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsIamAccessKey</code> (p. 99)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsIamRole</code> (p. 99)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsKmsKey</code> (p. 100)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsLambdaFunction</code> (p. 101)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsLambdaLayerVersion</code> (p. 104)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsRdsDbInstance</code> (p. 105)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsS3Bucket</code> (p. 109)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsS3Object</code> (p. 110)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsSnsTopic</code> (p. 111)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsSqsQueue</code> (p. 111)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>AwsWafWebAcl</code> (p. 112)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>Container</code> (p. 115)</td>
</tr>
</tbody>
</table>

For example, if the resource is an S3 bucket, then set the resource Type to `AwsS3Bucket`, and provide the resource details in the `AwsS3Bucket` subfield.

The Other (p. 115) subfield allows you to provide custom fields and values. You use the Other subfield in the following cases.
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>• The resource type (the value of the resource Type) does not have a corresponding subfield. To provide details for the resource, you use the Other details subfield.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The subfield for the resource type does not include all of the fields you want to populate. In this case, use the subfield for the resource type to populate the available fields. Use the Other subfield to populate the fields that are not in the type-specific subfield.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The resource type is not one of the provided types. In this case, set the resource Type to Other, and use the Other details subfield to populate the details.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Type: object

Example:

```json
"Details": {
  "AwsEc2Instance": {
    "Type": "i3.xlarge",
    "ImageId": "ami-abcd1234",
    "Ipv4Addresses": [ "54.194.252.215", "192.168.1.88" ],
    "Ipv6Addresses": [ "2001:db8:1234:1a2b::123" ],
    "KeyName": "my_keypair",
    "IamInstanceProfileArn": "arn:aws:iam::111111111111:instance-profile/AdminRole",
    "VpcId": "vpc-11112222",
    "SubnetId": "subnet-56f5f633",
    "LaunchedAt": "2018-05-08T16:46:19.000Z"
  },
  "AwsS3Bucket": {
    "OwnerId": "da4d66eac431652a4d4d490a00500bded52c97d235b7b4752f9f688566fe6de",
    "OwnerName": "acmes3bucketowner"
  },
  "Other": [
    { "Key": "LightPen", "Value": "blinky" },
    { "Key": "SerialNo", "Value": "1234abcd" }
  ]
}
```

<table>
<thead>
<tr>
<th>Id</th>
<th>Yes</th>
<th>The canonical identifier for the given resource type.</th>
</tr>
</thead>
<tbody>
<tr>
<td>For AWS resources that are identified by ARNs, this must be the ARN.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>For all other AWS resource types that lack ARNs, this must be the identifier as defined by the AWS service that created the resource.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>For non-AWS resources, this should be a unique identifier associated with the resource.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type: string (512 characters max) or ARN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```json
"Id": "arn:aws:s3:::example-bucket"
```
## ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Partition</strong></td>
<td>No</td>
<td>The canonical AWS partition name that the Region is assigned to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: enum</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Partition</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Commercial</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-cn</td>
</tr>
<tr>
<td></td>
<td></td>
<td>China</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-us-gov</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWS GovCloud (US)</td>
</tr>
</tbody>
</table>

Example:

```
"Partition": "aws"
```

| **Region** | No       | The canonical AWS external Region name where this resource is located.      |
|            |          | Type: string (16 characters max)                                            |

Example:

```
"Region": "us-west-2"
```

| **Tags**   | No       | A list of AWS tags that are associated with a resource at the time the finding was processed. Include the Tags attribute only for resources that have an associated tag. If a resource has no associated tag, don't include a Tags attribute in the finding. |
|           |          | Type: map of up to 50 tags (values are limited to 256 characters max)       |
|           |          | The following basic restrictions apply to tags:                             |
|           |          | • You can provide only tags that actually exist on an AWS resource in this field. To provide data for a resource type that isn't defined in the AWS Security Finding Format, use the Other details subfield. |
|           |          | • Values are limited to alphanumeric characters, white space, +, -, =, ~, =>, \;, /, and @. |
|           |          | • Values are limited to the AWS Tag value length of 256 characters max.      |

Example:

```
"Tags": {
  "billingCode": "Lotus-1-2-3",
  "needsPatching": "true"
}
```
## ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Yes</td>
<td>The type of the resource that you are providing details for. Whenever possible, use one of the provided resource types, such as AwsEc2Instance or AwsS3Bucket. If the resource type does not match any the provided resource types, then set the resource Type to Other, and use the Other details subfield to populate the details. Type: string (256 characters max) Valid values are as follows. If a type has a corresponding subfield, then to view the details for the subfield, choose the type name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AutoscalingAutoscalingGroup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsAccount</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsApiGatewayMethod</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsApiGatewayRestApi</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsAppStreamFleet</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCertificateManagerCertificate</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCloudFormationStack</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCloudFrontDistribution (p. 83)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCloudTrailTrail</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCloudWatchAlarm</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCodeBuildProject (p. 85)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCodeCommitRepository</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCodeDeployApplication</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCodeDeployDeploymentGroup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCodePipelinePipeline</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCognitoIdentityPool</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsCognitoUserPool</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsDynamoDbTable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEc2Eip</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEc2Instance (p. 89)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEc2NetworkInterface (p. 90)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEc2SecurityGroup (p. 91)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEc2Snapshot</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEc2Volume</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEc2Vpc</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEcsService</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEcsTaskDefinition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEfsFileSystem</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsEksCluster</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsElastiCacheCacheCluster</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsElasticsearchDomain (p. 94)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsElbLoadBalancer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsElbv2LoadBalancer (p. 97)</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
|             |          | - AwsEmrCluster  
|             |          | - AwsIamAccessKey (p. 99)  
|             |          | - AwsIamUser  
|             |          | - AwsIamRole (p. 99)  
|             |          | - AwsIamPolicy  
|             |          | - AwsKinesisStream  
|             |          | - AwsKmsKey (p. 100)  
|             |          | - AwsLambdaFunction (p. 101)  
|             |          | - AwsLambdaLayerVersion (p. 104)  
|             |          | - AwsLogsLogGroup  
|             |          | - AwsRdsDbInstance (p. 105)  
|             |          | - AwsRdsDbSnapshot  
|             |          | - AwsRedshiftCluster  
|             |          | - AwsS3Bucket (p. 109)  
|             |          | - AwsS3Object (p. 110)  
|             |          | - AwsSnsTopic (p. 111)  
|             |          | - AwsSqsQueue (p. 111)  
|             |          | - AwsWafWebAcl (p. 112)  
|             |          | - Container (p. 115)  
|             |          | - Other (p. 115)  

Example:

"Type": "AwsS3Bucket"

**AwsCloudFrontDistribution**

The *AwsCloudFrontDistribution* object provides details about a distribution configuration. It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DomainName</td>
<td>No</td>
<td>The domain name corresponding to the distribution. Type: string</td>
</tr>
<tr>
<td>Etag</td>
<td>No</td>
<td>The entity tag is a hash of the object. Type: string</td>
</tr>
<tr>
<td>LastModifiedTime</td>
<td>No</td>
<td>The date and time that the distribution was last modified. Type: string</td>
</tr>
<tr>
<td>Logging (p. 84)</td>
<td>No</td>
<td>A complex type that controls whether access logs are written for the distribution. Type: object</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origins (p. 84)</td>
<td>No</td>
<td>A complex type that contains information about origins and origin groups for this distribution. Type: string</td>
</tr>
<tr>
<td>Status</td>
<td>No</td>
<td>Indicates the current status of the distribution. Type: string</td>
</tr>
<tr>
<td>WebAclId</td>
<td>No</td>
<td>A unique identifier that specifies the AWS WAF web ACL, if any, to associate with this distribution. Type: string</td>
</tr>
</tbody>
</table>

#### Logging

The **Logging** object provides information about the logging for the distribution.

It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bucket</td>
<td>No</td>
<td>The S3 bucket to store the access logs in. Type: string</td>
</tr>
<tr>
<td>Enabled</td>
<td>No</td>
<td>With this field, you can enable or disable the selected distribution. Type: Boolean</td>
</tr>
<tr>
<td>IncludeCookies</td>
<td>No</td>
<td>Specifies whether you want CloudFront to include cookies in access logs. Type: Boolean</td>
</tr>
<tr>
<td>Prefix</td>
<td>No</td>
<td>An optional string that you want CloudFront to prefix to the access log filenames for this distribution. Type: string</td>
</tr>
</tbody>
</table>

#### Origins

The **Origins** object contains information about origins and origin groups for this distribution.

It can contain the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Items</td>
<td>No</td>
<td>A complex type that contains origins or origin groups for this distribution. Type: object</td>
</tr>
</tbody>
</table>

Each item can have the following attributes.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OriginPath</td>
<td>No</td>
<td>An optional element that causes CloudFront to request your content from a directory in your S3 bucket or your custom origin. Type: string</td>
</tr>
<tr>
<td>Id</td>
<td>No</td>
<td>A unique identifier for the origin or origin group. Type: string</td>
</tr>
<tr>
<td>DomainName</td>
<td>No</td>
<td>Amazon S3 origins: The DNS name of the S3 bucket from which you want CloudFront to get objects for this origin. Type: string</td>
</tr>
</tbody>
</table>

**AwsCodeBuildProject**

The `AwsCodeBuildProject` object provides information about an AWS CodeBuild project.

Example:

```json
"AwsCodeBuildProject": {
   "EncryptionKey": "my-symm-key",
   "Environment": {
      "Type": "LINUX_CONTAINER",
      "Certificate": "myX509",
      "ImagePullCredentialsType": "CODEBUILD",
      "RegistryCredential": {
         "Credential": "my_dockerhub_secret",
         "CredentialProvider": "SECRETS_MANAGER"
      }
   },
   "Name": "my-cd-project",
   "Source": {
      "Type": "CODECOMMIT",
      "Location": "https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo",
      "GitCloneDepth": 1
   },
   "ServiceRole": "arn:aws:iam:myrole",
   "VpcConfig": {
      "VpcId": "vpc-123456",
      "Subnets": ["sub-12345666"],
      "SecurityGroupIds": ["sg-123456789012"]
   }
}
```

The `AwsCodeBuildProject` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EncryptionKey</td>
<td>No</td>
<td>The AWS KMS customer master key (CMK) to be used for encrypting the build output artifacts.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can use a cross-account KMS key to encrypt the build output artifacts if your service role has permission to that key.</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td><strong>You can specify either the ARN of the CMK or, if available, the CMK alias (using the format <code>alias/alias-name</code>).</strong> Type: string Length constraints: Minimum length of 1</td>
</tr>
<tr>
<td>Environment (p. 86)</td>
<td>No</td>
<td>Information about the build environment for this build project. Type: object</td>
</tr>
<tr>
<td>Name</td>
<td>No</td>
<td>The name of the build project. Type: string Length constraints: Minimum length of 2. Maximum length of 255. Pattern: [A-Za-z0-9][A-Za-z0-9-_]{1,254}</td>
</tr>
<tr>
<td>ServiceRole</td>
<td>No</td>
<td>The ARN of the IAM role that enables CodeBuild to interact with dependent AWS services on behalf of the AWS account. Type: string Length constraints: Minimum length of 1</td>
</tr>
<tr>
<td>Source (p. 87)</td>
<td>No</td>
<td>Information about the build input source code for this build project. Type: object</td>
</tr>
<tr>
<td>VpcConfig (p. 88)</td>
<td>No</td>
<td>Information about the VPC configuration that CodeBuild accesses. Type: object</td>
</tr>
</tbody>
</table>

### Environment

The **Environment** object provides information about the build environment for the build project. It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certificate</td>
<td>No</td>
<td>The certificate to use with this build project. Type: string</td>
</tr>
<tr>
<td>ImagePullCredentialsType</td>
<td>No</td>
<td>The type of credentials CodeBuild uses to pull images in your build. There are two valid values: CODEBUILD specifies that CodeBuild uses its own credentials. This requires that you modify your ECR repository policy to trust the CodeBuild service principal. SERVICE_ROLE specifies that CodeBuild uses your build project's service role.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>----------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When you use a cross-account or private registry image, you must use SERVICE_ROLE credentials. When you use a CodeBuild curated image, you must use CODEBUILD credentials.</td>
</tr>
<tr>
<td>RegistryCredential</td>
<td>No</td>
<td>The credentials for access to a private registry.</td>
</tr>
<tr>
<td>Type</td>
<td>Yes</td>
<td>The type of build environment to use for related builds.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: WINDOWS_CONTAINER</td>
</tr>
</tbody>
</table>

Each registry credential in the RegistryCredentials object has the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential</td>
<td>Yes</td>
<td>The ARN or name of credentials created using AWS Secrets Manager.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The credential can use the name of the credentials only if they exist in your current AWS Region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Length constraints: Minimum length of 1</td>
</tr>
<tr>
<td>CredentialProvider</td>
<td>Yes</td>
<td>The service that created the credentials to access a private Docker registry. The valid value, SECRETS_MANAGER, is for Secrets Manager.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: SECRETS_MANAGER</td>
</tr>
</tbody>
</table>

**Source**

The Source object provides information about the build input source code for this build project. It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GitCloneDepth</td>
<td>No</td>
<td>Information about the Git clone depth for the build project.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: integer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid range: Minimum value of 0</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Location  | No       | Information about the location of the source code to be built. Type: string Valid values:  
  • For source code settings that are specified in the source action of a pipeline in AWS CodePipeline, location should not be specified. If it is specified, CodePipeline ignores it. This is because CodePipeline uses the settings in a pipeline's source action instead of this value.  
  • For source code in an AWS CodeCommit repository, the HTTPS clone URL to the repository that contains the source code and the buildspec file (for example, https://git-codecommit.region-ID.amazonaws.com/v1/repos/repo-name).  
  • For source code in an S3 input bucket, one of the following.  
    • The path to the ZIP file that contains the source code (for example, bucket-name/path/to/object-name.zip).  
    • The path to the folder that contains the source code (for example, bucket-name/path/to/source-code/folder/).  
  • For source code in a GitHub repository, the HTTPS clone URL to the repository that contains the source and the buildspec file.  
  • For source code in a Bitbucket repository, the HTTPS clone URL to the repository that contains the source and the buildspec file. |
| Type      | Yes      | The type of repository that contains the source code to be built. Type: String Valid values:  
  • BITBUCKET - The source code is in a Bitbucket repository.  
  • CODECOMMIT - The source code is in a CodeCommit repository.  
  • CODEPIPELINE - The source code settings are specified in the source action of a pipeline in CodePipeline.  
  • GITHUB - The source code is in a GitHub repository.  
  • GITHUB_ENTERPRISE - The source code is in a GitHub Enterprise repository.  
  • NO_SOURCE - The project does not have input source code.  
  • S3 - The source code is in an Amazon S3 input bucket. |

**VpcConfig**

The `VpcConfig` object provides information about the VPC configuration that CodeBuild accesses. It can have the following attributes.
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SecurityGroupIds</td>
<td>No</td>
<td>A list of one or more security group IDs in your Amazon VPC.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: Array of strings</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Array members: Maximum number of 5 items</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Length constraints: Minimum length of 1</td>
</tr>
<tr>
<td>Subnets</td>
<td>No</td>
<td>A list of one or more subnet IDs in your Amazon VPC.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: Array of strings</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Array members: Maximum number of 16 items</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Length constraints: Minimum length of 1</td>
</tr>
<tr>
<td>VpcId</td>
<td>No</td>
<td>The ID of the VPC.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Length constraints: Minimum length of 1</td>
</tr>
</tbody>
</table>

**AwsEc2Instance**

The details of an Amazon EC2 instance.

Type: object

The `AwsEc2Instance` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IamInstanceProfileArn</td>
<td>No</td>
<td>The IAM profile ARN of the instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (conforms to the AWS ARN format)</td>
</tr>
<tr>
<td>ImageId</td>
<td>No</td>
<td>The Amazon Machine Image (AMI) ID of the instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (64 characters max)</td>
</tr>
<tr>
<td>IpV4Addresses</td>
<td>No</td>
<td>The IPv4 addresses that are associated with the instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: array of up to 10 IPv4 addresses</td>
</tr>
<tr>
<td>IpV6Addresses</td>
<td>No</td>
<td>The IPv6 addresses that are associated with the instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: array of up to 10 IPv6 addresses</td>
</tr>
<tr>
<td>KeyName</td>
<td>No</td>
<td>The key name that is associated with the instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (128 characters max)</td>
</tr>
<tr>
<td>LaunchedAt</td>
<td>No</td>
<td>The date and time when the instance was launched.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: timestamp</td>
</tr>
<tr>
<td>SubnetId</td>
<td>No</td>
<td>The identifier of the subnet where the instance was launched.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Type</td>
<td>No</td>
<td>The instance type of the instance. This must be a valid EC2 instance type. Type: string (16 characters max)</td>
</tr>
<tr>
<td>VpcId</td>
<td>No</td>
<td>The identifier of the VPC where the instance was launched. Type: string (32 characters max)</td>
</tr>
</tbody>
</table>

**AwsEc2NetworkInterface**

The `AwsEc2NetworkInterface` object provides information about an Amazon EC2 network interface.

Example:

```json
"AwsEc2NetworkInterface": {
    "Attachment": {
        "AttachTime": "2019-01-01T03:03:21Z",
        "AttachmentId": "eni-attach-43348162",
        "DeleteOnTermination": true,
        "DeviceIndex": 123,
        "InstanceId": "i-1234567890abcdef0",
        "InstanceOwnerId": "123456789012",
        "Status": 'ATTACHED'
    },
    "SecurityGroups": [
        {
            "GroupName": "my-security-group",
            "GroupId": "sg-903004f8"
        },
    ],
    "NetworkInterfaceId": 'eni-686ea200',
    "SourceDestCheck": false
}
```

The `AwsEc2NetworkInterface` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Attachment</strong> (p. 91)</td>
<td>No</td>
<td>Information about the network interface attachment. Type: object</td>
</tr>
<tr>
<td>NetworkInterfaceId</td>
<td>No</td>
<td>The ID of the network interface. Type: string</td>
</tr>
<tr>
<td><strong>SecurityGroups</strong> (p. 91)</td>
<td>No</td>
<td>Security groups for the network interface. Type: array of group objects</td>
</tr>
<tr>
<td>SourceDestCheck</td>
<td>No</td>
<td>Indicates whether traffic to or from the instance is validated. Type: Boolean</td>
</tr>
</tbody>
</table>
Attachment

The Attachment object provides information about the network interface attachment. It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AttachmentId</td>
<td>No</td>
<td>The identifier of the network interface attachment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>AttachTime</td>
<td>No</td>
<td>The timestamp indicating when the attachment initiated.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: timestamp</td>
</tr>
<tr>
<td>DeleteOnTermination</td>
<td>No</td>
<td>Indicates whether the network interface is deleted when the instance is terminated.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: Boolean</td>
</tr>
<tr>
<td>DeviceIndex</td>
<td>No</td>
<td>The device index of the network interface attachment on the instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: integer</td>
</tr>
<tr>
<td>InstanceId</td>
<td>No</td>
<td>The ID of the instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>InstanceOwnerId</td>
<td>No</td>
<td>The AWS account ID of the owner of the instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>Status</td>
<td>No</td>
<td>The attachment state.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: String [attaching]</td>
</tr>
</tbody>
</table>

SecurityGroups

The SecurityGroups object contains the list of security groups for the network interface. Each security group can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupId</td>
<td>No</td>
<td>The ID of the security group.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>GroupName</td>
<td></td>
<td>The name of the security group.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
</tbody>
</table>

AwsEc2SecurityGroup

The AwsEc2SecurityGroup object describes an Amazon EC2 security group.
Example:

```
"AwsEc2SecurityGroup": {
    "GroupName": "MySecurityGroup",
    "GroupId": "sg-903004f8",
    "OwnerId": "123456789012",
    "VpcId": "vpc-1a2b3c4d",
    "IpPermissions": [
        {
            "IpProtocol": "-1",
            "IpRanges": [],
            "UserIdGroupPairs": [
                {
                    "UserId": "123456789012",
                    "GroupId": "sg-903004f8"
                }
            ],
            "PrefixListIds": [],
            "PrefixListIds": [
                {"PrefixListId": "pl-63a5400a"}
            ]
        },
        {
            "PrefixListIds": [],
            "FromPort": 22,
            "IpRanges": [
                {
                    "CidrIp": "203.0.113.0/24"
                }
            ],
            "ToPort": 22,
            "IpProtocol": "tcp",
            "UserIdGroupPairs": []
        }
    ]
}
```

The `AwsEc2SecurityGroup` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupId</td>
<td>No</td>
<td>The ID of the security group.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>GroupName</td>
<td>No</td>
<td>The name of the security group.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td><strong>IpPermissions</strong> (p. 93)</td>
<td>No</td>
<td>The inbound rules associated with the security group.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: array of IP permission objects</td>
</tr>
<tr>
<td><strong>IpPermissionsEgress</strong> (p. 93)</td>
<td>No</td>
<td>[VPC only] The outbound rules associated with the security group.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: Array of IP permission objects</td>
</tr>
<tr>
<td>OwnerId</td>
<td>No</td>
<td>The AWS account ID of the owner of the security group.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td>VpcId</td>
<td>No</td>
<td>[VPC only] The ID of the VPC for the security group.</td>
</tr>
</tbody>
</table>

92
### IP permission object

The `IpPermissions` and `IpPermissionsEgress` objects both contain an array of IP permission objects.

Each IP permission object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FromPort</td>
<td>No</td>
<td>The start of the port range for the TCP and UDP protocols, or an ICMP/ICMPv6 type number. A value of -1 indicates all ICMP/ICMPv6 types. If you specify all ICMP/ICMPv6 types, you must specify all codes. Type: integer</td>
</tr>
<tr>
<td>IpProtocol</td>
<td>No</td>
<td>The IP protocol name (tcp, udp, icmp, icmpv6) or number (see the protocol numbers list). [VPC only] Use -1 to specify all protocols. When authorizing security group rules, specifying -1 or a protocol number other than tcp, udp, icmp, or icmpv6 allows traffic on all ports, regardless of any port range you specify. For tcp, udp, and icmp, you must specify a port range. For icmpv6, the port range is optional. If you omit the port range, traffic for all types and codes is allowed. Type: string</td>
</tr>
<tr>
<td>IpRanges</td>
<td>No</td>
<td>The ranges of IP addresses. Type: array of IP range objects</td>
</tr>
<tr>
<td>PrefixListIds</td>
<td>No</td>
<td>[VPC only] The prefix list IDs for an AWS service. With outbound rules, this is the AWS service to access through a VPC endpoint from instances associated with the security group. Type: array of prefix list ID objects</td>
</tr>
<tr>
<td>ToPort</td>
<td>No</td>
<td>The end of the port range for the TCP and UDP protocols, or an ICMP/ICMPv6 code. A value of -1 indicates all ICMP/ICMPv6 codes. If you specify all ICMP/ICMPv6 types, you must specify all codes. Type: integer</td>
</tr>
<tr>
<td>UserIdGroupPairs</td>
<td>No</td>
<td>The security group and AWS account ID pairs. Type: array of user ID group pair objects</td>
</tr>
</tbody>
</table>
Each entry in the `IpRanges` array can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CidrIp</td>
<td>No</td>
<td>A range of IP addresses. You can either specify a CIDR range or a source security group, but not both. To specify a single IPv4 address, use the /32 prefix length. To specify a single IPv6 address, use the /128 prefix length. Type: string</td>
</tr>
</tbody>
</table>

Each entry in the `PrefixListIds` array can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PrefixListId</td>
<td>No</td>
<td>The ID of the prefix. Type: String</td>
</tr>
</tbody>
</table>

Each entry in the `UserIdGroupPairs` array can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupId</td>
<td>No</td>
<td>The ID of the security group. Type: string</td>
</tr>
<tr>
<td>UserId</td>
<td>No</td>
<td>The ID of an AWS account. For a referenced security group in another VPC, the account ID of the referenced security group is returned in the response. If the referenced security group is deleted, this value is not returned. [Amazon EC2-Classic] Required when adding or removing rules that reference a security group in another AWS account. Type: string</td>
</tr>
</tbody>
</table>

**AwsElasticSearchDomain**

The `AwsElasticSearchDomain` object provides details about an Elasticsearch domain.

It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AccessPolicies</td>
<td>No</td>
<td>IAM policy document specifying the access policies for the new Amazon ES domain.</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DomainEndpointOptions (p. 95)</td>
<td>No</td>
<td>Additional options for the domain endpoint.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td>DomainStatus (p. 95)</td>
<td>No</td>
<td>Details about the domain status.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td>ElasticsearchVersion</td>
<td>No</td>
<td>Elasticsearch version.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>EncryptionAtRestOptions (p. 96)</td>
<td>No</td>
<td>Details about the configuration for encryption at rest.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td>NodeToNodeEncryptionOptions (p. 96)</td>
<td>No</td>
<td>Details about the configuration for node-to-node encryption.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td>VPCOptions (p. 97)</td>
<td>No</td>
<td>Information that Amazon ES derives based on VPCOptions for the domain.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
</tbody>
</table>

#### DomainEndpointOptions

The `DomainEndpointOptions` object provides information about additional options for the domain endpoint.

It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EnforceHTTPS</td>
<td>No</td>
<td>Whether to require that all traffic to the domain arrive over HTTPS.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: Boolean</td>
</tr>
<tr>
<td>TLSSecurityPolicy</td>
<td>No</td>
<td>The TLS security policy to apply to the HTTPS endpoint of the Elasticsearch domain.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Policy-Min-TLS-1-0-2019-07, which supports TLSv1.0 and higher</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Policy-Min-TLS-1-2-2019-07, which only supports TLSv1.2</td>
</tr>
</tbody>
</table>

#### DomainStatus

The `DomainStatus` object provides details about the domain status.

It can have the following attributes.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DomainId</td>
<td>No</td>
<td>Unique identifier for an Amazon ES domain.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>DomainName</td>
<td>No</td>
<td>Name of an Amazon ES domain.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Domain names are unique across all domains owned by the same account within an AWS Region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Domain names must start with a lowercase letter and must be between 3 and 28 characters.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid characters are a-z (lowercase only), 0-9, and – (hyphen).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>Endpoint</td>
<td>No</td>
<td>Domain-specific endpoint used to submit index, search, and data upload requests to an Amazon ES domain.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The endpoint is a service URL.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>Endpoints</td>
<td>No</td>
<td>The key-value pair that exists if the Amazon ES domain uses VPC endpoints.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: map of key-value pairs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;vpc&quot;: &quot;&lt;VPC_ENDPOINT&gt;&quot;</td>
</tr>
</tbody>
</table>

**EncryptionAtRestOptions**

The EncryptionAtRestOptions object provides details about the configuration for encryption at rest. It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>No</td>
<td>Whether encryption at rest is enabled.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: Boolean</td>
</tr>
<tr>
<td>KmsKeyId</td>
<td>No</td>
<td>The AWS KMS key ID. Takes the form 1a2a3a4-1a2a-3a4a-5a6a-1a2a3a4a5a6a.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
</tbody>
</table>

**NodeToNodeEncryptionOptions**

The NodeToNodeEncryptionOptions object provides details about the configuration for node-to-node encryption. It can have the following attributes.
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>No</td>
<td>Whether node-to-node encryption is enabled. Type: Boolean</td>
</tr>
</tbody>
</table>

#### VpcOptions

The `VpcOptions` object contains information that Amazon ES derives based on the `VPCOptions` for the domain.

It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AvailabilityZones</td>
<td>No</td>
<td>The list of Availability Zones associated with the VPC subnets. Type: array of strings</td>
</tr>
<tr>
<td>SecurityGroupIds</td>
<td>No</td>
<td>The list of security group IDs associated with the VPC endpoints for the domain. Type: array of strings</td>
</tr>
<tr>
<td>SubnetIds</td>
<td>No</td>
<td>A list of subnet IDs associated with the VPC endpoints for the domain. Type: array of strings</td>
</tr>
<tr>
<td>VPCId</td>
<td>No</td>
<td>ID for the VPC. Type: string</td>
</tr>
</tbody>
</table>

#### AwsElbv2LoadBalancer

The `AwsElbv2LoadBalancer` object provides information about a load balancer.

It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AvailabilityZones (p. 98)</td>
<td>No</td>
<td>The Availability Zones for the load balancer. Type: object</td>
</tr>
<tr>
<td>CanonicalHostedZoneId</td>
<td>No</td>
<td>The ID of the Amazon Route 53 hosted zone associated with the load balancer. Type: string</td>
</tr>
<tr>
<td>CreatedTime</td>
<td>No</td>
<td>The date and time the load balancer was created. Type: string</td>
</tr>
<tr>
<td>DNSName</td>
<td>No</td>
<td>The public DNS name of the load balancer. Type: string</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IpAddressType</td>
<td>No</td>
<td>The type of IP addresses used by the subnets for your load balancer. The possible values are <code>ipv4</code> (for IPv4 addresses) and <code>dualstack</code> (for IPv4 and IPv6 addresses). Type: string</td>
</tr>
<tr>
<td>Scheme</td>
<td>No</td>
<td>The nodes of an Internet-facing load balancer have public IP addresses. Type: string</td>
</tr>
<tr>
<td>SecurityGroups</td>
<td>No</td>
<td>The IDs of the security groups for the load balancer. Type: array of strings</td>
</tr>
<tr>
<td>State (p. 98)</td>
<td>No</td>
<td>The state of the load balancer. Type: object</td>
</tr>
<tr>
<td>Type</td>
<td>No</td>
<td>The type of load balancer. Type: string</td>
</tr>
<tr>
<td>VpcId</td>
<td>No</td>
<td>The ID of the VPC for the load balancer. Type: string</td>
</tr>
</tbody>
</table>

**AvailabilityZones**

Specifies the availability zones for the load balancer.

Each availability zone can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SubnetId</td>
<td>No</td>
<td>The ID of the subnet. Type: string</td>
</tr>
<tr>
<td>ZoneName</td>
<td>No</td>
<td>The name of the Availability Zone. Type: string</td>
</tr>
</tbody>
</table>

**State**

Information about the state of the load balancer.

The State object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>No</td>
<td>The state code. The initial state of the load balancer is provisioning.</td>
</tr>
</tbody>
</table>
After the load balancer is fully set up and ready to route traffic, its state is active. If the load balancer could not be set up, its state is failed. Type: string

Reason
No
A description of the state. Type: string

**AwsIamAccessKey**

IAM access key details that are related to a finding.

Type: object

The **AwsIamAccessKey** object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CreatedAt</td>
<td>No</td>
<td>The creation date and time of the IAM access key that is related to a finding. Type: timestamp</td>
</tr>
<tr>
<td>PrincipalId</td>
<td>No</td>
<td>The ID of the principal associated with an access key. Type: string</td>
</tr>
<tr>
<td>PrincipalName</td>
<td>No</td>
<td>The name of the principal. Type: string</td>
</tr>
<tr>
<td>PrincipalType</td>
<td>No</td>
<td>The type of principal. Type: string</td>
</tr>
<tr>
<td>Status</td>
<td>No</td>
<td>The status of the IAM access key that is related to a finding. Valid values are ACTIVE and INACTIVE. Type: enum</td>
</tr>
</tbody>
</table>

**AwsIamRole**

Contains information about an IAM role, including all of the role's policies.

Type: object

The **AwsIamRole** object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AssumeRolePolicyDocument</td>
<td>No</td>
<td>The trust policy that grants permission to assume the role.</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>CreateDate</td>
<td>No</td>
<td>The date and time, in ISO 8601 date-time format, when the role was created.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>RoleId</td>
<td>No</td>
<td>The stable and unique string identifying the role.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>RoleName</td>
<td>No</td>
<td>The friendly name that identifies the role.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>MaxSessionDuration</td>
<td>No</td>
<td>The maximum session duration (in seconds) that you want to set for the specified role.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: integer</td>
</tr>
<tr>
<td>Path</td>
<td>No</td>
<td>The path to the role.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
</tbody>
</table>

### AwsKmsKey

Details about a AWS KMS customer master key (CMK).

Type: object

The `AwsKmsKey` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWSAccountId</td>
<td>No</td>
<td>The AWS account identifier of the account that owns the CMK.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>CreationDate</td>
<td>No</td>
<td>The date and time when the CMK was created.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: timestamp</td>
</tr>
<tr>
<td>KeyId</td>
<td>Yes</td>
<td>The globally unique identifier for the CMK.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The minimum length is 1. The maximum length is 2048.</td>
</tr>
<tr>
<td>KeyManager</td>
<td>No</td>
<td>The manager of the CMK. CMKs in an AWS account are either customer managed or AWS managed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: AWS</td>
</tr>
<tr>
<td>KeyState</td>
<td>No</td>
<td>The state of the CMK.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Valid values:</strong> Enabled</td>
</tr>
<tr>
<td>Origin</td>
<td>No</td>
<td>The source of the CMK's key material.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When this value is <code>AWS_KMS</code>, AWS KMS created the key material.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When this value is <code>EXTERNAL</code>, either the key material was imported from your existing key management infrastructure, or the CMK lacks key material.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When this value is <code>AWS_CLOUDHSM</code>, the key material was created in the AWS CloudHSM cluster associated with a custom key store.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: <code>AWS_KMS</code></td>
</tr>
</tbody>
</table>

**AwsLambdaFunction**

The `AwsLambdaFunction` object provides details about a Lambda function’s configuration. It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Code (p. 102)</strong></td>
<td>No</td>
<td>An <code>AwsLambdaFunctionCode</code> object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td><strong>CodeSha256</strong></td>
<td>No</td>
<td>The SHA256 hash of the function’s deployment package.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td><strong>DeadLetterConfig (p. 103)</strong></td>
<td>No</td>
<td>The function’s dead letter queue.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td><strong>Environment (p. 103)</strong></td>
<td>No</td>
<td>A function’s environment variable settings.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td><strong>FunctionName</strong></td>
<td>No</td>
<td>The name of the function.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td><strong>Handler</strong></td>
<td>No</td>
<td>The function that Lambda calls to begin executing your function.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td><strong>KmsKeyArn</strong></td>
<td>No</td>
<td>The AWS KMS key that’s used to encrypt the function’s environment variables.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This key is only returned if you’ve configured a customer managed CMK.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td><strong>LastModified</strong></td>
<td>No</td>
<td>The date and time that the function was last updated, in ISO-8601 format (YYYY-MM-DDThh:mm:ssZ).</td>
</tr>
</tbody>
</table>
### Attribute

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Layers (p. 104)</td>
<td>No</td>
<td>The function's layers.</td>
</tr>
<tr>
<td>Type: object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MasterArn</td>
<td>No</td>
<td>For Lambda@Edge functions, the ARN of the master function.</td>
</tr>
<tr>
<td>Type: string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MemorySize</td>
<td>No</td>
<td>The memory that's allocated to the function.</td>
</tr>
<tr>
<td>Type: integer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RevisionId</td>
<td>No</td>
<td>The latest updated revision of the function or alias.</td>
</tr>
<tr>
<td>Type: string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Role</td>
<td>No</td>
<td>The function's execution role.</td>
</tr>
<tr>
<td>Type: string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Runtime</td>
<td>No</td>
<td>The runtime environment for the Lambda function.</td>
</tr>
<tr>
<td>Type: string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Timeout</td>
<td>No</td>
<td>The amount of time that Lambda allows a function to run before stopping it.</td>
</tr>
<tr>
<td>Type: integer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TracingConfig (p. 104)</td>
<td>No</td>
<td>The function's AWS X-Ray tracing configuration.</td>
</tr>
<tr>
<td>Type: object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Version</td>
<td>No</td>
<td>The version of the Lambda function.</td>
</tr>
<tr>
<td>Type: string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VpcConfig (p. 104)</td>
<td>No</td>
<td>The function's networking configuration.</td>
</tr>
<tr>
<td>Type: object</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Code

An AwsLambdaFunctionCode object.

The Code object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S3Bucket</td>
<td>No</td>
<td>An S3 bucket in the same AWS Region as your function. The bucket can be in a different AWS account.</td>
</tr>
<tr>
<td>Type: string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S3Key</td>
<td>No</td>
<td>The Amazon S3 key of the deployment package.</td>
</tr>
<tr>
<td>Type: string</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>S3ObjectVersion</td>
<td>No</td>
<td>For versioned objects, the version of the deployment package object to use.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>ZipFile</td>
<td>No</td>
<td>The base64-encoded contents of the deployment package. AWS SDK and AWS CLI clients handle the encoding for you.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
</tbody>
</table>

**DeadLetterConfig**

Contains information about the Lambda function's dead letter queue.

The `DeadLetterConfig` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TargetArn</td>
<td>No</td>
<td>The Amazon Resource Name (ARN) of the Amazon SQS queue or Amazon SNS topic containing the dead letter queue.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
</tbody>
</table>

**Environment**

Contains the Lambda function's environment variable settings.

The `Environment` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variables</td>
<td>No</td>
<td>Environment variable key-value pairs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string to string map</td>
</tr>
<tr>
<td>Error</td>
<td>No</td>
<td>Error messages for environment variables that couldn't be applied.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
</tbody>
</table>

The `Error` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ErrorCode</td>
<td>No</td>
<td>The error code.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>Message</td>
<td>No</td>
<td>The error message.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
</tbody>
</table>
Layers
The Lambda function's layers.
Each layer object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arn</td>
<td>No</td>
<td>The Amazon Resource Name (ARN) of the function layer. Type: string</td>
</tr>
<tr>
<td>CodeSize</td>
<td>No</td>
<td>The size of the layer archive in bytes. Type: integer</td>
</tr>
</tbody>
</table>

TracingConfig
Contains the function's AWS X-Ray tracing configuration.
The TracingConfig object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>No</td>
<td>The tracing mode. Type: string</td>
</tr>
</tbody>
</table>

VpcConfig
Contains the Lambda function's networking configuration.
The VpcConfig object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SecurityGroupIds</td>
<td>No</td>
<td>A list of VPC security groups IDs. Type: array of strings</td>
</tr>
<tr>
<td>SubnetIds</td>
<td>No</td>
<td>A list of VPC subnet IDs. Type: array of strings</td>
</tr>
</tbody>
</table>

AwsLambdaLayerVersion
The AwsLambdaLayerVersion object provides details about a Lambda layer version.
Example:

```json
"AwsLambdaLayerVersion": { 
  "Version": 2,
  "CompatibleRuntimes": [ 
    "java8"
  ],
  "CreatedAt": "2019-10-09T22:02:00.274+0000"
}
```
The `AwsLambdaLayerVersion` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CompatibleRuntimes</code></td>
<td>No</td>
<td>The layer's compatible runtimes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: array of strings</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maximum number of items: 5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: <code>nodejs10.x</code></td>
</tr>
<tr>
<td><code>CreatedDate</code></td>
<td>No</td>
<td>The date that the version was created, in ISO 8601 format.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For example, 2018-11-27T15:45.123+0000.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td><code>Version</code></td>
<td>No</td>
<td>The version number.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: long</td>
</tr>
</tbody>
</table>

**AwsRdsDbInstance**

The `AwsRdsDbInstance` object provides details about an Amazon RDS DB instance.

It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AssociatedRoles</code></td>
<td>No</td>
<td>The IAM roles associated with the DB instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: array of role objects</td>
</tr>
<tr>
<td><code>CACertificateIdentifier</code></td>
<td>No</td>
<td>The identifier of the CA certificate for this DB instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td><code>DBClusterIdentifier</code></td>
<td>No</td>
<td>If the DB instance is a member of a DB cluster, contains the name of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DB cluster that the DB instance is a member of.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td><code>DBInstanceClass</code></td>
<td>No</td>
<td>Contains the name of the compute and memory capacity class of the DB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td><code>DBInstanceIdentifier</code></td>
<td>No</td>
<td>Contains a user-supplied database identifier. This identifier is the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>unique key that identifies a DB instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td><code>DbInstancePort</code></td>
<td>No</td>
<td>Specifies the port that the DB instance listens on. If the DB instance is</td>
</tr>
<tr>
<td></td>
<td></td>
<td>part of a DB cluster, this can be a different port than the DB cluster port.</td>
</tr>
</tbody>
</table>

105
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DbiResourceId</td>
<td>No</td>
<td>The AWS Region-unique, immutable identifier for the DB instance. This identifier is found in CloudTrail log entries whenever the AWS KMS key for the DB instance is accessed.</td>
</tr>
<tr>
<td>DBName</td>
<td>No</td>
<td>The meaning of this parameter differs according to the database engine you use. MySQL, MariaDB, SQL Server, PostgreSQL Contains the name of the initial database of this instance that was provided at create time, if one was specified when the DB instance was created. This same name is returned for the life of the DB instance.</td>
</tr>
<tr>
<td>DeletionProtection</td>
<td>No</td>
<td>Indicates whether the DB instance has deletion protection enabled. The database can't be deleted when deletion protection is enabled.</td>
</tr>
<tr>
<td>Endpoint</td>
<td>No</td>
<td>Specifies the connection endpoint.</td>
</tr>
<tr>
<td>Engine</td>
<td>No</td>
<td>Provides the name of the database engine to be used for this DB instance.</td>
</tr>
<tr>
<td>EngineVersion</td>
<td>No</td>
<td>Indicates the database engine version.</td>
</tr>
</tbody>
</table>
## ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IAMDatabaseAuthenticationEnabled</td>
<td>No</td>
<td>True if mapping of IAM accounts to database accounts is enabled, and otherwise false. IAM database authentication can be enabled for the following database engines.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For MySQL 5.6, minor version 5.6.34 or higher.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For MySQL 5.7, minor version 5.7.16 or higher.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Aurora 5.6 or higher.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: Boolean</td>
</tr>
<tr>
<td>InstanceCreateTime</td>
<td>No</td>
<td>Provides the date and time the DB instance was created.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: timestamp</td>
</tr>
<tr>
<td>KmsKeyId</td>
<td>No</td>
<td>If StorageEncrypted is true, the AWS KMS key identifier for the encrypted DB instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>PubliclyAccessible</td>
<td>No</td>
<td>Specifies the accessibility options for the DB instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A value of true specifies an Internet-facing instance with a publicly resolvable DNS name, which resolves to a public IP address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A value of false specifies an internal instance with a DNS name that resolves to a private IP address.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: Boolean</td>
</tr>
<tr>
<td>StorageEncrypted</td>
<td>No</td>
<td>Specifies whether the DB instance is encrypted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: Boolean</td>
</tr>
<tr>
<td>TdeCredentialArn</td>
<td>No</td>
<td>The ARN from the key store with which the instance is associated for TDE encryption.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>VpcSecurityGroups (p. 108)</td>
<td>No</td>
<td>List of VPC security groups that the DB instance belongs to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: array of objects</td>
</tr>
</tbody>
</table>

### AssociatedRoles

The AssociatedRoles array lists the IAM roles associated with the DB instance. Each role object in the AssociatedRoles array can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FeatureName</td>
<td>No</td>
<td>The name of the feature associated with the IAM role.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>Attribute</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RoleArn</td>
<td>No</td>
<td>The ARN of the IAM role that is associated with the DB instance. Type: string</td>
</tr>
</tbody>
</table>
| Status      | No       | Describes the state of association between the IAM role and the DB instance. Type: string  
  Valid values:  
  • ACTIVE - The IAM role ARN is associated with the DB instance and can be used to access other AWS services on your behalf.  
  • PENDING - The IAM role ARN is being associated with the DB instance.  
  • INVALID - The IAM role ARN is associated with the DB instance, but the DB instance is unable to assume the IAM role in order to access other AWS services on your behalf. |

**Endpoint**

The **Endpoint** object provides details about the connection endpoint for the DB instance. It can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>No</td>
<td>Specifies the DNS address of the DB instance. Type: string</td>
</tr>
<tr>
<td>HostedZoneId</td>
<td>No</td>
<td>Specifies the ID that Amazon Route 53 assigns when you create a hosted zone. Type: string</td>
</tr>
<tr>
<td>Port</td>
<td>No</td>
<td>Specifies the port that the database engine is listening on. Type: integer</td>
</tr>
</tbody>
</table>

**VpcSecurityGroups**

The **VpcSecurityGroups** array provides the list of VPC security groups that the DB instance belongs to. Each object in the **VpcSecurityGroups** array can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VpcSecurityGroupId</td>
<td>No</td>
<td>The name of the VPC security group. Type: string</td>
</tr>
<tr>
<td>Status</td>
<td>No</td>
<td>The status of the VPC security group. Type: string</td>
</tr>
</tbody>
</table>
**AwsS3Bucket**

The details of an Amazon S3 bucket.

Type: object

The `AwsS3Bucket` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CreatedAt</td>
<td>No</td>
<td>The date and time when the S3 bucket was created. Type: string (Uses the RFC 3339 format)</td>
</tr>
<tr>
<td>OwnerId</td>
<td>No</td>
<td>The canonical user ID of the owner of the Amazon S3 bucket. Type: string (64 char max)</td>
</tr>
<tr>
<td>OwnerName</td>
<td>No</td>
<td>The display name of the owner of the Amazon S3 bucket. Type: string (128 char max)</td>
</tr>
<tr>
<td>ServerSideEncryptionConfig</td>
<td></td>
<td>The encryption rules that are applied to the S3 bucket. Type: object Consists of a <code>Rules</code> object, which contains the <code>ApplyServerSideEncryptionByDefault</code> object. Example:</td>
</tr>
</tbody>
</table>

```json
"ServerSideEncryptionConfiguration": {
"Rules": [
  {
  "ApplyServerSideEncryptionByDefault": {
    "KMSMasterKeyID": "12345678-abcd-abcd-abcd-123456789012",
    "SSEAlgorithm": "aws.kms"
  }
  }
  }
}
```

**ApplyServerSideEncryptionByDefault**

Specifies the default server-side encryption to apply to new objects in the bucket.

`ApplyServerSideEncryptionByDefault` can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KMSMasterKeyId</td>
<td>No</td>
<td>AWS KMS customer master key (CMK) ID to use for the default encryption. Can be either the key ID or the CMK ARN. Type: string</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSEAAlgorithm</td>
<td>Yes</td>
<td>Server-side encryption algorithm to use for the default encryption.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: AES256</td>
</tr>
</tbody>
</table>

### AWS3Object

Details about an AWS S3 object.

Example:

```json
"AwsS3Object": {
  "ContentType": "text/html",
  "ETag": "\"30a6ec7e1a9ad79c203d05a589c8b400\"",
  "LastModified": "2012-04-23T18:25:43.511Z",
  "ServerSideEncryption": "aws:kms",
  "SSEKMSKeyId": "arn:aws:kms:us-west-2:123456789012:key/4dff8393-e225-4793-a9a0-608ec069e5a7",
  "VersionId": "ws31OurygO0jH_HHllIzPE35P.MELYaYh"
}
```

**AWS3Object** can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ContentType</td>
<td>No</td>
<td>A standard MIME type describing the format of the object data.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>ETag</td>
<td>No</td>
<td>The opaque identifier assigned by a web server to a specific version of a resource found at a URL.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>LastModified</td>
<td>No</td>
<td>The date and time when the object was last modified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: datetime</td>
</tr>
<tr>
<td>ServerSideEncryption</td>
<td>No</td>
<td>If the object is stored using server-side encryption, the value of the server-side encryption algorithm used when storing this object in Amazon S3.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>SSEKMSKeyId</td>
<td>No</td>
<td>The identifier of the AWS Key Management Service) symmetric customer managed customer master key (CMK) that was used for the object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
</tbody>
</table>
### ASFF attributes

**Attribute** | **Required** | **Description**
--- | --- | ---
VersionId | No | The version of the object.
| | | Type: string

**AwsSnsTopic**

A wrapper type for the topic's Amazon Resource Name (ARN).

Type: object

The `AwsSnsTopic` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KmsMasterKeyId</td>
<td>No</td>
<td>The ID of an AWS-managed customer master key (CMK) for Amazon SNS or a custom CMK.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>Subscription</td>
<td>No</td>
<td>Subscription is an embedded property that describes the subscription endpoints of an Amazon SNS topic.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: array of objects</td>
</tr>
<tr>
<td>TopicName</td>
<td>No</td>
<td>The name of the topic.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>Owner</td>
<td>No</td>
<td>The subscription's owner.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
</tbody>
</table>

**Subscription**

The `Subscription` object in an `AwsSnsTopic` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoint</td>
<td>No</td>
<td>The subscription's endpoint (format depends on the protocol).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td>Protocol</td>
<td>No</td>
<td>The subscription's protocol.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
</tbody>
</table>

**AwsSqsQueue**

Data about an Amazon SQS queue.

Type: object

The `AwsSqsQueue` object can have the following attributes.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KmsDataKeyReusePeriodSeconds</td>
<td>No</td>
<td>The length of time, in seconds, for which Amazon SQS can reuse a data key to encrypt or decrypt messages before calling AWS KMS again. Type: integer</td>
</tr>
<tr>
<td>KmsMasterKeyId</td>
<td>No</td>
<td>The ID of an AWS-managed customer master key (CMK) for Amazon SQS or a custom CMK. Type: string</td>
</tr>
<tr>
<td>QueueName</td>
<td>No</td>
<td>The name of the new queue. Type: string</td>
</tr>
<tr>
<td>DeadLetterTargetArn</td>
<td>No</td>
<td>The Amazon Resource Name (ARN) of the dead-letter queue to which Amazon SQS moves messages after the value of maxReceiveCount is exceeded. Type: string</td>
</tr>
</tbody>
</table>

**AwsWafWebAcl**

The `AwsWafWebAcl` object provides details about an AWS WAF WebACL.

Example:

```json
"AwsWafWebAcl": {
    "DefaultAction": "ALLOW",
    "Name": "MyWafAcl",
    "Rules": [
      {
        "Action": {
          "Type": "ALLOW"
        },
        "ExcludedRules": [
          {
            "RuleId": "5432a230-0113-5b83-bbb2-89375c5bfa98"
          }
        ],
        "OverrideAction": {
          "Type": "NONE"
        },
        "Priority": 1,
        "RuleId": "5432a230-0113-5b83-bbb2-89375c5bfa98",
        "Type": "REGULAR"
      }
    ],
    "WebAclId": "waf-1234567890"
}
```

An `AwsWafWebAcl` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DefaultAction</td>
<td>Yes</td>
<td>The action to perform if none of the rules contained in the WebACL match. The action is specified by the WafAction object.</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: WafAction object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>No</td>
<td>A friendly name or description of the WebACL. You can't change the name of a WebACL after you create it.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Length constraints: Minimum length of 1. Maximum length of 128.</td>
</tr>
<tr>
<td>Rules (p. 113)</td>
<td>Yes</td>
<td>A list of rules for the WebACL.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: array of objects</td>
</tr>
<tr>
<td>WebAclId</td>
<td>Yes</td>
<td>A unique identifier for a WebACL.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Length constraints: Minimum length of 1. Maximum length of 128.</td>
</tr>
</tbody>
</table>

**Rule object**

The **Rules array** provides the list of rules for the WebACL.

Each rule object in the array can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>No</td>
<td>Specifies the action that CloudFront or AWS WAF takes when a web request matches the conditions in the rule.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
<tr>
<td>ExcludedRules</td>
<td>No</td>
<td>An array of rules to exclude from a rule group. This is applicable only when the ActivatedRule refers to a rule group.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: array of objects</td>
</tr>
<tr>
<td>OverrideAction</td>
<td>No</td>
<td>Use the OverrideAction to test your RuleGroup.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Any rule in a rule group can potentially block a request. If you set the OverrideAction to None, the rule group blocks a request if any individual rule in the rule group matches the request and is configured to block that request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>However, if you first want to test the rule group, set the OverrideAction to Count. The rule group then overrides any block action specified by individual rules contained within the group. Instead of blocking matching requests, those requests are counted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ActivatedRule</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: object</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>Yes</td>
<td>Specifies the order in which to evaluate the rules in a WebACL. Rules with a lower value for Priority are evaluated before rules with a higher value. The value must be a unique integer. If you add multiple rules to a WebACL, the values don't need to be consecutive. Type: integer</td>
</tr>
<tr>
<td>RuleId</td>
<td>Yes</td>
<td>The identifier for a rule. Type: String Length constraints: Minimum length of 1. Maximum length of 128.</td>
</tr>
<tr>
<td>Type</td>
<td>No</td>
<td>The rule type, either REGULAR, RATE_BASED, or GROUP. The default is REGULAR. Type: string Valid values: REGULAR</td>
</tr>
</tbody>
</table>

Each action can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
</table>
| Type      | No       | Specifies how you want AWS WAF to respond to requests that match the settings in a rule.  
- **ALLOW** - AWS WAF allows requests  
- **BLOCK** - AWS WAF blocks requests  
- **COUNT** - AWS WAF increments a counter of the requests that match all of the conditions in the rule. AWS WAF then continues to inspect the web request based on the remaining rules in the web ACL. You can't specify COUNT for the default action for a WebACL. Type: string Valid values: BLOCK | ALLOW | COUNT |

Each excluded rule can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RuleId</td>
<td>Yes</td>
<td>The unique identifier for the rule to exclude from the rule group. Type: string Length constraints: Minimum length of 1. Maximum length of 128.</td>
</tr>
</tbody>
</table>
Each override action for a rule can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Yes</td>
<td>COUNT overrides the action specified by the individual rule within a rule group. If set to NONE, the rule's action takes place. Type: string Valid values: NONE</td>
</tr>
</tbody>
</table>

**Container**

Container details that are related to a finding.

Type: object

Example:

```
"Container": {
  "Name": "Secret Service Container",
  "ImageId": "image12",
  "ImageName": "SecSvc v1.2 Image",
  "LaunchedAt": "2018-09-29T01:25:54Z"
}
```

The `Container` object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageId</td>
<td>No</td>
<td>The identifier of the image that is related to a finding. Type: string (128 characters max)</td>
</tr>
<tr>
<td>ImageName</td>
<td>No</td>
<td>The name of the image that is related to a finding. Type: string (128 characters max)</td>
</tr>
<tr>
<td>LaunchedAt</td>
<td>No</td>
<td>The date and time that the container was started. Type: timestamp</td>
</tr>
<tr>
<td>Name</td>
<td>No</td>
<td>The name of the container that is related to a finding. Type: string (128 characters max)</td>
</tr>
</tbody>
</table>

**Other**

The `Other` subfield allows you to provide custom fields and values. You use the `Other` subfield in the following cases:

- The resource type does not have a corresponding subfield. To provide details for the resource, you use the `Other` subfield.
• The subfield for the resource type does not include all of the fields you want to populate. In this case, use the subfield for the resource type to populate the available fields. Use the Other subfield to populate the fields that are not in the type-specific subfield.
• The resource type is not one of the provided types. In this case, you set Resource.Type to Other, and use the Other subfield to populate the details.

**Type:** map of up to 50 key-value pairs

Each key-value pair must meet the following requirements.

• The key must contain fewer than 128 characters.
• The value must contain fewer than 1,024 characters.

**Severity**

Details about the severity of the finding.

The finding provider can set the initial severity, but cannot update it after that. The severity can only be updated using `BatchUpdateFindings`. It can only be updated by a master account. It cannot be updated by a member account.

The finding severity does not consider the criticality of the involved assets or the underlying resource. Criticality is defined as the level of importance of the resources associated with the finding. For example, a resource associated with a mission critical application versus one associated with non-production testing. To capture information about resource criticality, use the `Criticality` field.

The finding must have either `Label` or `Normalized` populated. If neither attribute is populated, then the finding is invalid. `Label` is the preferred attribute.

The **Severity** object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Label</strong></td>
<td>No</td>
<td>The severity value for the finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: enum</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available values: INFORMATIONAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>At a high level, the <code>Label</code> values can be interpreted as follows.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• INFORMATIONAL – No issue was found.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• LOW – The issue does not require action on its own.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MEDIUM – The issue must be addressed but not urgently.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• HIGH – The issue must be addressed as a priority.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• CRITICAL – The issue must be remediated immediately to avoid it escalating.</td>
</tr>
<tr>
<td><strong>Normalized (To be deprecated)</strong></td>
<td>No</td>
<td>The normalized severity of a finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We plan to deprecate this attribute. Instead of <code>Normalized</code>, provide <code>Label</code>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: integer</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Normalized</strong></td>
<td>No</td>
<td>The value of Normalized must be an integer between 0 and 100. Zero means that no severity applies, and 100 means that the finding has the maximum possible severity. For guidance on setting the value of Normalized, see the section called “Guidance for Assigning the Normalized Severity (AWS Services and Partners)” (p. 117).</td>
</tr>
<tr>
<td><strong>Product</strong></td>
<td>No</td>
<td>The native severity as defined by the finding product that generated the finding. Type: number (single-precision 32-bit IEEE 754 floating point number, restricted to finite values)</td>
</tr>
</tbody>
</table>

### How Security Hub maps Label and Normalized values

If you only provide Label or only provide Normalized, then Security Hub automatically populates the value of the other field.

If you provide Normalized and do not provide Label, Label is set automatically as follows.

<table>
<thead>
<tr>
<th>Normalized</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>INFORMATIONAL</td>
</tr>
<tr>
<td>1–39</td>
<td>LOW</td>
</tr>
<tr>
<td>40–69</td>
<td>MEDIUM</td>
</tr>
<tr>
<td>70–89</td>
<td>HIGH</td>
</tr>
<tr>
<td>90–100</td>
<td>CRITICAL</td>
</tr>
</tbody>
</table>

If you only provide Label and do not provide Normalized, Normalized is set automatically as follows.

<table>
<thead>
<tr>
<th>Label</th>
<th>Normalized</th>
</tr>
</thead>
<tbody>
<tr>
<td>INFORMATIONAL</td>
<td>0</td>
</tr>
<tr>
<td>LOW</td>
<td>1</td>
</tr>
<tr>
<td>MEDIUM</td>
<td>40</td>
</tr>
<tr>
<td>HIGH</td>
<td>70</td>
</tr>
<tr>
<td>CRITICAL</td>
<td>90</td>
</tr>
</tbody>
</table>

### Guidance for Assigning the Normalized Severity (AWS Services and Partners)

For findings generated by AWS services and third-party partner products, the severity is based on the following.

- **Most severe** – Findings that are associated with actual data loss or denial of service
• Second-most severe – Findings that are associated with an active compromise but that do not indicate that data loss or other negative effects have occurred
• Third-most severe – Findings associated with issues that indicate potential for a future compromise

We recommend that you use the following guidance when translating findings' native severity scores to a normalized severity for the ASFF:

• Informational findings. For example, a finding for a passed check or a sensitive data identification.
  Suggested score: 0
• Findings that are associated with issues that could result in future compromises. For example, vulnerabilities, configuration weaknesses, exposed passwords.
  This generally aligns to the Software and Configuration Checks namespace under a finding's type.
  Suggested score: 1–39 (Low)
• Findings that are associated with issues that indicate an active compromise, but no indication that an adversary completed their objectives. For example, malware activity, hacking activity, or unusual behavior detection.
  This generally aligns to the Threat Detections and Unusual Behavior namespaces under a finding's type.
  Suggested score: 40–69 (Medium)
• Findings that indicate that an adversary completed their objectives, such as active data loss or compromise or a denial of service.
  This generally aligns to the Effects namespace under a finding's type.
  Suggested score: 70–100 (High or Critical)

**ThreatIntelIndicators**

Threat intelligence details that are related to a finding.

Type: array of up to five threat intelligence indicator objects

Example:

```
"ThreatIntelIndicators": [ 
  { 
    "Type": "IPV4_ADDRESS", 
    "Value": "8.8.8.8", 
    "Category": "BACKDOOR", 
    "LastObservedAt": "2018-09-27T23:37:31Z", 
    "Source": "Threat Intel Weekly", 
    "SourceUrl": "http://threatintelweekly.org/backdoors/8888" 
  } 
]
```

Each threat intelligence indicator object can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>No</td>
<td>The category of a threat intel indicator.</td>
</tr>
</tbody>
</table>
### ASFF attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Attribute</strong></td>
<td><strong>Required</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: enum</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: BACKDOOR</td>
</tr>
<tr>
<td>LastObservedAt</td>
<td>No</td>
<td>The date and time of the last observation of a threat intelligence indicator.</td>
</tr>
<tr>
<td>Source</td>
<td>No</td>
<td>The source of the threat intelligence.</td>
</tr>
<tr>
<td>SourceUrl</td>
<td>No</td>
<td>The URL for more details from the source of the threat intelligence.</td>
</tr>
<tr>
<td>Type</td>
<td>No</td>
<td>The type of a threat intelligence indicator.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: enum</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: DOMAIN</td>
</tr>
<tr>
<td>Value</td>
<td>No</td>
<td>The value of a threat intelligence indicator.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: string (512 characters max)</td>
</tr>
</tbody>
</table>

### Workflow

The **Workflow** object provides information about the status of the investigation into a finding.

It is not intended for finding providers. It is only to be used by customers and by remediation, orchestration, and ticketing tools used by customers.

The workflow status can only be updated using **BatchUpdateFindings**. Customers can also update it from the console. See the section called “Setting the workflow status for a finding” (p. 45). The workflow status can only be updated by a master account. It cannot be updated by a member account.

**Workflow** can have the following attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>No</td>
<td>The status of the investigation into the finding.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type: enum</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Valid values: NEW</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NEW – The initial state of a finding, before it is reviewed.</td>
</tr>
</tbody>
</table>


Types taxonomy for ASFF

The following information describes the first three levels of the Types path. In the list, the top-level bullets are namespaces, the second-level bullets are categories, and the third-level bullets are classifiers. Only the Software and Configuration Checks namespace has defined classifiers.

- Namespaces
  - Categories
    - Classifiers

Finding providers must use the defined namespaces. The defined categories and classifiers are recommended for use, but are not required.

A finding provider might define a partial path for namespace/category/classifier. For example, the following finding types are all valid.

- TTPs
- TTPs/Defense Evasion
- TTPs/Defense Evasion/CloudTrailStopped

TTPs stands for Tactics, Techniques, and Procedures. The TTP categories in the following list align to the MITRE ATT&CK Matrix™. Unusual Behaviors reflect general unusual behavior, such as general statistical anomalies, and are not aligned with a specific TTP. However, you could classify a finding with both Unusual Behaviors and TTPs finding types.

- Software and Configuration Checks
  - Vulnerabilities
    - CVE
  - AWS Security Best Practices
    - Network Reachability
    - Runtime Behavior Analysis
  - Industry and Regulatory Standards
    - CIS Host Hardening Benchmarks
    - CIS AWS Foundations Benchmark
    - PCI-DSS Controls
    - Cloud Security Alliance Controls
    - ISO 90001 Controls

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOTIFIED</td>
<td>•</td>
<td>Indicates that you notified the resource owner about the security issue. Used when the initial reviewer is not the resource owner, and needs intervention from the resource owner.</td>
</tr>
<tr>
<td>RESOLVED</td>
<td>•</td>
<td>The finding was reviewed and remediated and is now considered resolved.</td>
</tr>
<tr>
<td>SUPPRESSED</td>
<td>•</td>
<td>The finding will not be reviewed again and will not be acted upon.</td>
</tr>
</tbody>
</table>
- ISO 27001 Controls
- ISO 27017 Controls
- ISO 27018 Controls
- SOC 1
- SOC 2
- HIPAA Controls (USA)
- NIST 800-53 Controls (USA)
- NIST CSF Controls (USA)
- IRAP Controls (Australia)
- K-ISMS Controls (Korea)
- MTCS Controls (Singapore)
- FISC Controls (Japan)
- My Number Act Controls (Japan)
- ENS Controls (Spain)
- Cyber Essentials Plus Controls (UK)
- G-Cloud Controls (UK)
- CS Controls (Germany)
- IT-Grundschutz Controls (Germany)
- GDPR Controls (Europe)
- TISAX Controls (Europe)
- TTPs
  - Initial Access
  - Execution
  - Persistence
  - Privilege Escalation
  - Defense Evasion
  - Credential Access
  - Discovery
  - Lateral Movement
  - Collection
  - Command and Control
- Effects
  - Data Exposure
  - Data Exfiltration
  - Data Destruction
  - Denial of Service
  - Resource Consumption
- Unusual Behaviors
  - Application
  - Network Flow
  - IP address
  - User
- VM
- Container
- Serverless
• Process
• Database
• Data
• Sensitive Data Identifications
  • PII
  • Passwords
  • Legal
  • Financial
  • Security
  • Business
Product integrations in AWS Security Hub

AWS Security Hub provides available integrations with several AWS services and third-party products. You can also send findings that are generated from your own custom security products.

**Important**
Security Hub receives and consolidates only those security findings from the supported AWS and partner product integrations that are generated after Security Hub is enabled in your AWS accounts.
It does not retroactively receive and consolidate security findings that were generated before you enabled Security Hub.

For details on how Security Hub charges for ingested findings, see [Security Hub pricing](#).

**Topics**
- Managing product integrations (p. 123)
- Available AWS service integrations (p. 125)
- Available third-party partner product integrations (p. 126)
- Using custom product integrations to send findings to AWS Security Hub (p. 131)

Managing product integrations

The **Integrations** page provides access to all of the available AWS and third-party product integrations. The AWS Security Hub API also provides operations to allow you to manage integrations.

**Note**
Some integrations are not available in AWS GovCloud (US-East) or AWS GovCloud (US-West). If an integration is not supported, it is not listed on the **Integrations** page.

**Viewing and filtering the list of integrations**

From the **Integrations** page, you can view and filter the list of integrations.

**To view the list of integrations**

2. In the Security Hub navigation pane, choose **Integrations**.

On the **Integrations** page, the integrations with other AWS services are listed first, followed by the integrations with third-party products.

For each integration, the **Integrations** page provides the following information.

- The name of the company
- The name of the product
- A description of the integration
- The categories that the integration applies to
Enabling an integration

On the Integrations page, each integration provides the required steps to enable the integration.

For most of the integrations with other AWS services, the only required step is to enable the other service. The integration information includes a link to the service home page. When you enable the other service, a resource-level permission that allows Security Hub to receive findings from the service is then automatically created and applied.

For third-party product integrations, you may need to purchase the integration from the AWS Marketplace, and then configure the integration. The integration information provides links to perform those tasks.

If more than one version of a product is available in AWS Marketplace, select the version to subscribe to and then choose Continue to Subscribe. For example, some products offer a standard version and an AWS GovCloud (US) version.

When you enable a product integration, a resource policy is automatically attached to that product subscription. This resource policy defines the permissions that Security Hub needs to receive findings from that product.

Disabling and enabling the flow of findings from an integration (Console)

On the Integrations page, for integrations that send findings, the Status field indicates whether you are currently accepting findings.

To stop accepting findings, choose Stop accepting findings.

To resume accepting findings, choose Accept findings.

Disabling and enabling the flow of findings from an integration (API)

To use the API to stop receiving findings from an integration, use the DisableImportFindingsForProduct operation. To disable the flow of findings from the integration, you need the ARN for your subscription. To get the subscription ARNs for your currently enabled integrations, use the ListEnabledProductsForImport operation.

To use the API to enable receiving findings from an integration, use the EnableImportFindingsForProduct operation. To enable Security Hub to receive findings from an integration, you need the product ARN. To get the ARNs for the available integrations, use the DescribeProducts operation.
Viewing the findings from an integration

For integrations that you are accepting findings for (Status is Accepting findings), to view a list of findings, choose See findings.

The findings list shows the active findings for the selected integration that have a workflow status of NEW or NOTIFIED.

From the findings list, you can perform the following actions.

- Change the filters and grouping for the list (p. 44)
- View details for individual findings (p. 45)
- Update the workflow status of findings (p. 45)
- Send findings to custom actions (p. 46)

Available AWS service integrations

Security Hub supports integrations with several AWS services.

**Note**
Some integrations are not available in AWS GovCloud (US-East) or AWS GovCloud (US-West). If an integration is not supported, it is not listed on the Integrations page.

For these services, the integration allows the service to send findings to Security Hub.

- AWS Firewall Manager (p. 125)
- IAM Access Analyzer (p. 125)
- Amazon GuardDuty (p. 126)
- Amazon Inspector (p. 126)
- Amazon Macie (p. 126)

Security Hub also supports an integration with Amazon Detective (p. 126). That integration allows you to pivot from Security Hub to Detective to investigate a GuardDuty finding.

Here are the details about each AWS service integration.

**AWS Firewall Manager**

Firewall Manager sends findings to Security Hub when a WAF policy for resources or a Web ACL rule is not in compliance. Firewall Manager also sends findings when Shield Advanced is not protecting resources, or when an attack is identified.

If you are already using Firewall Manager, Security Hub automatically enables this integration. You do not need to take any additional action to begin to receive findings from Firewall Manager.

To learn more about the integration, view the Integrations page in the Security Hub console.

To learn more about Firewall Manager, see the AWS WAF Developer Guide.

**IAM Access Analyzer**

With IAM Access Analyzer, all findings are sent to Security Hub.

IAM Access Analyzer uses logic-based reasoning to analyze resource-based policies applied to supported resources in your account. IAM Access Analyzer generates a finding when it detects a policy statement that allows an external principal access to a resource in your account.
Available third-party partner product integrations

AWS Security Hub is integrated with the following third-party products. For each provider, the list indicates whether the product sends findings to Security Hub, receives findings from Security Hub, or both.

**Note**
Some integrations are not available in AWS GovCloud (US-East) or AWS GovCloud (US-West). If an integration is not supported, it is not listed on the Integrations page.
If applicable, the list also specifies the product ARN. Integrations that send findings to Security Hub always have an ARN.

<table>
<thead>
<tr>
<th>Company name</th>
<th>Product name</th>
<th>Integrations</th>
<th>Product ARN</th>
<th>Product description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alert Logic</td>
<td>SIEMless ThreatManagement</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:733251395267:product/alertlogic/althreatmanagement</td>
<td>Get the right level of coverage: vulnerability and asset visibility, threat detection and incident management, WAF, and assigned SOC analyst options.</td>
</tr>
<tr>
<td>Armor</td>
<td>Armor Anywhere</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/armordefense/armoranywhere</td>
<td>Armor Anywhere delivers managed security and compliance for AWS.</td>
</tr>
<tr>
<td>Atlassian</td>
<td>OpsGenie</td>
<td>Receive</td>
<td></td>
<td>Opsgenie is a modern incident management solution for operating always-on services, empowering Dev &amp; Ops teams to plan for service disruptions and stay in control during incidents. Integrating with Security Hub ensures that mission critical security related incidents are routed to the appropriate teams for immediate resolution.</td>
</tr>
<tr>
<td>Barracuda Networks</td>
<td>Cloud Security Guardian</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/barracuda/cloudsecurityguardian</td>
<td>Barracuda Cloud Security Sentry helps organizations stay secure while building applications in, and moving workloads to, the public cloud.</td>
</tr>
<tr>
<td>BigID</td>
<td>BigID Enterprise</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/bigid/bigid-enterprise</td>
<td>BigID Enterprise Privacy Management Platform</td>
</tr>
<tr>
<td>Capitis Solutions</td>
<td>C2VS</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/capitis/c2vs</td>
<td>C2VS is a customizable compliance solution designed to automatically identify your application specific misconfigurations and their root cause.</td>
</tr>
<tr>
<td>Company name</td>
<td>Product name</td>
<td>Integrations</td>
<td>Product ARN</td>
<td>Product description</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------</td>
<td>--------------</td>
<td>------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Checkpoint</td>
<td>CloudGuard IaaS</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:758245563457:product/checkpoint/cloudguard-iaas</td>
<td>Check Point CloudGuard easily extends comprehensive threat prevention security to AWS while protecting assets in the cloud.</td>
</tr>
<tr>
<td>Cloud Custodian</td>
<td>Cloud Custodian and Receive</td>
<td></td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/cloud-custodian/cloud-custodian</td>
<td>Cloud Custodian enables users to be well managed in the cloud. The simple YAML DSL allows easily define rules to enable a well-managed cloud infrastructure, that's both secure and cost optimized.</td>
</tr>
<tr>
<td>CrowdStrike</td>
<td>CrowdStrike Falcon</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/crowdstrike/crowdstrike-falcon</td>
<td>CrowdStrike Falcon's single lightweight sensor unifies next-generation antivirus, endpoint detection and response, and 24/7 managed hunting via the cloud.</td>
</tr>
<tr>
<td>CyberArk</td>
<td>Privileged Send Threat Analytics</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/cyberark/cyberark-pta</td>
<td>CyberArk Privileged Threat Analytics collect, detect, alert, and respond to high-risk activity and behavior of privileged accounts to contain in-progress attacks.</td>
</tr>
<tr>
<td>F5 Networks</td>
<td>Advanced Send WAF</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/f5networks/f5-advanced-waf</td>
<td>F5 Advanced WAF addresses malicious bot protection, L7 DoS mitigation, API inspection, behavior analytics, and more to defend against web app attacks.</td>
</tr>
<tr>
<td>FireEye</td>
<td>FireEye Helix</td>
<td>Receive</td>
<td></td>
<td>FireEye Helix is a cloud-hosted security operations platform that allows organizations to take control of any incident from alert to fix.</td>
</tr>
<tr>
<td>Forcepoint</td>
<td>Forcepoint CASB</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/forcepoint/forcepoint-casb</td>
<td>Forcepoint CASB allows you to discover cloud application use, analyze risk, and enforce appropriate controls for SaaS and custom applications.</td>
</tr>
<tr>
<td>Company name</td>
<td>Product name</td>
<td>Integrations</td>
<td>Product ARN</td>
<td>Product description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------</td>
<td>--------------</td>
<td>-------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>Forcepoint</td>
<td>Forcepoint</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/forcepoint/forcepoint-ngfw</td>
<td>Forcepoint NGFW lets you connect your AWS environment into your enterprise network with the scalability, protection, and insights needed to manage your network and respond to threats.</td>
</tr>
<tr>
<td>GuardiCore</td>
<td>Infection Monkey</td>
<td>Send Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/guardicore/aws-infection-monkey</td>
<td>Infection Monkey is an attack simulation tool designed to test networks against attackers.</td>
</tr>
<tr>
<td>IBM</td>
<td>QRadar</td>
<td>Send Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/ibm/qradar-siem</td>
<td>IBM QRadar SIEM provides security teams with the ability to quickly and accurately detect, prioritize, investigate and respond to threats.</td>
</tr>
<tr>
<td>Imperva</td>
<td>Attack Analytics</td>
<td>Send Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/imperva/imperva-attack-analytics</td>
<td>Imperva Attack Analytics correlates and distills thousands of security events into a few readable security incidents.</td>
</tr>
<tr>
<td>McAfee</td>
<td>MVISION Cloud for AWS</td>
<td>Send Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:product/mcafee-skyhigh/mcafee-mvision-cloud-aws</td>
<td>McAfee MVISION Cloud for Amazon Web Services is a comprehensive monitoring, auditing, and remediation solution for your AWS environment.</td>
</tr>
<tr>
<td>PagerDuty</td>
<td>PagerDuty</td>
<td>Receive</td>
<td></td>
<td>PagerDuty's digital operations management platform empowers teams to proactively mitigate customer-impacting issues by automatically turning any signal into the right insight and action.</td>
</tr>
</tbody>
</table>

AWS users can use PagerDuty's set of AWS integrations to scale their AWS and hybrid environments with confidence.

When coupled with AWS Security Hub's aggregated and organized security alerts, PagerDuty allows teams to automate their threat response process and quickly set up custom actions to prevent potential issues.

PagerDuty users undertaking a cloud migration project can move quickly, while decreasing the impact of issues that occur throughout the migration lifecycle.
<table>
<thead>
<tr>
<th>Company name</th>
<th>Product name</th>
<th>Integrations</th>
<th>Product ARN</th>
<th>Product description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Qualys</strong></td>
<td>Vulnerability Management</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:805950163170:product/qualys/qualys-vm</td>
<td>Qualys Vulnerability Management (VM) continuously scans and identifies vulnerabilities, protecting your assets.</td>
</tr>
<tr>
<td><strong>Rackspace</strong></td>
<td>Cloud Native Security</td>
<td>Receive</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:336818582268:product/rackspace-secure</td>
<td>Managed security services on top of native AWS security products for 24x7x365 monitoring by Rackspace SOC, advanced analysis, and threat remediation.</td>
</tr>
<tr>
<td><strong>Rapid7</strong></td>
<td>InsightConnect</td>
<td>Receive</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:336818582268:product/rapid7/insightconnect</td>
<td>Rapid7’s InsightConnect is a security orchestration and automation solution that enables your team to optimize SOC operations with little to no code.</td>
</tr>
<tr>
<td><strong>Rapid7</strong></td>
<td>InsightVM</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:336818582268:product/rapid7/insightvm</td>
<td>Rapid7’s InsightVM provides vulnerability management for modern environments, allowing you to efficiently find, prioritize, and remediate vulnerabilities.</td>
</tr>
<tr>
<td><strong>ServiceNow</strong></td>
<td>ITSM</td>
<td>Receive</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:336818582268:product/insightconnect</td>
<td>The ServiceNow Security Hub integration allows security findings from Security Hub to be viewed within ServiceNow ITSM.</td>
</tr>
<tr>
<td><strong>Slack</strong></td>
<td>Slack</td>
<td>Receive</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:336818582268:product/insightconnect</td>
<td>Slack is a layer of the business technology stack that brings together people, data, and applications. It is a single place where people can effectively work together, find important information, and access hundreds of thousands of critical applications and services to do their best work.</td>
</tr>
</tbody>
</table>
Using custom product integrations to send findings to AWS Security Hub

In addition to findings generated by the integrated AWS services and third-party products, Security Hub can also consume findings that are generated by other custom security products you may use.

<table>
<thead>
<tr>
<th>Company name</th>
<th>Product name</th>
<th>Integration types</th>
<th>Product ARN</th>
<th>Product description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Splunk</td>
<td>Splunk Enterprise</td>
<td>Receive</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:112543817624:product/splunk/splunk-enterprise</td>
<td>Splunk uses Amazon CloudWatch Events as a consumer of Security Hub findings. Send your data to Splunk for advanced security analytics and SIEM.</td>
</tr>
<tr>
<td>Splunk</td>
<td>Splunk Phantom</td>
<td>Receive</td>
<td></td>
<td>With the Splunk Phantom App for AWS Security Hub, findings are sent to Phantom for automated context enrichment with additional threat intelligence information or to perform automated response actions.</td>
</tr>
<tr>
<td>Sumo Logic</td>
<td>Machine Data Analytics</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:956882708938:product/sumologicinc/sumologic-mda</td>
<td>Sumo Logic is a secure, machine data analytics platform that enables DevSecOps teams build, run, and secure their AWS applications.</td>
</tr>
<tr>
<td>Symantec</td>
<td>Cloud Workload Protection</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:754237914691:product/symantec-corp/symantec-cwp</td>
<td>Cloud Workload Protection provides complete protection for your Amazon EC2 instances with anti-malware, intrusion prevention, and file integrity monitoring.</td>
</tr>
<tr>
<td>Tenable</td>
<td>Tenable.io</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:574259144590:product/tenable/tenable-io</td>
<td>Tenable ensures that your cloud infrastructure is secure, compliant, scalable, and cost optimized.</td>
</tr>
<tr>
<td>Turbot</td>
<td>Turbot</td>
<td>Receive</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Twistlock</td>
<td>Enterprise Send Edition</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:496947949261:product/twistlock/twistlock-enterprise</td>
<td>Twistlock is a cloud native cybersecurity platform that protects VMs, containers, and serverless platforms.</td>
</tr>
<tr>
<td>Vectra AI</td>
<td>Cognito Detect</td>
<td>Send</td>
<td>arn:aws:securityhub:&lt;REGION&gt;:360159260416:product/vectra-ai/cognito-detect</td>
<td>Vectra AI is transforming cybersecurity by applying advanced AI to detect and respond to hidden cyberattackers before they can steal or cause damage.</td>
</tr>
</tbody>
</table>
You can send these findings into Security Hub manually using the BatchImportFindings API operation.

**Requirements and recommendations for sending findings from custom security products**

Before you can successfully invoke the BatchImportFindings API operation, you must enable Security Hub.

You must provide the finding details using the AWS Security Finding format. For the findings from your custom integration, use the following requirements and recommendations.

**Setting the product ARN**

When you enable Security Hub, a default product Amazon Resource Name (ARN) for Security Hub is generated in your current account.

This product ARN has the following format: arn:aws:securityhub:<region>:<account-id>:product/<account-id>/default. For example, arn:aws:securityhub:us-west-2:123456789012:product/123456789012/default.

Use this product ARN as the value for the ProductArn attribute when invoking the BatchImportFindings API operation.

**Defining the product name**

We recommend that you use the ProductFields attribute to define the name of the product that generates the findings that you are sending to Security Hub.

For example:

```
"ProductFields":
{
  "ProviderName": "<name of the product>",
  "ProviderVersion": "<product version>"
}
```

**Setting the finding IDs**

You must supply, manage, and increment your own finding IDs, using the Id attribute. Each new finding must have a unique finding ID.

**Setting the account ID**

You must specify your own account ID, using the AwsAccountId attribute.

**Setting the created at and updated at dates**

You must supply your own timestamps for the CreatedAt and UpdatedAt attributes.

**Updating findings from custom products**

In addition to sending new findings from custom products, you can also use the BatchImportFindings API operation to update existing findings from custom products.

To update existing findings, use the existing finding ID (via the Id attribute). Resend the full finding with the appropriate information updated in the request, including a modified UpdatedAt timestamp.
Example custom integrations

You can use the following example custom product integrations as a guide to create your own custom solution.

AWS Fargate and Prowler

   Prowler provides dozens of security configuration checks related to services such as Amazon Redshift, Amazon ElastiCache, Amazon API Gateway, and Amazon CloudFront.

   To learn more, see Use AWS Fargate and Prowler to send security configuration findings about AWS services to Security Hub.

Importing AWS Config rules

   You can use an AWS CloudFormation template to import AWS Config rules into Security Hub.

   To learn more, see How to import AWS Config rules evaluations as findings in Security Hub.
Security standards in AWS Security Hub

AWS Security Hub consumes, aggregates, and analyzes security findings from various supported AWS and third-party products.

Security Hub also generates its own findings as the result of running automated and continuous checks against the rules in a set of supported security standards. These checks provide a readiness score and identify specific accounts and resources that require attention.

Security Hub provides controls for the following standards.

- CIS AWS Foundations (p. 141)
- Payment Card Industry Data Security Standard (PCI DSS) (p. 181)
- AWS Foundational Security Best Practices (p. 217)

For information on Security Hub pricing for security checks, see Security Hub pricing.

Topics

- AWS Config requirements for running security checks (p. 134)
- Schedule for running security checks (p. 135)
- Results of security checks (p. 136)
- Disabling or enabling a security standard (p. 138)
- Viewing details for controls (p. 139)
- Disabling and enabling individual controls (p. 140)
- CIS AWS Foundations standard (p. 141)
- Payment Card Industry Data Security Standard (PCI DSS) (p. 181)
- AWS Foundational Security Best Practices standard (p. 217)

AWS Config requirements for running security checks

To run security checks on your environment’s resources, AWS Security Hub either uses steps specified by the standard, or uses specific AWS Config managed rules. These AWS Config managed rules are referred to as service-linked rules, because they are enabled and controlled by the Security Hub service.

Enabling AWS Config for Security Hub

For the standards to be functional in Security Hub, before you enable a security standard, you must also enable AWS Config in your Security Hub accounts.
Security Hub does not manage AWS Config for you. If you already have AWS Config enabled, you can continue configuring its settings through the AWS Config console or APIs.

If you don’t have AWS Config enabled, you can enable it manually, or you can use the AWS CloudFormation "Enable AWS Config" template in AWS CloudFormation StackSets Sample Templates. If you use the Security Hub multi-account, multi-region enablement script, it also enables AWS Config for you.

**Important**
- If you enable AWS Config in your Security Hub master account, this does not automatically enable AWS Config in the Security Hub member accounts for this master account.
- If you want Security Hub to generate findings against security standards for the resources in a Security Hub member account, you must enable AWS Config in that member account.

**Important**
- When you turn on the AWS Config recorder as part of enabling AWS Config, choose to record all resources supported in a given Region, including global resources.
- For more information, see Getting started with AWS Config in the AWS Config Developer Guide.

### How Security Hub generates AWS Config service-linked rules for Security Standards

After you enable a security standard, Security Hub automatically creates the AWS Config service-linked rules that it needs to run checks against the enabled controls.

For every control that uses an AWS Config service-linked rule or rules, Security Hub creates instances of the required rules in your AWS environment. These service-linked rules are specific to Security Hub. It creates these service-linked rules even if other instances of the same rules already exist.

**Note**
- For AWS Config managed rules, the limit is 150 rules per account per Region. However, when you enable a security standard in Security Hub, the service-linked AWS Config rules that are automatically created do not count towards that limit. You can enable a security standard even if you already have 150 AWS Config managed rules in your account.
- For service-linked rules such as the ones that Security Hub adds for security standards, the limit is 150 rules per account per Region. This is in addition to the 150-rule limit on AWS Config managed rules.

For information about the specific AWS Config managed rules that Security Hub uses for each CIS AWS Foundations control, see CIS AWS Foundations controls (p. 142). For the list of required AWS Config resources for CIS AWS Foundations controls, see the section called “AWS Config resources required for CIS controls” (p. 142).

For information about the specific AWS Config managed rules that Security Hub uses for each Payment Card Industry Data Security Standard (PCI DSS) control, see the section called “PCI DSS controls” (p. 182). For the list of required AWS Config resources for PCI DSS controls, see the section called “AWS Config resources required for PCI DSS controls” (p. 181).

If a finding is generated by a security check that is based on an AWS Config rule, the finding details include a Rules link to open the associated AWS Config rule. To navigate to the AWS Config rule, you must also have an IAM permission in the selected account to navigate to AWS Config.

### Schedule for running security checks

After you enable a security standard, AWS Security Hub begins to run the checks within 2 hours.
After the initial check, the schedule for each control may be either periodic or change-triggered.

- Periodic checks run automatically within 12 hours after the most recent run. You cannot change the periodicity.
- Change-triggered checks run when the associated resource changes state. Even if the resource does not change state, the updated at time for change-triggered checks is refreshed every 18 hours. This helps to indicate that the control is still enabled.

In general, Security Hub uses change-triggered rules whenever possible. For a resource to use a change-triggered rule, there must be AWS Config Configuration Item support.

For a control that is based on a managed AWS Config rule, the control description includes a link to the rule description in the AWS Config Developer Guide. That description includes whether the rule is change-triggered or periodic.

Checks that use Security Hub custom Lambda functions are always periodic.

Results of security checks

When it runs checks against the enabled controls for the enabled security standards, AWS Security Hub generates findings. These findings use the AWS Security Finding Format (ASFF).

Standards-related information in the ASFF

For findings generated by security checks, the Compliance field in the ASFF contains the standards-related findings details. The Compliance field includes the following information.

- A Status field that contains result of the most recent check that Security Hub ran against a given control. The results of the previous checks are kept in an archived state for 90 days.
- A RelatedRequirements array that contains a list of related requirements for the control

Determining the severity of security standards findings

For security standards findings, the severity (Severity.Label) is determined based on an assessment on how easy it would be to compromise AWS resources if the issue is detected.

- CRITICAL – The issue must be remediated immediately to avoid it escalating.
  
  For example, an open S3 bucket is considered a critical severity finding. Because so many actors scan for open S3 buckets, the data in an exposed S3 bucket is likely to be discovered and accessed by others. In general, resources with overly-broad access rights that allow unintended public access are considered critical.
  
  These issues should have the highest SLAs in terms of response times.

- HIGH – The issue must be addressed as a priority.
  
  For example, CloudTrail logging not being enabled is considered a high severity issue, because malicious actors can disable logging to hide the actions taken within infrastructure in which they've established a presence.
  
  We recommend that you treat this security issue as a priority and take immediate remediation steps.
Determined the overall status of a control from its findings

In the findings generated by security checks, the Status field is assigned one of the following values.

- **PASSED**
- **FAILED**
- **WARNING** – Indicates that the check was completed, but Security Hub cannot determine whether the resource is in a PASSED or FAILED state
- **NOT_AVAILABLE** – Indicates that the check cannot be completed because there is a server failure, the resource was deleted, or the result of the AWS Config evaluation was NOT_APPLICABLE.

If the AWS Config evaluation result was NOT_APPLICABLE, then Security Hub automatically archives the finding.

The overall status for each control is based on the Compliance.Status and Workflow.Status values of the active findings for that control. It uses the active findings in the master account and the member accounts. The available values for the overall status are as follows:

- **Passed** – Indicates that one of the following is true.
  - All findings have a Compliance.Status of PASSED.
  - All findings have a Workflow.Status of RESOLVED. If all findings are RESOLVED, then the control status is **Passed**, regardless of Compliance.Status.
  - All findings either have a Compliance.Status of PASSED, or have a Compliance.Status of FAILED but a Workflow.Status of SUPPRESSED.
- **Failed** – Indicates that the following is true.
  - At least one finding has a Compliance.Status of FAILED.
  - None of those FAILED findings has a Workflow.Status of SUPPRESSED or RESOLVED.
- **Unknown** – Indicates that the following is true.
  - At least one finding has a Compliance.Status of WARNING or NOT_AVAILABLE.
  - None of those WARNING or NOT_AVAILABLE findings has a Workflow.Status of SUPPRESSED or RESOLVED.
  - No findings have a Compliance.Status of FAILED.
Determining the security score for a security standard

On the Security standards page, each enabled standard displays a security score, which is between 0% and 100%.

The security score represents the proportion of Passed controls to enabled controls. The score is displayed as a percentage. For example, if 10 controls are enabled for a standard, and 7 of those controls are in a Passed state, then the security score is 70%.

On the Summary page, the Security standards card also displays the security scores for each enabled standard. It also displays a consolidated security score that represents the proportion of passed controls to enabled controls across all of the enabled standards.

Rules for updating standards-related findings

If a subsequent check against a given rule generates a new result (for example, the status of “Avoid the use of the root account” changes from FAILED to PASSED), a new finding is generated that contains the most recent result.

If a subsequent check against a given rule generates a result that is identical to the current result, the existing finding is updated, and no new finding is generated.

Security Hub automatically archives findings from controls if the associated resource was deleted, based on one of the following criteria.

- The finding was not updated in three days.
- The associated AWS Config evaluation returned NOT_APPLICABLE.

Disabling or enabling a security standard

You can disable or enable each security standard.

Remember that Security Hub is regional. When you enable or disable a security standard, it is enabled or disabled only in the current Region, or in the Region that you specify in an API request.

When you disable a security standard, the following occurs.

- The checks for its controls are no longer performed
- No additional findings are generated for its controls
- The related AWS Config rules that Security Hub created are removed

When you enable a security standard, all of the controls for that standard are enabled by default. You can then disable individual controls. See the section called “Disabling and enabling individual controls” (p. 140).

Disabling a security standard (Console)

On the Security standards page, each enabled standard includes an option to disable the standard.

To disable a standard

2. Confirm that you are using Security Hub in the Region in which you want to disable the standard.
4. For the standard you want to disable, choose Disable.

Disabling a security standard (API)

To disable a security standard programmatically, use the BatchDisableStandards operation.

Enabling a security standard (Console)

On the Security standards page, each disabled standard includes an option to enable the standard.

To enable a security standard

1. Make sure that you have enabled AWS Config in the master account and all of the member accounts. See the section called “AWS Config requirements for running security checks” (p. 134).
3. Confirm that you are using Security Hub in the Region in which you want to disable the standard.
5. For the standard you want to enable, choose Enable.

Enabling a security standard (API)

To enable a security standard programmatically, use the BatchEnableStandards operation.

Viewing details for controls

The Security standards page provides access to the supported security standards in Security Hub.

For each enabled standard, you can view and filter the list of controls. For each control, you can view a details page that includes the list of findings for the control.

Displaying the controls for an enabled standard

From the Security standards page, you can display the list of controls associated with that standard.

To display the list of controls for an enabled standard

3. For the standard to display the controls for, choose View results.

For each control, the controls page provides the following information.

- The control identifier and title
- Whether the control is enabled or disabled
- The overall status of the control. See the section called “Determining the overall status of a control from its findings” (p. 137).
- The severity associated with the control
Filtering the list of controls

By default, the list of controls includes all of the controls for the selected standard. You can filter the list based on the control identifier, description, related requirements, status, or severity.

To filter the list of controls based on text in the identifier, description, or a related requirement, begin typing the text. The list is updated automatically to only include controls that contain the matching text.

To filter the list of controls based on the control status, from the dropdown list, select the status to include. For enabled controls, you can show all enabled controls, or only show enabled controls that have a specific overall status (Passed, Failed, or Unknown). You can also choose to only display disabled standards.

To filter the list of controls based on the control severity, from the Severity dropdown list, select the severity to include.

Viewing details and findings for a control

For each control, you can display the details page for that control. The details page includes an overview and the list of findings.

To display the details for a control, choose the control name.

At the top of the details page is an overview of the control and its current status.

At the bottom of the details page is the list of findings for the control. The findings list shows the active findings for the selected control that have a workflow status of NEW, NOTIFIED, or RESOLVED.

You can only filter the list based on text in the finding list. You cannot add or remove filters, or group the findings.

You can also perform the following actions.

- View details for individual findings (p. 45)
- Update the workflow status of findings (p. 45)
- Send findings to custom actions (p. 46)

Disabling and enabling individual controls

When you enable a standard, all of the controls for that standard are enabled by default. You can then disable and enable specific controls within an enabled standard.

When you disable a control, the following occurs.

- The check for the control is no longer performed.
- No additional findings are generated for that control.
- The related AWS Config rules that Security Hub created are removed.

It can be useful to turn off security checks for controls that are not relevant to your environment. For example, if you use a single S3 bucket to log your CloudTrail logs, you can turn off controls related to CloudTrail logging in all accounts and Regions except for the account and Region where the centralized
Disabling a control (Console)

On the Security Hub console, each enabled control provides an option to disable that control.

**To disable a control**

2. Confirm that you are using Security Hub in the Region in which you want to disable the control.
4. For the standard that you want to disable controls for, choose View results.
5. Identify the control that you want to disable, then choose Disable.
6. Enter a reason why you are disabling the control. This can help others in your organization understand why the control is disabled.
7. Choose Disable.

Enabling a control (Console)

For a disabled control, Disabled is displayed next to the control title.

To enable the control at any time, choose Enable.

Disabling or enabling a control (API)

To enable or disable a control programmatically, use the UpdateStandardsControl operation of the Security Hub API.

CIS AWS Foundations standard

Security Hub supports the CIS AWS Foundations standard. For more information, see Securing Amazon Web Services on the CIS website.

AWS Security Hub has satisfied the requirements of CIS Security Software Certification and is hereby awarded CIS Security Software Certification for the following CIS Benchmarks:

- CIS Benchmark for CIS Amazon Web Services Foundations Benchmark, v1.2.0, Level 1
- CIS Benchmark for CIS Amazon Web Services Foundations Benchmark, v1.2.0, Level 2

**Topics**

- AWS Config resources required for CIS controls (p. 142)
- CIS AWS Foundations controls (p. 142)
- CIS AWS Foundations controls that you may want to disable (p. 180)
AWS Security Hub User Guide

AWS Config resources required for CIS controls

To run security checks for the enabled CIS AWS Foundations controls on your environment’s resources, Security Hub either runs through the exact audit steps prescribed for the checks in Securing Amazon Web Services or uses specific AWS Config managed rules.

If you don’t enable all resources in AWS Config, a finding is generated for the control 2.5 – Ensure AWS Config is enabled (p. 156). For other CIS controls, for Security Hub to accurately report findings, you must enable the following resources in AWS Config.

- AWS CloudTrail trail
- Amazon EC2 security group
- Amazon EC2 VPC
- IAM policy
- IAM user
- AWS KMS key
- S3 bucket

If a finding is generated by a security check that is based on an AWS Config rule, the finding details include a Rules link to open the associated AWS Config rule. To navigate to the AWS Config rule, you must also have an IAM permission in the selected account to navigate to AWS Config.

CIS AWS Foundations controls

For the CIS AWS Foundations standard, Security Hub supports the following controls. For each control, the information includes the required AWS Config rule and the remediation steps.

1.1 – Avoid the use of the "root" account

Severity: Critical

AWS Config rule: None

The root account has unrestricted access to all resources in the AWS account. We highly recommend that you avoid using this account. The root account is the most privileged account. Minimizing the use of this account and adopting the principle of least privilege for access management reduces the risk of accidental changes and unintended disclosure of highly privileged credentials.

As a best practice, use your root credentials only when required to perform account and service management tasks. Apply IAM policies directly to groups and roles but not users. For a tutorial on how to set up an administrator for daily use, see Creating your first IAM admin user and group in the IAM User Guide

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.3 in the CIS AWS Foundations Benchmark v1.2. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

Remediation

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter. These are the same steps to remediate findings for 3.3 – Ensure a log metric filter and alarm exist for usage of "root" account (p. 162).
To create an Amazon SNS topic

2. Create an Amazon SNS topic that receives all CIS alarms.

   Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the Amazon Simple Notification Service Developer Guide.

3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).

   Make a note of the associated log group name.

To create a metric filter and alarm

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

   ```
   {$:.userIdentity.type="Root" && $.userIdentity.invokedBy NOT EXISTS && $.eventType !="AwsServiceEvent"
   ```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.

   This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.

   The filter is created, and its details appear.
10. Choose Create Alarm.
11. Under Alarm details, enter a Name and Description for the alarm, such as CIS-1.1-RootAccountUsage.
12. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.
13. Choose Create Alarm.

1.2 – Ensure multi-factor authentication (MFA) is enabled for all IAM users that have a console password

Severity: Medium

AWS Config rule: mfa-enabled-for-iam-console-access

Multi-factor authentication (MFA) adds an extra layer of protection on top of a user name and password. With MFA enabled, when a user signs in to an AWS website, they're prompted for their user name and password as well as for an authentication code from their AWS MFA device.

We recommend enabling MFA for all accounts that have a console password. Enabling MFA provides increased security for console access because it requires the authenticating principal to possess a device that emits a time-sensitive key and have knowledge of a credential.
**Important**
The AWS Config rule used for this check may take up to 4 hours to accurately report results for MFA. Any findings that are generated within the first 4 hours after enabling the CIS security checks may not be accurate. It may also take up to 4 hours after remediating this issue for the check to pass.

**Remediation**

**To configure MFA for a user**

2. Choose **Users**.
3. Choose the **User name** of the user to configure MFA for.
4. Choose **Security credentials** and then choose **Manage** next to **Assigned MFA device**.
5. Follow the **Manage MFA Device** wizard to assign the type of device appropriate for your environment.


**1.3 – Ensure credentials unused for 90 days or greater are disabled**

**Severity:** Medium

**AWS Config rule:** `iam-user-unused-credentials-check`

IAM users can access AWS resources using different types of credentials, such as passwords or access keys.

We recommend that you remove or deactivate all credentials that have been unused in 90 days or more. Disabling or removing unnecessary credentials reduces the window of opportunity for credentials associated with a compromised or abandoned account to be used.

The AWS Config rule for this control uses the `GetCredentialReport` and `GenerateCredentialReport` API operations, which are only updated every four hours. Changes to IAM users can take up to four hours to be visible to this control.

**Remediation**

To get some of the information that you need to monitor accounts for dated credentials, use the IAM console. For example, when you view users in your account, there is a column for **Access key age**, **Password age**, and **Last activity**. If the value in any of these columns is greater than 90 days, make the credentials for those users inactive.

You can also use credential reports to monitor user accounts and identify those with no activity for 90 or more days. You can download credential reports in .csv format from the IAM console. For more information about credential reports, see [Getting credential reports for your AWS Account](https://aws.amazon.com/blogs/security/how-to-delegate-management-of-multi-factor-authentication-to-aws-iam-users/).

After you identify the inactive accounts or unused credentials, use the following steps to disable them.

2. Choose **Users**.
3. Choose the name of the user with credentials over 90 days old.
4. Choose **Security credentials** and then choose **Make inactive** for all sign-in credentials and access keys that haven’t been used in 90 days or more.
1.4 – Ensure access keys are rotated every 90 days or less

Severity: Medium

AWS Config managed rule: access-keys-rotated

Access keys consist of an access key ID and secret access key, which are used to sign programmatic requests that you make to AWS. AWS users need their own access keys to make programmatic calls to AWS from the AWS Command Line Interface (AWS CLI), Tools for Windows PowerShell, the AWS SDKs, or direct HTTP calls using the APIs for individual AWS services.

We recommend that you regularly rotate all access keys. Rotating access keys reduces the chance for an access key that is associated with a compromised or terminated account to be used. Rotate access keys to ensure that data can't be accessed with an old key that might have been lost, cracked, or stolen.

Remediation

To ensure that access keys aren't more than 90 days old

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Users.
3. For each user that shows an Access key age that is greater than 90 days, choose the User name to open the settings for that user.
5. To create a new key for the user:
   a. Choose Create access key.
   b. To save the key content, either download the secret access key, or choose Show and then copy it from the page.
   c. Store the key in a secure location to provide to the user.
   d. Choose Close.
6. Update all applications that were using the previous key to use the new key.
7. For the previous key, choose Make inactive to make the access key inactive. Now the user can't make requests using that key.
8. Confirm that all applications work as expected with the new key.
9. After confirming that all applications work with the new key, delete the previous key. After you delete the access key, you can't recover it.

To delete the previous key, choose the X at the end of the row and then choose Delete.

1.5 – Ensure IAM password policy requires at least one uppercase letter

Severity: Medium

AWS Config rule: iam-password-policy

Password policies, in part, enforce password complexity requirements. Use IAM password policies to ensure that passwords use different character sets.

We recommend that the password policy require at least one uppercase letter. Setting a password complexity policy increases account resiliency against brute force login attempts.
Remediation

To modify the password policy
1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Account settings.
3. Select Requires at least one uppercase letter and then choose Apply password policy.

1.6 – Ensure IAM password policy requires at least one lowercase letter

Severity: Medium
AWS Config rule: iam-password-policy

Password policies, in part, enforce password complexity requirements. Use IAM password policies to ensure that passwords use different character sets. We recommend that the password policy require at least one lowercase letter. Setting a password complexity policy increases account resiliency against brute force login attempts.

Remediation

To modify the password policy
1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Account settings.
3. Select Requires at least one lowercase letter and then choose Apply password policy.

1.7 – Ensure IAM password policy requires at least one symbol

Severity: Medium
AWS Config rule: iam-password-policy

Password policies, in part, enforce password complexity requirements. Use IAM password policies to ensure that passwords use different character sets.

We recommend that the password policy require at least one symbol. Setting a password complexity policy increases account resiliency against brute force login attempts.

Remediation

To modify the password policy
1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Account settings.
3. Select Require at least one non-alphanumeric character and then choose Apply password policy.

1.8 – Ensure IAM password policy requires at least one number

Severity: Medium
AWS Config rule: iam-password-policy
Password policies, in part, enforce password complexity requirements. Use IAM password policies to ensure that passwords use different character sets.

We recommend that the password policy require at least one number. Setting a password complexity policy increases account resiliency against brute force login attempts.

Remediation

To modify the password policy
1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Account settings.
3. Select Requires at least one number and then choose Apply password policy.

1.9 – Ensure IAM password policy requires a minimum length of 14 or greater

Severity: Medium

AWS Config rule: iam-password-policy

Password policies, in part, enforce password complexity requirements. Use IAM password policies to ensure that passwords are at least a given length.

We recommend that the password policy require a minimum password length of 14 characters. Setting a password complexity policy increases account resiliency against brute force login attempts.

Remediation

To modify the password policy
1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Account settings.
3. In the Minimum password length field, enter 14, then choose Apply password policy.

1.10 – Ensure IAM password policy prevents password reuse

Severity: Low

AWS Config rule: iam-password-policy

IAM password policies can prevent the reuse of a given password by the same user.

We recommend that the password policy prevent the reuse of passwords. Preventing password reuse increases account resiliency against brute force login attempts.

Remediation

To modify the password policy
1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Account settings.
3. Select Prevent password reuse and then enter 24 for Number of passwords to remember.
4. Choose Apply password policy.
1.11 – Ensure IAM password policy expires passwords within 90 days or less

Severity: Low

AWS Config rule: iam-password-policy

IAM password policies can require passwords to be rotated or expired after a given number of days.

We recommend that the password policy expire passwords after 90 days or less. Reducing the password lifetime increases account resiliency against brute force login attempts. Requiring regular password changes also helps in the following scenarios:

- Passwords can be stolen or compromised without your knowledge. This can happen via a system compromise, software vulnerability, or internal threat.
- Certain corporate and government web filters or proxy servers can intercept and record traffic even if it's encrypted.
- Many people use the same password for many systems such as work, email, and personal.
- Compromised end-user workstations might have a keystroke logger.

Remediation

To modify the password policy

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Account settings.
3. Select Enable password expiration and then enter 90 for Password expiration period (in days).
4. Choose Apply password policy.

1.12 – Ensure no root account access key exists

Severity: Critical

AWS Config rule: iam-root-access-key-check

The root account is the most privileged user in an AWS account. AWS Access Keys provide programmatic access to a given account.

We recommend that all access keys be associated with the root account be removed. Removing access keys associated with the root account limits vectors that the account can be compromised by. Removing the root access keys also encourages the creation and use of role-based accounts that are least privileged.

Remediation

To deactivate or delete access keys

1. Log in to your account using the root credentials.
2. Choose the account name near the top-right corner of the page and then choose My Security Credentials.
3. In the pop-up warning, choose Continue to Security Credentials.
4. Choose Access keys (access key ID and secret access key).
5. For any existing keys, do one of the following:
• Choose Make Inactive to prevent the key from being used to authenticate the account.
• Choose Delete and then choose Yes to permanently delete the key. You can't recover deleted keys.

1.13 – Ensure MFA is enabled for the "root" account

Severity: Critical

AWS Config rule: root-account-mfa-enabled

The root account is the most privileged user in an account. MFA adds an extra layer of protection on top of a user name and password. With MFA enabled, when a user signs in to an AWS website, they're prompted for their user name and password and for an authentication code from their AWS MFA device.

When you use virtual MFA for root accounts, Security Hub recommends that the device used is not a personal device. Instead, use a dedicated mobile device (tablet or phone) that you manage to keep charged and secured independent of any individual personal devices. This lessens the risks of losing access to the MFA due to device loss, device trade-in, or if the individual owning the device is no longer employed at the company.

Note
This control is not supported in AWS GovCloud (US-East) or AWS GovCloud (US-West).

Remediation

To enable MFA for the root account

1. Log in to your account using the root credentials.
2. Choose the account name near the top-right corner of the page and then choose My Security Credentials.
3. In the pop-up warning, choose Continue to Security Credentials.
5. Choose Activate MFA.
6. Choose the type of device to use for MFA and then choose Continue.
7. Complete the steps to configure the device type appropriate to your selection.

Choose a hardware-based authentication mechanism for best results in passing the check 1.14 – Ensure hardware MFA is enabled for the "root" account (p. 149).

1.14 – Ensure hardware MFA is enabled for the "root" account

Severity: Critical

AWS Config rule: root-account-hardware-mfa-enabled

The root account is the most privileged user in an account. MFA adds an extra layer of protection on top of a user name and password. With MFA enabled, when a user signs in to an AWS website, they're prompted for their user name and password and for an authentication code from their AWS MFA device.

For Level 2, Security Hub recommends that you protect the root account with a hardware MFA. A hardware MFA has a smaller attack surface than a virtual MFA. For example, a hardware MFA doesn't suffer the attack surface introduced by the mobile smartphone that a virtual MFA resides on.

Using hardware MFA for many, many accounts might create a logistical device management issue. If this occurs, consider implementing this Level 2 recommendation selectively to the highest security accounts. You can then apply the Level 1 recommendation to the remaining accounts.
Note
This control is not supported in AWS GovCloud (US-East) or AWS GovCloud (US-West).

Remediation

To enable hardware-based MFA for the root account
1. Log in to your account using the root credentials.
2. Choose the account name near the top-right corner of the page and then choose My Security Credentials.
3. In the pop-up warning, choose Continue to Security Credentials.
5. Choose Activate MFA.
6. Choose a hardware-based (not virtual) device to use for MFA and then choose Continue.
7. Complete the steps to configure the device type appropriate to your selection.

1.16 – Ensure IAM policies are attached only to groups or roles

Severity: Low

AWS Config rule: iam-user-no-policies-check

By default, IAM users, groups, and roles have no access to AWS resources. IAM policies are how privileges are granted to users, groups, or roles.

We recommend that you apply IAM policies directly to groups and roles but not users. Assigning privileges at the group or role level reduces the complexity of access management as the number of users grow. Reducing access management complexity might in turn reduce opportunity for a principal to inadvertently receive or retain excessive privileges.

Remediation

To resolve this issue, create an IAM group, assign the policy to the group, and then add the users to the group. The policy is applied to each user in the group.

To create an IAM group
1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Groups and then choose Create New Group.
3. Enter a name for the group to create and then choose Next Step.
4. Select each policy to assign to the group and then choose Next Step.

The policies that you choose should include any policies currently attached directly to a user account. The next step to resolve a failed check is to add users to a group and then assign the policies to that group. Each user in the group gets assigned the policies assigned to the group.

5. Confirm the details on the Review page and then choose Create Group.

For more information about creating groups, see Creating IAM groups in the IAM User Guide.

To add users to an IAM group
1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Groups.
3. Choose Group Actions and then choose Add Users to Group.
4. Select the users to add to the group and then choose Add Users.

For more information about adding users to groups, see Adding and removing users in an IAM group.

To remove a policy attached directly to a user

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Users.
3. For the user to detach a policy from, choose the name in the User name column.
4. For each policy listed under Attached directly, choose the X on the right side of the page to remove the policy from the user and then choose Remove.
5. Confirm that the user can still use AWS services as expected.

1.20 - Ensure a support role has been created to manage incidents with AWS Support

Severity: Low

AWS Config rule: iam-policy-in-use

AWS provides a support center that can be used for incident notification and response, as well as technical support and customer services.

Create an IAM role to allow authorized users to manage incidents with AWS Support. By implementing least privilege for access control, an IAM role will require an appropriate IAM policy to allow support center access in order to manage incidents with AWS Support.

Remediation

To remediate this issue, create a role to allow authorized users to manage AWS Support incidents.

To create the role to use for AWS Support access

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. In the IAM navigation pane, choose Roles, then choose Create role.
3. For Role type, choose the Another AWS account.
4. For Account ID, enter the AWS account ID of the AWS account to which you want to grant access to your resources.

If the users or groups that will assume this role are in the same account, then enter the local account number.

Note

The administrator of the specified account can grant permission to assume this role to any IAM user in that account. To do this, the administrator attaches a policy to the user or a group that grants permission for the sts:AssumeRole action. In that policy, the resource must be the role ARN.

5. Choose Next: Permissions.
7. Select the check box for the AWSSupportAccess managed policy.
8. Choose Next: Tags.
9. (Optional) To add metadata to the role, attach tags as key-value pairs.
For more information about using tags in IAM, see Tagging IAM users and roles in the IAM User Guide.

10. Choose Next: Review.
11. For Role name, enter a name for your role.
   Role names must be unique within your AWS account. They are not case sensitive.
12. (Optional) For Role description, enter a description for the new role.
13. Review the role, then choose Create role.

1.22 – Ensure IAM policies that allow full "*::*" administrative privileges are not created

Severity: Critical

AWS Config rule: iam-policy-no-statements-with-admin-access

IAM policies define a set of privileges granted to users, groups, or roles. It's recommended and considered a standard security advice to grant least privilege—that is, granting only the permissions required to perform a task. Determine what users need to do and then craft policies that let the users perform only those tasks, instead of allowing full administrative privileges.

It's more secure to start with a minimum set of permissions and grant additional permissions as necessary, rather than starting with permissions that are too lenient and then trying to tighten them later. Providing full administrative privileges instead of restricting to the minimum set of permissions that the user is required to do exposes the resources to potentially unwanted actions.

You should remove IAM policies that have a statement with "Effect": "Allow" with "Action": "*" over "Resource": "*".

Remediation

To modify an IAM policy

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Policies.
3. Select the radio button next to the policy to remove.
4. From the Policy actions drop-down menu, choose Detach.
5. On the Detach policy page, select the radio button next to each user to detach the policy from and then choose Detach policy.

Confirm that the user that you detached the policy from can still access AWS services and resources as expected.

2.1 – Ensure CloudTrail is enabled in all Regions

Severity: Critical

AWS Config rule: multi-region-cloud-trail-enabled

CloudTrail is a service that records AWS API calls for your account and delivers log files to you. The recorded information includes the identity of the API caller, the time of the API call, the source IP address of the API caller, the request parameters, and the response elements returned by the AWS service.
CloudTrail provides a history of AWS API calls for an account, including API calls made via the AWS Management Console, AWS SDKs, command-line tools, and higher-level AWS services (such as AWS CloudFormation).

The AWS API call history produced by CloudTrail enables security analysis, resource change tracking, and compliance auditing. Additionally:

- Ensuring that a multi-Region trail exists ensures that unexpected activity occurring in otherwise unused Regions is detected
- Ensuring that a multi-Region trail exists ensures that Global Service Logging is enabled for a trail by default to capture recording of events generated on AWS global services
- For a multi-Region trail, ensuring that management events configured for all type of Read/Writes ensures recording of management operations that are performed on all resources in an AWS account

Remediation

To create a new trail in CloudTrail

1. Sign in to the AWS Management Console and open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. If you haven't used CloudTrail before, choose Get Started Now.
3. Choose Trails and then choose Create trail.
4. Enter a name for the trail.
5. For Apply trail to all regions, choose Yes.
6. Under Storage location, do one of the following:
   - To create a new S3 bucket for CloudTrail logs, choose Yes next to Create a new S3 bucket and then enter a name for the bucket.
   - Choose No next to Create a new S3 bucket and then select the bucket to use.
7. Choose Advanced and, for Enable log file validation, choose Yes to pass 2.2. – Ensure CloudTrail log file validation is enabled (p. 153).
8. Choose Create.

To update an existing trail in CloudTrail

1. Sign in to the AWS Management Console and open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. Choose Trails.
3. Choose the name of the trail in the Name column.
4. Choose the pencil icon for the Trail settings.
5. For Apply trail to all regions, choose Yes and then choose Save.
6. Choose the pencil icon for the Management events.
7. Select All for Read/Write events, then choose Save.
8. Choose the pencil icon for the Storage location.
9. Choose Yes for Enable log file validation to pass check 2.2, then choose Save.

2.2. – Ensure CloudTrail log file validation is enabled

Severity: Low

AWS Config rule: cloud-trail-log-file-validation-enabled
CloudTrail log file validation creates a digitally signed digest file containing a hash of each log that CloudTrail writes to S3. You can use these digest files to determine whether a log file was changed, deleted, or unchanged after CloudTrail delivered the log.

We recommend that you enable file validation on all trails. Enabling log file validation provides additional integrity checking of CloudTrail logs.

**Remediation**

**To enable CloudTrail log file validation**

1. Open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. Choose **Trails**.
3. Choose the name of a trail to edit in the **Name** column.
4. Choose the pencil icon for the **Storage location**.
5. For **Enable log file validation**, choose **Yes** and then choose **Save**.

---

**2.3 – Ensure the S3 bucket CloudTrail logs to is not publicly accessible**

**Severity:** Critical

**AWS Config rules:** s3-bucket-public-read-prohibited, s3-bucket-public-write-prohibited

CloudTrail logs a record of every API call made in your account. These log files are stored in an S3 bucket. We recommend that the bucket policy, or access control list (ACL), applied to the S3 bucket that CloudTrail logs to prevents public access to the CloudTrail logs. Allowing public access to CloudTrail log content might aid an adversary in identifying weaknesses in the affected account's use or configuration.

**Important**

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled. For example, if you are using Security Hub in the US East (N. Virginia) Region, and you are storing AWS CloudTrail logs in a bucket in the US West (N. California) Region, Security Hub cannot find the bucket in a the US West (N. California) Region. When this happens, the check returns a warning that the resource cannot be located. Similarly, if you are aggregating logs from multiple accounts into a single bucket, the CIS check returns a warning finding for all accounts except the account that owns the bucket. Failed findings are returned when the bucket is located in the account and region where the check is being performed and that bucket is publicly accessible.

To run this check, Security Hub first uses custom logic to look for the bucket where your CloudTrail logs are stored. It then uses the AWS Config managed rules to check that bucket is publicly accessible.

If Security Hub cannot discover the bucket because it is in a different account or region, a warning finding is generated.

If the bucket is discovered and is publicly accessible, the check generates a failed finding.

**Remediation**

**To remove public access for an Amazon S3 bucket**

1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
2. Choose the name of the bucket where your CloudTrail are stored.
3. Choose **Permissions** and then choose **Public access settings**.
4. Choose **Edit**, select all four options, and then choose **Save**.

5. If prompted, enter **confirm** and then choose **Confirm**.

### 2.4 – Ensure CloudTrail trails are integrated with Amazon CloudWatch Logs

**Severity:** Low

**AWS Config rule:** cloud-trail-cloud-watch-logs-enabled

CloudTrail is a web service that records AWS API calls made in a given account. The recorded information includes the identity of the API caller, the time of the API call, the source IP address of the API caller, the request parameters, and the response elements returned by the AWS service.

CloudTrail uses Amazon S3 for log file storage and delivery, so log files are stored durably. In addition to capturing CloudTrail logs in a specified Amazon S3 bucket for long-term analysis, you can perform real-time analysis by configuring CloudTrail to send logs to CloudWatch Logs.

For a trail that is enabled in all Regions in an account, CloudTrail sends log files from all those Regions to a CloudWatch Logs log group.

We recommend that you send CloudTrail logs to CloudWatch Logs.

**Note**

The intent of this recommendation is to ensure that account activity is being captured, monitored, and appropriately alarmed on. CloudWatch Logs is a native way to accomplish this using AWS services but doesn’t preclude the use of an alternate solution.

Sending CloudTrail logs to CloudWatch Logs facilitates real-time and historic activity logging based on user, API, resource, and IP address. It provides the opportunity to establish alarms and notifications for anomalous or sensitivity account activity.

**Remediation**

**To ensure that CloudTrail trails are integrated with CloudWatch Logs**

2. Choose **Trails**.
3. Choose a trail that there is no value for in the **CloudWatch Logs Log group** column.
4. Scroll down to the **CloudWatch Logs** section and then choose **Configure**.
5. In the **New or existing log group** field, do one of the following:
   - To use the default log group, keep the name as is.
   - To use an existing log group, enter the name of the log group to use.
   - To create a new log group, enter a name for the log group to create.
6. Choose **Continue**.
7. Do one of the following:
   - To use the default IAM role, go to the next step.
   - To specify the role to use, choose **View Details**.
     - For **IAM role**, do one of the following:
       - Choose the **CloudTrail_CloudWatchLogs_role** and then select the policy to use in the **Policy Name** drop-down list.
       - Choose **Create a new IAM Role** and then enter a name for the role to create.

A role is created and assigned a policy that grants the necessary permissions.
8. Choose Allow.

For more information, see Configuring CloudWatch Logs monitoring with the console in the AWS CloudTrail User Guide.

2.5 – Ensure AWS Config is enabled

Severity: Medium

AWS Config rule: None

AWS Config is a web service that performs configuration management of supported AWS resources in your account and delivers log files to you. The recorded information includes the configuration item (AWS resource), relationships between configuration items (AWS resources), and any configuration changes between resources.

We recommend that you enable AWS Config in all Regions. The AWS configuration item history that AWS Config captures enables security analysis, resource change tracking, and compliance auditing.

Note
CIS 2.5 requires that AWS Config is enabled in all Regions in which you are using Security Hub. Because Security Hub is a regional service, the check performed for this control checks only the current Region for the account. It does not check all Regions.
You also must record global resources so that security checks against global resources can be checked in each Region.

To run this check, Security Hub performs custom logic to perform the audit steps prescribed for it in the CIS AWS Foundations Benchmark v1.2. Security Hub also requires that global resources are recorded in each Region, because Security Hub is a regional service and performs its security checks on a Region-by-Region basis.

Remediation

To configure AWS Config settings

1. Open the AWS Config console at https://console.aws.amazon.com/config/.
2. Select the Region to configure AWS Config in.
3. If you haven't used AWS Config before, choose Get started.
4. On the Settings page, do the following:
   • Under Resource types to record, select Record all resources supported in this region and Include global resources (e.g., AWS IAM resources).
   • Under Amazon S3 bucket, specify the bucket to use or create a bucket and optionally include a prefix.
   • Under Amazon SNS topic, select an Amazon SNS topic from your account or create one. For more information about Amazon SNS, see the Amazon Simple Notification Service Getting Started Guide.
   • Under AWS Config role, either choose Create AWS Config service-linked role or choose Choose a role from your account and then select the role to use.
5. Choose Next.
6. On the AWS Config rules page, choose Skip.
7. Choose Confirm.

For more information about using AWS Config from the AWS Command Line Interface, see Turning on AWS Config in the AWS Config Developer Guide.
You can also use an AWS CloudFormation template to automate this process. For more information, see the AWS CloudFormation StackSets sample template in the AWS CloudFormation User Guide.

2.6 – Ensure S3 bucket access logging is enabled on the CloudTrail S3 bucket

**Severity:** Low

**AWS Config rule:** s3-bucket-logging-enabled

Amazon S3 bucket access logging generates a log that contains access records for each request made to your S3 bucket. An access log record contains details about the request, such as the request type, the resources specified in the request worked, and the time and date the request was processed.

We recommend that you enable bucket access logging on the CloudTrail S3 bucket.

By enabling S3 bucket logging on target S3 buckets, you can capture all events that might affect objects in a target bucket. Configuring logs to be placed in a separate bucket enables access to log information, which can be useful in security and incident response workflows.

**Important**

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled.

For example, if you are using Security Hub in the US East (N. Virginia) Region, and you are storing AWS CloudTrail logs in a bucket in the US West (N. California) Region, Security Hub cannot find the bucket in a the US West (N. California) Region. When this happens, the check returns a warning that the resource cannot be located.

Similarly, if you are aggregating logs from multiple accounts into a single bucket, the CIS check returns a warning finding for all accounts except the account that owns the bucket.

Failed findings are returned when the bucket is located in the account and region where the check is being performed and that bucket is publicly accessible.

To run this check, Security Hub first uses custom logic to look for the bucket where your CloudTrail logs are stored and then uses the AWS Config managed rule to check if logging is enabled.

If the bucket cannot be discovered because it is in a different account or region, a warning finding is generated. If the bucket is discovered and is publicly accessible, the check generates a failed finding.

**Remediation**

**To enable S3 bucket access logging**

1. Open the Amazon S3 console at [https://console.aws.amazon.com/s3/](https://console.aws.amazon.com/s3/).
2. Choose the bucket used for CloudTrail.
3. Choose Properties.
4. Choose Server access logging, then choose Enable logging.
5. Select a bucket from the Target bucket list, and optionally enter a prefix.
6. Choose Save.

2.7 – Ensure CloudTrail logs are encrypted at rest using AWS KMS CMKs

**Severity:** Medium

**AWS Config rule:** cloud-trail-encryption-enabled

157
CloudTrail is a web service that records AWS API calls for an account and makes those logs available to users and resources in accordance with IAM policies. AWS Key Management Service (AWS KMS) is a managed service that helps create and control the encryption keys used to encrypt account data, and uses hardware security modules (HSMs) to protect the security of encryption keys.

You can configure CloudTrail logs to leverage server-side encryption (SSE) and AWS KMS customer-created master keys (CMKs) to further protect CloudTrail logs.

We recommend that you configure CloudTrail to use SSE-KMS.

Configuring CloudTrail to use SSE-KMS provides additional confidentiality controls on log data because a given user must have S3 read permission on the corresponding log bucket and must be granted decrypt permission by the CMK policy.

Remediation

To enable encryption for CloudTrail logs

1. Open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. Choose Trails.
3. Choose the trail to update.
4. Under Storage location, choose the pencil icon to edit the settings.
5. For Encrypt log files with SSE-KMS, choose Yes.
6. For Create a new KMS key, do one of the following:
   • To create a key, choose Yes and then enter an alias for the key in the KMS key field. The key is created in the same Region as the bucket.
   • To use an existing key, choose No and then select the key from the KMS key list.
   Note
   The AWS KMS key and S3 bucket must be in the same Region.
7. Choose Save.

You might need to modify the policy for CloudTrail to successfully interact with your CMK. For more information, see Encrypting CloudTrail log files with AWS KMS–Managed Keys (SSE-KMS) in the AWS CloudTrail User Guide.

2.8 – Ensure rotation for customer-created CMKs is enabled

Severity: High

AWS Config rule: cmk-backing-key-rotation-enabled

AWS KMS enables customers to rotate the backing key, which is key material stored in AWS KMS and is tied to the key ID of the CMK. It’s the backing key that is used to perform cryptographic operations such as encryption and decryption. Automated key rotation currently retains all previous backing keys so that decryption of encrypted data can take place transparently.

We recommend that you enable CMK key rotation. Rotating encryption keys helps reduce the potential impact of a compromised key because data encrypted with a new key can’t be accessed with a previous key that might have been exposed.

Remediation

To enable CMK rotation

2. To change the AWS Region, use the Region selector in the upper-right corner of the page.
3. Choose Customer managed keys.
4. Choose the alias of the key to update in the Alias column.
5. Choose Key rotation.
6. Select Automatically rotate this CMK every year and then choose Save.

2.9 – Ensure VPC flow logging is enabled in all VPCs

Severity: Medium

AWS Config rule: vpc-flow-logs-enabled

VPC flow logs is a feature that enables you to capture information about the IP traffic going to and from network interfaces in your VPC. After you have created a flow log, you can view and retrieve its data in CloudWatch Logs.

We recommend that you enable flow logging for packet rejects for VPCs. Flow logs provide visibility into network traffic that traverses the VPC and can detect anomalous traffic or insight during security workflows.

Remediation

To enable VPC flow logging
1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. Choose Your VPCs.
3. Select a VPC to update.
4. Choose the Flow Logs tab in the bottom section of the page.
5. Choose Create flow log.
6. For Filter, choose Reject.
7. For Destination log group, select the log group to use.
8. For IAM role, select the IAM role to use.
9. Choose Create.

3.1 – Ensure a log metric filter and alarm exist for unauthorized API calls

Severity: Medium

AWS Config rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms.

We recommend that you create a metric filter and alarm unauthorized API calls. Monitoring unauthorized API calls helps reveal application errors and might reduce time to detect malicious activity.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.1 in the CIS AWS Foundations Benchmark v1.2. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.
Important
Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled. For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region. When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

Remediation
The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

To create an Amazon SNS topic
2. Create an Amazon SNS topic that receives all CIS alarms.

   Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the Amazon Simple Notification Service Developer Guide.
3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).

   Make a note of the associated log group name.

To create a metric filter and alarm
2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

   ```{(($.errorCode="UnauthorizedOperation") || ($.errorCode="AccessDenied"))}

6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.

   This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.

   The filter is created, and its details appear.
10. Choose Create Alarm.
11. Under Alarm details, enter a Name and Description for the alarm, such as CIS-3.1-UnauthorizedAPICalls.
12. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.
13. Choose Create Alarm.
3.2 – Ensure a log metric filter and alarm exist for AWS Management Console sign-in without MFA

Severity: Medium

AWS Config rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms.

We recommend that you create a metric filter and alarm console logins that aren’t protected by MFA. Monitoring for single-factor console logins increases visibility into accounts that aren’t protected by MFA.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.2 in the CIS AWS Foundations Benchmark v1.2. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

Important

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled. For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can’t locate the CloudWatch alarms or SNS topics in the US West (N. California) Region. When this happens, the check returns a warning that the resource can’t be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

Remediation

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

To create an Amazon SNS topic

2. Create an Amazon SNS topic that receives all CIS alarms.

Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the Amazon Simple Notification Service Developer Guide.

3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).

Make a note of the associated log group name.

To create a metric filter and alarm

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

```java
{($.eventName="ConsoleLogin") && ($.additionalEventData.MFAUsed !="Yes")}
```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.
   
   This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.
   
   The filter is created, and its details appear.
10. Choose Create Alarm.
11. Under Alarm details, enter a Name and Description for the alarm, such as CIS-3.2-ConsoleSigninWithoutMFA.
12. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.
13. Choose Create Alarm.

3.3 – Ensure a log metric filter and alarm exist for usage of "root" account

Severity: Medium

AWS Config rule: None

You can do real-time monitoring of API calls directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms.

We recommend that you create a metric filter and alarm for root login attempts. Monitoring for root account logins provides visibility into the use of a fully privileged account and an opportunity to reduce the use of it.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.3 in the CIS AWS Foundations Benchmark v1.2. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

Important

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled.

For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region.

When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

Remediation

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

To create an Amazon SNS topic

2. Create an Amazon SNS topic that receives all CIS alarms.
   
   Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the Amazon Simple Notification Service Developer Guide.
3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).

Make a note of the associated log group name.

**To create a metric filter and alarm**

2. Choose **Logs**.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the **Metric Filters** column.
4. Choose **Add Metric Filter**.
5. Copy the following pattern and then paste it into the **Filter Pattern** field.

```plaintext
{$.userIdentity.type="Root" && $.userIdentity.invokedBy NOT EXISTS && $.eventType != "AwsServiceEvent"}
```

6. Choose **Assign Metric**.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for **Metric Namespace** is **LogMetrics**. This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the **Metric Name** field and then choose **Create Filter**.

The filter is created, and its details appear.
10. Choose **Create Alarm**.
11. Under **Alarm details**, enter a **Name** and **Description** for the alarm, such as **CIS-3.3-RootAccountUsage**.
12. Under **Actions**, for **Send notification to**, choose **Enter list** and then enter the name of the topic that you created in the previous procedure.
13. Choose **Create Alarm**.

**3.4 – Ensure a log metric filter and alarm exist for IAM policy changes**

**Severity**: Medium

**AWS Config** rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms.

We recommend that you create a metric filter and alarm for changes made to IAM policies. Monitoring these changes helps ensure that authentication and authorization controls remain intact.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.4 in the **CIS AWS Foundations Benchmark v1.2**. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

**Important**

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled.

For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running
in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region. When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

**Remediation**

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

**To create an Amazon SNS topic**

2. Create an Amazon SNS topic that receives all CIS alarms.
3. Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the Amazon Simple Notification Service Developer Guide.
4. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).

Make a note of the associated log group name.

**To create a metric filter and alarm**

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

```
{($.eventName=DeleteGroupPolicy) || ($.eventName=DeleteRolePolicy) || ($.eventName=DeleteUserPolicy) || ($.eventName=PutGroupPolicy) || ($.eventName=PutRolePolicy) || ($.eventName=PutUserPolicy) || ($.eventName=CreatePolicy) || ($.eventName=DeletePolicy) || ($.eventName=CreatePolicyVersion) || ($.eventName=DeletePolicyVersion) || ($.eventName=AttachRolePolicy) || ($.eventName=DetachRolePolicy) || ($.eventName=AttachUserPolicy) || ($.eventName=DetachUserPolicy) || ($.eventName=AttachGroupPolicy) || ($.eventName=DetachGroupPolicy)}
```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.
9. Enter a name in the Metric Name field and then choose Create Filter.
10. The filter is created, and its details appear.
11. Choose Create Alarm.
12. Under Alarm details, enter a Name and Description for the alarm, such as CIS-3.4-IAMPolicyChanges.
13. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.
13. Choose **Create Alarm**.

### 3.5 – Ensure a log metric filter and alarm exist for CloudTrail configuration changes

**Severity:** Medium

**AWS Config** rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms.

We recommend that you create a metric filter and alarm for changes to CloudTrail configuration settings. Monitoring these changes helps ensure sustained visibility to activities in the account.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.5 in the **CIS AWS Foundations Benchmark v1.2**. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

**Important**

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled. For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region.

When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

**Remediation**

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

**To create an Amazon SNS topic**

2. Create an Amazon SNS topic that receives all CIS alarms.

   Create at least one subscriber to the topic. For more information, see **Getting started with Amazon SNS** in the **Amazon Simple Notification Service Developer Guide**.

3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – **Ensure CloudTrail is enabled in all Regions** (p. 152).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

2. Choose **Logs**.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the **Metric Filters** column.
4. Choose **Add Metric Filter**.
5. Copy the following pattern and then paste it into the **Filter Pattern** field.

```
{($.eventName=CreateTrail) || ($.eventName=UpdateTrail) || ($.eventName=DeleteTrail) ||
($.eventName=StartLogging) || ($.eventName=StopLogging)}
```

6. Choose **Assign Metric**.

7. (Optional) Update the filter name to a name of your choice.

8. Confirm that the value for **Metric Namespace** is **LogMetrics**.

   This ensures that all CIS Benchmark metrics are grouped together.

9. Enter a name in the **Metric Name** field and then choose **Create Filter**.

   The filter is created, and its details appear.

10. Choose **Create Alarm**.

11. Under **Alarm details**, enter a **Name** and **Description** for the alarm, such as **CIS-3.5-CloudTrailChanges**.

12. Under **Actions**, for **Send notification to**, choose **Enter list** and then enter the name of the topic that you created in the previous procedure.

13. Choose **Create Alarm**.

### 3.6 – Ensure a log metric filter and alarm exist for AWS Management Console authentication failures

**Severity:** Medium

**AWS Config** rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms.

We recommend that you create a metric filter and alarm for failed console authentication attempts. Monitoring failed console logins might decrease lead time to detect an attempt to brute-force a credential, which might provide an indicator, such as source IP, that you can use in other event correlations.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.6 in the **CIS AWS Foundations Benchmark v1.2**. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

**Important**

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled. For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can’t locate the CloudWatch alarms or SNS topics in the US West (N. California) Region.

When this happens, the check returns a warning that the resource can’t be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

**Remediation**

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.
To create an Amazon SNS topic

2. Create an Amazon SNS topic that receives all CIS alarms.
   
   Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the *Amazon Simple Notification Service Developer Guide*.
3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).
   
   Make a note of the associated log group name.

To create a metric filter and alarm

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.
   
   ```
   {($.eventName=ConsoleLogin) && ($.errorMessage="Failed authentication")}
   ```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.
   
   This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.
   
   The filter is created, and its details appear.
10. Choose Create Alarm.
11. Under Alarm details, enter a Name and Description for the alarm, such as CIS-3.6-ConsoleAuthenticationFailure.
12. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.
13. Choose Create Alarm.

3.7 – Ensure a log metric filter and alarm exist for disabling or scheduled deletion of customer created CMKs

**Severity:** Medium

**AWS Config** rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms.

We recommend that you create a metric filter and alarm for customer-created CMKs that have changed state to disabled or scheduled deletion. Data encrypted with disabled or deleted keys is no longer accessible.
To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.7 in the CIS AWS Foundations Benchmark v1.2. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

**Important**
Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled.
For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region.
When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

**Remediation**

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

**To create an Amazon SNS topic**
2. Create an Amazon SNS topic that receives all CIS alarms.
   Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the Amazon Simple Notification Service Developer Guide.
3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).
   Make a note of the associated log group name.

**To create a metric filter and alarm**
2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

```plaintext
{($.eventSource=kms.amazonaws.com) && ($eventName=DisableKey) || ($eventName=ScheduleKeyDeletion))
```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.
   This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.
   The filter is created, and its details appear.
10. Choose Create Alarm.
11. Under Alarm details, enter a Name and Description for the alarm, such as CIS-3.7-DisableOrDeleteCMK.
12. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.

13. Choose Create Alarm.

### 3.8 – Ensure a log metric filter and alarm exist for S3 bucket policy changes

**Severity:** Medium

**AWS Config** rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms.

We recommend that you create a metric filter and alarm for changes to S3 bucket policies. Monitoring these changes might reduce time to detect and correct permissive policies on sensitive S3 buckets.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.8 in the CIS AWS Foundations Benchmark v1.2. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

**Important**

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled. For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region. When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

**Remediation**

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

**To create an Amazon SNS topic**

2. Create an Amazon SNS topic that receives all CIS alarms.
3. Create at least one subscriber to the topic. For more information, see [Getting started with Amazon SNS](https://docs.aws.amazon.com/sns/latest/dg/sns-getting-started.html) in the Amazon Simple Notification Service Developer Guide.
4. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).

Make a note of the associated log group name.

**To create a metric filter and alarm**

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

```
{($.eventSource=s3.amazonaws.com) && 
  (($.eventName=PutBucketAcl) || ($injectTemplateResult$.eventName=PutBucketPolicy) || ($injectTemplateResult$.eventName=PutBucketCors) ||
  ($injectTemplateResult$.eventName=PutBucketLifecycle) || ($injectTemplateResult$.eventName=PutBucketReplication) ||
  ($injectTemplateResult$.eventName=DeleteBucketPolicy) || ($injectTemplateResult$.eventName=DeleteBucketCors) ||
  ($injectTemplateResult$.eventName=DeleteBucketLifecycle) || ($injectTemplateResult$.eventName=DeleteBucketReplication))}
```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.
   This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.
   The filter is created, and its details appear.
10. Choose Create Alarm.
11. Under Alarm details, enter a Name and Description for the alarm, such as CIS-3.8-S3BucketPolicyChanges.
12. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.
13. Choose Create Alarm.

3.9 – Ensure a log metric filter and alarm exist for AWS Config configuration changes

Severity: Medium

AWS Config rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms.

We recommend that you create a metric filter and alarm for changes to AWS Config configuration settings. Monitoring these changes helps ensure sustained visibility of configuration items in the account.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.9 in the CIS AWS Foundations Benchmark v1.2. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

Important

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled.

For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region.

When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

Remediation

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.
To create an Amazon SNS topic

2. Create an Amazon SNS topic that receives all CIS alarms.

    Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the Amazon Simple Notification Service Developer Guide.
3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).

    Make a note of the associated log group name.

To create a metric filter and alarm

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

   ```
   { ($.eventSource=config.amazonaws.com) && ($.eventName=StopConfigurationRecorder) 
   || ($.eventName=DeleteDeliveryChannel) || ($.eventName=PutDeliveryChannel) || 
   ($.eventName=PutConfigurationRecorder))
   ```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.

    This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.

    The filter is created, and its details appear.
10. Choose Create Alarm.
11. Under Alarm details, enter a Name and Description for the alarm, such as CIS-3.9-AWSConfigChanges.
12. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.
13. Choose Create Alarm.

3.10 – Ensure a log metric filter and alarm exist for security group changes

Severity: Medium

AWS Config rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms. Security groups are a stateful packet filter that controls ingress and egress traffic in a VPC.

We recommend that you create a metric filter and alarm for changes to security groups. Monitoring these changes helps ensure that resources and services aren't unintentionally exposed.
To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.10 in the CIS AWS Foundations Benchmark v1.2. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

Important
Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled.
For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region.
When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

Remediation
The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

To create an Amazon SNS topic
2. Create an Amazon SNS topic that receives all CIS alarms.
   Create at least one subscriber to the topic. For more information, see Getting Started with Amazon SNS in the Amazon Simple Notification Service Developer Guide.
3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).
   Make a note of the associated log group name.

To create a metric filter and alarm
2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

```bash
{($.eventName=AuthorizeSecurityGroupIngress) || ($.eventName=AuthorizeSecurityGroupEgress) || ($.eventName=RevokeSecurityGroupIngress) || $.eventName=RevokeSecurityGroupEgress) || ($.eventName=CreateSecurityGroup) || ($.eventName=DeleteSecurityGroup)}
```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.
   This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.
   The filter is created, and its details appear.
10. Choose Create Alarm.
11. Under **Alarm details**, enter a **Name** and **Description** for the alarm, such as **CIS-3.10-SecurityGroupChanges**.

12. Under **Actions**, for **Send notification to**, choose **Enter list** and then enter the name of the topic that you created in the previous procedure.

13. Choose **Create Alarm**.

### 3.11 – Ensure a log metric filter and alarm exist for changes to Network Access Control Lists (NACL)

**Severity:** Medium

**AWS Config** rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms. NACLs are used as a stateless packet filter to control ingress and egress traffic for subnets in a VPC.

We recommend that you create a metric filter and alarm for changes to NACLs. Monitoring these changes helps ensure that AWS resources and services aren't unintentionally exposed.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.11 in the **CIS AWS Foundations Benchmark v1.2**. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

**Important**

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled.

For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region.

When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

**Remediation**

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

**To create an Amazon SNS topic**


2. Create an Amazon SNS topic that receives all CIS alarms.

   Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the [Amazon Simple Notification Service Developer Guide](https://docs.aws.amazon.com/sns/latest/dg/sns-getting-started.html).

3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in **2.1 – Ensure CloudTrail is enabled in all Regions** (p. 152).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the **Metric Filters** column.
5. Copy the following pattern and then paste it into the **Filter Pattern** field.

   ```
   {($.eventName=CreateNetworkAcl) || ($.eventName=CreateNetworkAclEntry) ||
   ($.eventName=DeleteNetworkAcl) || ($.eventName=DeleteNetworkAclEntry) ||
   ($.eventName=ReplaceNetworkAclEntry) || ($.eventName=ReplaceNetworkAclAssociation)}
   ```

6. Choose **Assign Metric**.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for **Metric Namespace** is **LogMetrics**.
9. Enter a name in the **Metric Name** field and then choose **Create Filter**.

The filter is created, and its details appear.
10. Choose **Create Alarm**.
11. Under **Alarm details**, enter a **Name** and **Description** for the alarm, such as **CIS-3.11-NetworkACLChanges**.
12. Under **Actions**, for **Send notification to**, choose **Enter list** and then enter the name of the topic that you created in the previous procedure.
13. Choose **Create Alarm**.

### 3.12 – Ensure a log metric filter and alarm exist for changes to network gateways

**Severity:** Medium

**AWS Config** rule: None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms. Network gateways are required to send and receive traffic to a destination outside a VPC.

We recommend that you create a metric filter and alarm for changes to network gateways. Monitoring these changes helps ensure that all ingress and egress traffic traverses the VPC border via a controlled path.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.12 in the **CIS AWS Foundations Benchmark v1.2**. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

**Important**

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled.

For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can’t locate the CloudWatch alarms or SNS topics in the US West (N. California) Region.

When this happens, the check returns a warning that the resource can’t be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.
Remediation

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

To create an Amazon SNS topic

2. Create an Amazon SNS topic that receives all CIS alarms.
   
   Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the Amazon Simple Notification Service Developer Guide.
3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).
   
   Make a note of the associated log group name.

To create a metric filter and alarm

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

   $(.eventName=CreateCustomerGateway) || $(.eventName=DeleteCustomerGateway) || $(.eventName=AttachInternetGateway) || $(.eventName=CreateInternetGateway) || $(.eventName=DeleteInternetGateway) || $(.eventName=DetachInternetGateway)

6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.
   
   This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.
   
   The filter is created, and its details appear.
10. Choose Create Alarm.
11. Under Alarm details, enter a Name and Description for the alarm, such as CIS-3.12- NetworkGatewayChanges.
12. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.
13. Choose Create Alarm.

3.13 – Ensure a log metric filter and alarm exist for route table changes

Severity: Medium

AWS Config rule: None
You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms. Routing tables route network traffic between subnets and to network gateways.

We recommend that you create a metric filter and alarm for changes to route tables. Monitoring these changes helps ensure that all VPC traffic flows through an expected path.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.13 in the CIS AWS Foundations Benchmark v1.2. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

**Important**
Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled.
For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region.
When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

**Remediation**

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

**To create an Amazon SNS topic**

2. Create an Amazon SNS topic that receives all CIS alarms.
   Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the Amazon Simple Notification Service Developer Guide.
3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in 2.1 – Ensure CloudTrail is enabled in all Regions (p. 152).
   Make a note of the associated log group name.

**To create a metric filter and alarm**

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

   ```
   ( ($.eventName=CreateRoute) || ($.eventName=CreateRouteTable) ||
   ($.eventName=ReplaceRoute) || ($.eventName=ReplaceRouteTableAssociation) ||
   ($.eventName=DeleteRoute) || ($.eventName=DeleteRouteTable) ||
   ($.eventName=DisassociateRouteTable))
   ```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.
This ensures that all CIS Benchmark metrics are grouped together.

9. Enter a name in the **Metric Name** field and then choose **Create Filter**.

The filter is created, and its details appear.

10. Choose **Create Alarm**.

11. Under **Alarm details**, enter a **Name** and **Description** for the alarm, such as **CIS-3.13-RouteTableChanges**.

12. Under **Actions**, for **Send notification to**, choose **Enter list** and then enter the name of the topic that you created in the previous procedure.

13. Choose **Create Alarm**.

### 3.14 – Ensure a log metric filter and alarm exist for VPC changes

**Severity:** Medium

**AWS Config rule:** None

You can do real-time monitoring of API calls by directing CloudTrail logs to CloudWatch Logs and establishing corresponding metric filters and alarms. You can have more than one VPC in an account, and you can create a peer connection between two VPCs, enabling network traffic to route between VPCs.

We recommend that you create a metric filter and alarm for changes to VPCs. Monitoring these changes helps ensure that authentication and authorization controls remain intact.

To run this check, Security Hub uses custom logic to perform the exact audit steps prescribed for control 3.14 in the **CIS AWS Foundations Benchmark v1.2**. This control fails if the exact metric filters prescribed by CIS are not used. Additional fields or terms cannot be added to the metric filters.

**Important**

Security Hub supports CIS AWS Foundations checks only on resources in the same Region and owned by the same account as the one in which Security Hub is enabled.

For example, if you enable Security Hub in the US East (N. Virginia) Region, but you create CloudWatch alarms or SNS topics in the US West (N. California) Region, Security Hub running in the US East (N. Virginia) Region can't locate the CloudWatch alarms or SNS topics in the US West (N. California) Region.

When this happens, the check returns a warning that the resource can't be located. A Failed finding is generated only when a resource is successfully located but is not compliant with CIS requirements for the control.

**Remediation**

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

**Create an Amazon SNS topic**


2. Create an Amazon SNS topic that receives all CIS alarms.

Create at least one subscriber to the topic. For more information, see Getting started with Amazon SNS in the **Amazon Simple Notification Service Developer Guide**.

3. Set up an active CloudTrail that applies to all Regions. To do so, follow the remediation steps in **2.1 – Ensure CloudTrail is enabled in all Regions** (p. 152).

Make a note of the associated log group name.
To create a metric filter and alarm

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

```plaintext
{($.eventName=CreateVpc) || ($.eventName=DeleteVpc) ||
($.eventName=ModifyVpcAttribute) || ($.eventName=AcceptVpcPeeringConnection) ||
($.eventName=CreateVpcPeeringConnection) || ($.eventName=DeleteVpcPeeringConnection)
|| ($.eventName=RejectVpcPeeringConnection) || ($.eventName=AttachClassicLinkVpc)
|| ($.eventName=DetachClassicLinkVpc) || ($.eventName=DisableVpcClassicLink) ||
($.eventName=EnableVpcClassicLink)}
```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.

This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.

The filter is created, and its details appear.
10. Choose Create Alarm.
11. Under Alarm details, enter a Name and Description for the alarm, such as CIS-3.14-VPCChanges.
12. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.
13. Choose Create Alarm.

4.1 – Ensure no security groups allow ingress from 0.0.0.0/0 to port 22

Severity: High

**AWS Config rule:** restricted-ssh

Security groups provide stateful filtering of ingress and egress network traffic to AWS resources.

We recommend that no security group allow unrestricted ingress access to port 22. Removing unfettered connectivity to remote console services, such as SSH, reduces a server’s exposure to risk.

**Remediation**

Perform the following steps for each security group associated with a VPC.

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the left pane, choose Security groups.
3. Select a security group.
4. In the bottom section of the page, choose the Inbound Rules tab.
5. Choose Edit rules.
6. Identify the rule that allows access through port 22 and then choose the X to remove it.
7. Choose **Save rules**.

### 4.2 – Ensure no security groups allow ingress from 0.0.0.0/0 to port 3389

**Severity:** High

**AWS Config rule:** restricted-common-ports

Security groups provide stateful filtering of ingress and egress network traffic to AWS resources.

We recommend that no security group allow unrestricted ingress access to port 3389. Removing unfettered connectivity to remote console services, such as RDP, reduces a server’s exposure to risk.

**Remediation**

Perform the following steps for each security group associated with a VPC.

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the left pane, choose **Security groups**.
3. Select a security group.
4. In the bottom section of the page, choose the **Inbound Rules** tab.
5. Choose **Edit rules**.
6. Identify the rule that allows access through port 3389 and then choose the **X** to remove it.
7. Choose **Save rules**.

### 4.3 – Ensure the default security group of every VPC restricts all traffic

**Severity:** Medium

**AWS Config rule:** vpc-default-security-group-closed

A VPC comes with a default security group with initial settings that deny all inbound traffic, allow all outbound traffic, and allow all traffic between instances assigned to the security group. If you don’t specify a security group when you launch an instance, the instance is automatically assigned to this default security group. Security groups provide stateful filtering of ingress and egress network traffic to AWS resources.

We recommend that the default security group restrict all traffic.

Update the default security group for the default VPC in every Region to comply. Any new VPCs automatically contain a default security group that you need to remediate to comply with this recommendation.

**Note**

When implementing this recommendation, you can use VPC flow logging, enabled for check 2.9 (p. 159), to determine the least-privilege port access required by systems to work properly because it can log all packet acceptances and rejections occurring under the current security groups.

Configuring all VPC default security groups to restrict all traffic encourages least-privilege security group development and mindful placement of AWS resources into security groups, which in turn reduces the exposure of those resources.
Remediation

To update the default security group to restrict all access

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. View the default security groups details to see the resources that are assigned to them.
3. Create a set of least-privilege security groups for the resources.
4. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
5. On the Amazon EC2 console, change the security group for the resources that use the default security groups to the least-privilege security group you created.
6. For each default security group, choose the **Inbound** tab and delete all inbound rules.
7. For each default security group, choose the **Outbound** tab and delete all outbound rules.

For more information, see Working with Security Groups in the *Amazon VPC User Guide*.

CIS AWS Foundations controls that you may want to disable

For the CIS AWS Foundations standard, here are some specific controls that you may want to disable.

- CIS AWS Foundations Benchmark 2.3 and 2.6 controls.
  
  These controls deal with logging of CloudTrail trails. If you log these trails in a centralized logging account, you only need to run these controls in the account and Region where centralized logging takes place.

- CIS AWS Foundations Benchmark 1.2-1.14, 1.16, 1.22, and 2.5 controls.

  To save on the cost of AWS Config rules, you can disable recording of global resources in all but one Region, and then disable these controls that deal with global resources in all Regions except for the Region that runs global recording.

  If you disable these 1.x controls and disable recording of global Regions, you also must disable 2.5. This is because 2.5 requires recording of global resources in order to pass.

- CIS AWS Foundations Benchmark 3.x controls.

  If you have an SNS topic in a centralized account that is aligned to these control requirements, you can disable these controls in all accounts except for that centralized account.

CIS AWS Foundations security checks that are not supported in Security Hub

The following rules are *not* supported in the CIS AWS Foundations standard in Security Hub, because they cannot be evaluated in an automated way. Security Hub focuses on automated security checks.

- 1.15 – Ensure security questions are registered in the AWS account
- 1.17 – Maintain current contact details
- 1.18 – Ensure security contact information is registered
- 1.19 – Ensure IAM instance roles are used for AWS resource access from instances
- 1.21 – Do not set up access keys during initial user setup for all IAM users that have a console password
- 4.4 – Ensure routing tables for VPC peering are "least access"

Payment Card Industry Data Security Standard (PCI DSS)

The Payment Card Industry Data Security Standard (PCI DSS) standard in Security Hub consists of a set of AWS security best practices controls. Each control applies to a specific AWS resource, and relates to one or more PCI DSS version 3.2.1 requirements. A PCI DSS requirement can be related to multiple controls. The details page for each PCI DSS control lists the specific PCI DSS requirements that are related to that control. See the section called “Viewing details for controls” (p. 139).

The PCI DSS Compliance Standard in Security Hub is designed to help you with your ongoing PCI DSS security activities. The controls cannot verify whether your systems are compliant with the PCI DSS standard. They can neither replace internal efforts nor guarantee that you will pass a PCI DSS assessment. Security Hub does not check procedural controls that require manual evidence collection.

Security Hub currently scopes the controls at the account level. It is recommended that you enable these controls in all of your accounts that have resources that store, process, and/or transmit cardholder data.

This standard was validated by AWS Security Assurance Services LLC (AWS SAS), which is a team of Qualified Security Assessors (QSAs) certified to provide PCI DSS guidance and assessments by the PCI DSS Security Standards Council (PCI SSC). AWS SAS have confirmed that the automated checks can assist a customer in preparing for a PCI DSS assessment.

Contents
- AWS Config resources required for PCI DSS controls (p. 181)
- PCI DSS controls (p. 182)

AWS Config resources required for PCI DSS controls

For AWS Security Hub to accurately report findings for all of the PCI DSS controls, you must enable the following resources in AWS Config.

- Amazon ES domain
- Auto scaling group
- CloudTrail trail
- CodeBuild project
- Amazon EC2 elastic IP
- Amazon EC2 security group
- Amazon EC2 volume
- IAM user
- AWS KMS key
- IAM policy
- Lambda function
- Amazon RDS DB instance
- Amazon RDS snapshot
- Amazon Redshift cluster
- S3 bucket
- Systems Manager patch compliance
PCI DSS controls

The PCI DSS security standard in Security Hub supports the following controls. For each control, the information includes the severity, the resource type, the AWS Config rule, and the remediation steps.

[PCI.AutoScaling.1] Auto scaling groups associated with a load balancer should use health checks

Severity: Low

Resource: Auto scaling group

AWS Config rule: autoscaling-group-elb-healthcheck-required

This control checks whether your Auto Scaling groups that are associated with a load balancer are using Elastic Load Balancing health checks.

PCI DSS does not require load balancing or highly available configurations. However, this check aligns with AWS best practices.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 2.2: Develop configuration standards for all system components. Assure that these standards address all known security vulnerabilities and are consistent with industry-accepted system hardening standards.

- Replicating systems using load balancing provides high availability and is a means to mitigate the effects of a DDoS event.

- This is one method used to implement system hardening configurations.

Remediation

To enable Elastic Load Balancing health checks

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. On the navigation pane, under Auto Scaling, choose Auto Scaling Groups.
3. To select the group from the list, choose the right box
4. From Actions, choose Edit
5. For Health Check Type, choose ELB.
6. For Health Check Grace Period, enter 300.
7. Choose Save.

For more information on using a load balancer with an auto scaling group, see the Amazon EC2 Auto Scaling User Guide.

[PCI.CloudTrail.1] CloudTrail logs should be encrypted at rest using AWS KMS CMKs

Severity: Medium

Resource: CloudTrail trail
AWS Config rule: cloud-trail-encryption-enabled

This control checks whether AWS CloudTrail is configured to use the server-side encryption (SSE) AWS KMS customer master key (CMK) encryption.

If you are only using the default encryption option, you can choose to disable this check.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 3.4: Render Primary Account Numbers (PAN) unreadable anywhere it is stored (including on portable digital media, backup media, and in logs).

If you are using AWS services to process and store PAN, your CloudTrail logs should be encrypted at rest to ensure that if logs capture PAN(s), the PAN(s) are protected.

By default, the log files delivered by CloudTrail to your S3 bucket are encrypted using Amazon server-side encryption with Amazon S3-managed encryption keys (SSE-S3). See the Amazon Simple Storage Service Developer Guide.

You can configure CloudTrail logs to leverage AWS KMS customer-created master keys (CMKs) to further protect CloudTrail logs.

These are methods used to render PAN unreadable.

Remediation

To enable encryption for CloudTrail logs

1. Open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. Choose Trails.
3. Choose the trail to update.
4. Under Storage location, to edit the settings, choose the pencil icon.
5. For Encrypt log files with SSE-KMS, choose Yes.
6. For Create a new KMS key, do one of the following:
   - To create a key, choose Yes and then in KMS key, enter an alias for the key. The key is created in the same Region as the S3 bucket.
   - To use an existing key, choose No and then from KMS key, select the key.
   The AWS KMS key and S3 bucket must be in the same Region.
7. Choose Save.

You might need to modify the policy for CloudTrail to successfully interact with your CMK. For more information on encrypting CloudTrail log files with AWS KMS managed keys (SSE-KMS), see the AWS CloudTrail User Guide.

[PCI.CloudTrail.2] CloudTrail should be enabled

Severity: High

Resource: Account

AWS Config rule: cloudtrail-enabled
This control checks whether CloudTrail is enabled in your AWS account.

However, some AWS services do not enable logging of all APIs and events. You should implement any additional audit trails other than CloudTrail and review the documentation for each service in CloudTrail Supported Services and Integrations.

Related PCI DSS requirements

This control is associated with the following PCI DSS requirements:

PCI DSS 10.1: Implement audit trails to link all access to system components to each individual user.

By enabling CloudTrail, Event History provides you with 90 days of readily available events and audit trails for access to system components by each individual user.

You can find the identity of the users in the eventSource section of the CloudTrail log.

PCI DSS 10.2.1: Implement automated audit trails for all system components to reconstruct the following events: All individual user accesses to cardholder data

Depending on where cardholder data is stored, individual user accesses to cardholder data could be found in the userIdentity, eventSource, eventName, or responseElements sections of the CloudTrail log.

PCI DSS 10.2.2: Implement automated audit trails for all system components to reconstruct the following events: All actions taken by any individual with root or administrative privileges

Root user identification is found in the userIdentity section of the log.

PCI DSS 10.2.3: Implement automated audit trails for all system components to reconstruct the following events: Access to all audit trails

Access to audit trails might be found in the eventSource, eventName, or responseElements sections of the log.

PCI DSS 10.2.4: Implement automated audit trails for all system components to reconstruct the following events: Invalid logical access attempts

You can find invalid logical access attempts in CloudTrail logs. For example: responseElements : "ConsoleLogin" and responseElements : "Failure".

PCI DSS 10.2.5: Implement automated audit trails for all system components to reconstruct the following events: Use of and changes to identification and authentication mechanisms—including but not limited to creation of new accounts and elevation of privileges—and all changes, additions, or deletions to accounts with root or administrative privileges

Use of and changes to identification and authentication mechanisms might be found in the userAgent, eventName, or responseElements sections of the log.

PCI DSS 10.2.6: Implement automated audit trails for all system components to reconstruct the following events: Initialization, stopping, or pausing of the audit logs

Starting and stopping logging is captured in the CloudTrail logs.

An example of audit log starting and stopping would look as follows within a CloudTrail Log:
eventName : "StopLogging" and eventName : "StartLogging"

PCI DSS 10.2.7: Implement automated audit trails for all system components to reconstruct the following events: Creation and deletion of system- level objects

Creation and deletion of system level-objects are captured in the CloudTrail logs. An example of a system-level object would be an AWS Lambda function.

CloudTrail captures the createFunction and deleteFunction API calls, as described in the AWS Lambda Developer Guide.
PCI DSS 10.3.1: Record at least the following audit trail entries for all system components for each event: User identification

You can find user identification in the `userIdentity` section of the CloudTrail logs.

PCI DSS 10.3.2: Record at least the following audit trail entries for all system components for each event: Type of event

You can find the type of event in the `eventName` section of the CloudTrail log.

PCI DSS 10.3.3: Record at least the following audit trail entries for all system components for each event: Date and time

You can find the date and time of an event in the `eventTime` section of the CloudTrail log.

PCI DSS 10.3.4: Record at least the following audit trail entries for all system components for each event: Success or failure indication

You can find the success or failure indication in the `responseElements` section of the CloudTrail log.

PCI DSS 10.3.5: Record at least the following audit trail entries for all system components for each event: Origination of event

You can find the origination of an event in the `userAgent` or `sourceIPAddress` section of the CloudTrail log.

PCI DSS 10.3.6: Record at least the following audit trail entries for all system components for each event: Identity or name of affected data, system component, or resource.

You can find the identity of the resource in the `eventSource` section of the CloudTrail log.

Remediation

To create a new trail in CloudTrail

1. Sign in to the AWS Management Console using the IAM user you configured for CloudTrail administration.
2. Open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
3. In the `Region` selector, choose the AWS Region where you want your trail to be created. This is the Home Region for the trail.

The Home Region is the only AWS Region where you can view and update the trail after it is created, even if the trail logs events in all AWS Regions.
4. In the navigation pane, choose Trails.
5. On the Trails page, choose Get Started Now. If you do not see that option, choose Create Trail.
6. In Trail name, give your trail a name, such as `My-Management-Events-Trail`.

As a best practice, use a name that quickly identifies the purpose of the trail. In this case, you’re creating a trail that logs management events.
7. For Apply trail to all regions, keep the default Yes.
8. In Management Events, make sure Read/Write events is set to All.
9. In Data Events, do not make any changes. This trail will not log any data events.
10. Create a new S3 bucket for the logs:

   a. In Storage Location, in Create a new S3 bucket, choose Yes.
   b. In S3 bucket, give your bucket a name, such as `my-bucket-for-storing-cloudtrail-logs`.
The name of your S3 bucket must be globally unique. For more information about S3 bucket naming requirements, see the AWS CloudTrail User Guide.

c. Under Advanced, choose Yes for both Encrypt log files with SSE-KMS and Enable log file validation.

11. Choose Create.

For more details, see the tutorial in the AWS CloudTrail User Guide.

**[PCI.CloudTrail.3] CloudTrail log file validation should be enabled**

**Severity:** Low

**Resource:** CloudTrail trail

**AWS Config rule:** cloud-trail-log-file-validation-enabled

This control checks whether CloudTrail log file validation is enabled. It does not check when configurations are altered.

To monitor and alert on log file changes, you can use CloudWatch Events or CloudWatch Metric Filters.

**Related PCI DSS requirements**

This control is related to the following PCI DSS requirements:

**PCI DSS 10.5.2:** Protect audit trail files from unauthorized modifications.

CloudTrail log file validation creates a digitally signed digest file containing a hash of each log that CloudTrail writes to Amazon S3.

You can use these digest files to determine whether a log file was changed, deleted, or unchanged after CloudTrail delivered the log.

This is a method that helps to protect audit trail files from unauthorized modifications.

**PCI DSS 10.5.5:** Use file-integrity monitoring or change-detection software on logs to ensure that existing log data cannot be changed without generating alerts.

CloudTrail log file validation creates a digitally signed digest file containing a hash of each log that CloudTrail writes to Amazon S3.

You can use these digest files to determine whether a log file was changed, deleted, or unchanged after CloudTrail delivered the log.

This is a method that helps to ensure file-integrity monitoring or change-detection software is used on logs.

**Remediation**

**To enable CloudTrail log file validation**

1. Open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. Choose Trails.
3. In the Name column, choose the name of a trail to edit.
4. Choose the pencil icon for the Storage location.
5. For Enable log file validation, choose Yes.
6. Choose Save.

[PCI.CloudTrail.4] CloudTrail trails should be integrated with CloudWatch Logs

Severity: Low

Resource: CloudTrail trail

AWS Config rule: cloud-trail-cloud-watch-logs-enabled

This control checks whether CloudTrail trails are configured to send logs to CloudWatch Logs.

It does not check for user permissions to alter logs or log groups. You should create specific CloudWatch rules to alert when CloudTrail logs are altered.

This control also does not check for any additional audit log sources other than CloudTrail being sent to a CloudWatch Logs group.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 10.5.3: Promptly back up audit trail files to a centralized log server or media that is difficult to alter.

CloudTrail uses Amazon S3 for log file storage and delivery, so log files are stored permanently.

CloudWatch Logs is a native way to promptly back up audit trail files.

Remediation

To ensure that CloudTrail trails are integrated with CloudWatch Logs

1. Open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. Choose Trails.
3. Choose a trail that there is no value for in the CloudWatch Logs Log group column.
4. Scroll down to the CloudWatch Logs section and then choose Configure.
5. For New or existing log group, do one of the following:
   a. To use the default log group, keep the name as is.
   b. To use an existing log group, enter the name of the log group to use.
   c. To create a new log group, enter a name for the log group to create.
6. Choose Continue.
7. You can either use the default IAM role, or specify a role.

To use the default IAM role, go to the next step.

To specify the role to use:

a. Choose View Details.
b. For IAM role, do one of the following:
• Choose the CloudTrail_CloudWatchLogs_role and then from Policy Name, choose the policy to use.
• Choose Create a new IAM Role and then enter the name of the role to create.

The new role is assigned a policy that grants the necessary permissions.

8. Choose Allow.

For more information about configuring CloudWatch Logs monitoring with the console, see the AWS CloudTrail User Guide.

[PCI.CodeBuild.1] CodeBuild GitHub or Bitbucket source repository URLs should use OAuth

Severity: Critical
Resource: CodeBuild project

AWS Config rule: codebuild-project-source-repo-url-check

This control checks whether the GitHub or Bitbucket source repository URL contains either personal access tokens or a user name and password.

Note
This control is not supported in AWS GovCloud (US-East) or AWS GovCloud (US-West).

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 8.2.1: Using strong cryptography, render all authentication credentials (such as passwords/phrases) unreadable during transmission and storage on all system components.

If you are using CodeBuild in your PCI DSS environment to compile your source code, run unit tests, or produce artifacts that are ready to deploy, authentication credentials should never be stored or transmitted in clear text or appear in the repository URL.

You should use OAuth instead of personal access tokens or a user name and password to grant authorization for accessing GitHub or Bitbucket repositories. This is a method to use strong cryptography to render authentication credentials unreadable.

Remediation

To remove basic authentication / (GitHub) Personal Access Token from CodeBuild Project Source

2. Select your Build project that contains personal access tokens or a user name and password
3. From Edit, choose Source.
4. Choose Disconnect from GitHub / Bitbucket.
5. Choose Connect using OAuth and then choose Connect to GitHub / Bitbucket.
6. In the message displayed by your source provider, authorize as appropriate.
7. Reconfigure your Repository URL and additional configuration settings, as needed.
8. Choose Update source.
To see CodeBuild use case-based samples, see the *AWS CodeBuild User Guide*.

**[PCI.CodeBuild.2] CodeBuild project environment variables should not contain clear text credentials**

**Severity:** Critical

**Resource:** CodeBuild project

**AWS Config rule:** codebuild-project-envvar-awscred-check

This control checks whether the project contains environment variables `AWS_ACCESS_KEY_ID` and `AWS_SECRET_ACCESS_KEY`.

**Note**
This control is not supported in AWS GovCloud (US-East) or AWS GovCloud (US-West).

**Related PCI DSS requirements**
This control is related to the following PCI DSS requirements:

**PCI DSS 8.2.1:** Using strong cryptography, render all authentication credentials (such as passwords/ phrases) unreadable during transmission and storage on all system components.

If you are using CodeBuild in your PCI DSS environment to compile your source code, runs unit tests, or produce artifacts that are ready to deploy, then the authentication credentials `AWS_ACCESS_KEY_ID` and `AWS_SECRET_ACCESS_KEY` should never be stored in clear text.

Using environmental variables to store credentials in your CodeBuild project may violate the requirement to use strong cryptography to render authentication credentials unreadable.

**Remediation**

To reference sensitive data in CodeBuild runtime using Environmental variables, use the following procedures.

**To remove an Environmental Variable**

2. Expand **Build**, choose **Build project**, and then choose the build project that contains plaintext credentials.
3. From **Edit**, choose **Environment**.
4. Expand **Additional configuration** and then scroll to **Environment variables**.
5. Choose **Remove** next to the environment variable.
6. Choose **Update environment**.

**To store sensitive values in the Amazon EC2 Systems Manager Parameter Store and then retrieve them from your build spec**

2. Expand **Build**, choose **Build project**, and then choose your build project that contains plaintext credentials.
3. From **Edit**, choose **Environment**.
4. Expand **Additional configuration** and then scroll to **Environment variables**.
5. In AWS Systems Manager, create a systems manager parameter that contains your sensitive data. For instructions on how to do this, refer to the tutorial in the AWS Systems Manager User Guide.

6. After you create the parameter, copy the parameter name.

7. Back in the CodeBuild console, choose Create environmental variable.

8. For name, enter the name of your variable as it appears in your build spec.

9. For value, paste in the name of your parameter.

10. From type, choose Parameter.

11. Choose Remove next to your non-compliant environmental variable that contains plaintext credentials.

12. Choose Update environment.

See the information on environment variables in build environments in the AWS CodeBuild User Guide.

[PCI.Config.1] AWS Config should be enabled

Severity: Medium

Resource: Account

AWS Config rule: None. To run this check, Security Hub runs through audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

This control checks whether AWS Config is enabled in current account and region.

It does not check for change detection for all critical system files and content files, as AWS Config supports only a subset of resource types.

For more information, see the AWS Config Developer Guide.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 10.5.2: Protect audit trail files from unauthorized modifications.

AWS Config continuously monitors, tracks, and evaluates your AWS resource configurations for desired settings and generates configuration change history files every six hours.

You should enable AWS Config to protect audit trail files from unauthorized modifications.

PCI DSS 11.5: Deploy a change-detection mechanism to alert personnel to unauthorized modification of critical system files, configuration files, or content files; and configure the software to perform critical file comparisons at least weekly.

AWS Config continuously monitors, tracks, and evaluates your AWS resource configurations for desired settings and generates configuration change history files every six hours.

You should enable AWS Config to ensure a change-detection mechanism is deployed and is configured to perform critical file comparisons at least weekly.

Remediation

To configure AWS Config settings

1. Open the AWS Config console at https://console.aws.amazon.com/config/.
2. Choose the Region to configure AWS Config in.
3. If you have not used AWS Config before, choose Get started.
4. On the Settings page, do the following:
   a. Under Resource types to record, choose Record all resources supported in this region and Include global resources (e.g., AWS IAM resources).
   b. Under Amazon S3 bucket, either specify the bucket to use or create a bucket and optionally include a prefix.
   c. Under Amazon SNS topic, either select an Amazon SNS topic from your account or create one. For more information about Amazon SNS, see the Amazon Simple Notification Service Getting Started Guide.
   d. Under AWS Config role, either choose Create AWS Config service-linked role or choose Choose a role from your account and then choose the role to use.
5. Choose Next.
6. On the AWS Config rules page, choose Skip.
7. Choose Confirm.

For more information about using AWS Config from the AWS CLI, see the AWS Config Developer Guide.

You can also use an AWS CloudFormation template to automate this process. For more information, see the AWS CloudFormation User Guide.

[PCI.CW.1] A log metric filter and alarm should exist for usage of the "root" user

Severity: Critical

Resource: Account

AWS Config rule: Security Hub runs through audit steps without creating an AWS Config managed rules in your AWS account for this check.

This control checks for the CloudWatch metric filters using the following pattern:

```java
{ $.userIdentity.type = "Root" && $.userIdentity.invokedBy NOT EXISTS && $.eventType != "AwsServiceEvent" }
```

It checks the following:

- The log group name is configured for use with active multi-region CloudTrail.
- There is at least one Event Selector for a Trail with IncludeManagementEvents set to true and ReadWriteType set to All.
- There is at least one active subscriber to an Amazon SNS topic associated with the alarm.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 7.2.1: Establish an access control system(s) for systems components that restricts access based on a user's need to know, and is set to "deny all" unless specifically allowed. This access control system(s) must include the following: Coverage of all system components.

The root user is the most privileged user in an AWS account and has unrestricted access to all resources in the AWS account.
You should set up log metric filters and alarms in the event that root credentials are used.

You should also ensure that CloudTrail is enabled to keep an audit trail of actions taken by any individual with root or administrative privileges (see the section called "[PCI.CloudTrail.2] CloudTrail should be enabled" (p. 183)). Root user identification would be found in the \texttt{userIdentity} section of the CloudTrail log.

**Remediation**

The steps to remediate this issue include setting up an Amazon SNS topic, a metric filter, and an alarm for the metric filter.

These are the same steps to remediate findings for the section called "3.3 – Ensure a log metric filter and alarm exist for usage of "root" account “ (p. 162).

**To create an Amazon SNS topic**

2. Create an Amazon SNS topic that receives all CIS alarms.

   Create at least one subscriber to the topic.

   For more information about creating Amazon SNS topics, see the Amazon Simple Notification Service Developer Guide.
3. Set up an active CloudTrail trail that applies to all Regions.

   To do this, follow the remediation steps in the section called “2.1 – Ensure CloudTrail is enabled in all Regions” (p. 152).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

2. Choose Logs.
3. Find the log group that you made a note of in the previous procedure and then choose the value in the Metric Filters column.
5. Copy the following pattern and then paste it into the Filter Pattern field.

   ```
   {$.userIdentity.type="Root" && $.userIdentity.invokedBy NOT EXISTS && $.eventType !="AwsServiceEvent"
   ```
6. Choose Assign Metric.
7. (Optional) Update the filter name to a name of your choice.
8. Confirm that the value for Metric Namespace is LogMetrics.

   This ensures that all CIS Benchmark metrics are grouped together.
9. Enter a name in the Metric Name field and then choose Create Filter.
10. Choose Create Alarm.
11. Under Alarm details, enter a Name and Description for the alarm, such as \texttt{CIS-1.1-RootAccountUsage}.
12. Under Actions, for Send notification to, choose Enter list and then enter the name of the topic that you created in the previous procedure.
13. Choose **Create Alarm**.

**[PCI.EC2.1] Amazon EBS snapshots should not be publicly restorable**

*Severity:* Critical

*Resource:* Amazon EC2 volume

*AWS Config rule:* `ebs-snapshot-public-restorable-check`

This control checks whether Amazon Elastic Block Store snapshots are not publicly restorable by everyone, which makes them public. Amazon EBS snapshots should not be publicly restorable by everyone unless you explicitly allow it, to avoid accidental exposure of your company’s sensitive data.

You should also ensure that permission to change Amazon EBS configurations are restricted to authorized AWS accounts only. Learn more about managing Amazon EBS snapshot permissions in the *Amazon EC2 User Guide for Linux Instances*.

**Related PCI DSS requirements**

This control is related to the following PCI DSS requirements:

**PCI DSS 1.2.1: Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.**

Amazon EBS snapshots are used to back up the data on your Amazon EBS volumes to Amazon S3 at a specific point in time, and can be used to restore previous states of EBS volumes.

If an Amazon EBS snapshot stores cardholder data, it should not be publicly restorable by everyone. This would violate the requirement to allow only necessary traffic to and from the CDE.

**PCI DSS 1.3.4: Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.**

Amazon EBS snapshots are used to back up the data on your Amazon EBS volumes to Amazon S3 at a specific point in time, and can be used to restore previous states of EBS volumes.

If an Amazon EBS snapshot stores cardholder data, it should not be publicly restorable by everyone. This would violate the requirement to block unauthorized outbound traffic from the cardholder data environment to the Internet.

**PCI DSS 7.2.1: Establish an access control system(s) for systems components that restricts access based on a user’s need to know, and is set to "deny all" unless specifically allowed. This access control system(s) must include the following: Coverage of all system components.**

Amazon EBS snapshots are used to back up the data on your Amazon EBS volumes to Amazon S3 at a specific point in time, and can be used to restore previous states of Amazon EBS volumes.
If an Amazon EBS snapshot stores cardholder data, it should not be publicly restorable by everyone. This may violate the requirement to ensure access to systems components is restricted to least privilege necessary, or a user’s need to know.

Remediation

To make a public Amazon EBS snapshot private

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. On the navigation pane, choose Snapshots and then choose your public snapshot
3. Choose Actions, then choose Modify permissions
4. Choose Private
5. Optionally, add AWS account numbers for authorized accounts to share your snapshot with
6. Choose Save

For more information about sharing an Amazon EBS snapshot, see the Amazon EC2 User Guide for Linux Instances.

[PCI.EC2.2] VPC default security group should prohibit inbound and outbound traffic

Severity: Medium

Resource: Amazon EC2 security group

AWS Config rule: vpc-default-security-group-closed

This control checks that the default security group of a VPC does not allow inbound or outbound traffic. It does not check for access restrictions for other security groups that are not default, and other VPC configurations.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 1.2.1: Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.

If a service that is in scope for PCI DSS is associated with the default security group, the default rules for the security group will allow all outbound traffic, as well as all inbound traffic from network interfaces (and their associated instances) that are assigned to the same security group.

You should change the default security group rules setting to restrict inbound and outbound traffic, as using the default might violate the requirement to allow only necessary traffic to and from the CDE.

PCI DSS 1.3.4: Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.

If a service that is in scope for PCI DSS is associated with the default security group, the default rules for the security group will allow all outbound traffic, as well as all inbound traffic from network interfaces (and their associated instances) that are assigned to the same security group.

You should change the default security group rules setting to restrict unauthorized inbound and outbound traffic, as using the default may violate the requirement to block unauthorized outbound traffic from the cardholder data environment to the Internet.
PCI DSS 2.1: Always change vendor-supplied defaults and remove or disable unnecessary default accounts before installing a system on the network.

If a service that is in scope for PCI DSS is associated with the default security group, the default rules for the security group will allow all outbound traffic, as well as all inbound traffic from network interfaces (and their associated instances) that are assigned to the same security group.

You should change the default security group rules setting to restrict inbound and outbound traffic, as using the default may violate the requirement to remove or disable unnecessary default accounts.

Remediation

To update the default security group to restrict all access

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. View the default security groups details to see the resources that are assigned to them.
3. Create a set of least-privilege security groups for the resources.
4. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
5. On the Amazon EC2 console, change the security group for the resources that use the default security groups to the least-privilege security group you created.
6. For each default security group, choose the Inbound tab and then delete all of the inbound rules.
7. For each default security group, choose the Outbound tab and then delete all of the outbound rules.

For more information about working with security groups in Amazon VPC, see the Amazon VPC User Guide.

[PCI.EC2.3] Unused EC2 security groups should be removed

Severity: Low

Resource: Amazon EC2 security group

AWS Config rule: ec2-security-group-attached-to-eni

This control will help you maintain an accurate asset inventory of needed security groups in your CDE by checking that security groups are attached to Amazon EC2 instances or to an ENI. A failed finding indicates you may have unused Amazon EC2 security groups.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 2.4: Maintain an inventory of system components that are in scope for PCI DSS.

If a security group is not attached to an Amazon EC2 instance or an elastic network interface (ENI), this is an indication that the resource is no longer in use.

Unless there is a business need to retain them, you should remove unused resources to maintain an accurate inventory of system components.

Remediation

You must perform the following steps for each security group not attached to an ENI.
To delete a security group

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose Security groups.
3. Choose the security group to delete.
5. Choose Yes, Delete.

[PCI.EC2.4] Unused EC2 EIPs should be removed

Severity: Low

Resource: EC2 EIP

AWS Config rule: eip-attached

This control checks whether Elastic IP addresses that are allocated to a VPC are attached to Amazon EC2 instances or in-use elastic network interfaces (ENIs).

A failed finding indicates you may have unused Amazon EC2 EIPs.

This will help you maintain an accurate asset inventory of EIPs in your CDE.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 2.4: Maintain an inventory of system components that are in scope for PCI DSS.

If an EIP is not attached to an Amazon EC2 instance, this is an indication that it is no longer in use.

Unless there is a business need to retain them, you should remove unused resources to maintain an accurate inventory of system components.

Remediation

If you no longer need an Elastic IP address, Security Hub recommends that you release it (the address must not be associated with an instance).

To release an Elastic IP address using the console

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the navigation pane, choose Elastic IPs.
3. Choose the Elastic IP address, choose Actions, and then choose Release addresses.

For more information, see the information on releasing Elastic IP addresses in the Amazon EC2 User Guide for Linux Instances.

[PCI.ES.1] Amazon Elasticsearch Service domains should be in a VPC

Severity: Critical

Resource: Amazon ES domain
AWS Config rule: `elasticsearch-in-vpc-only`

This control checks whether Amazon Elasticsearch Service domains are in a VPC.

It does not evaluate the VPC subnet routing configuration to determine public reachability.

This AWS control also does not check whether the Amazon ES resource-based policy permits public access by other accounts or external entities. You should ensure that Amazon ES domains are not attached to public subnets. See Resource-based policies in the Amazon Elasticsearch Service Developer Guide.

You should also ensure that your VPC is configured according to the recommended best practices. See Security best practices for your VPC in the Amazon VPC User Guide.

**Related PCI DSS requirements**

This control is related to the following PCI DSS requirements:

**PCI DSS 1.2.1: Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.**

If your Amazon ES clusters contain cardholder data, the Amazon ES domains should be placed in a VPC, which enables secure communication between Amazon ES and other services within the VPC without the need for an internet gateway, NAT device, or VPN connection port.

This method is used to allow only necessary traffic to and from the CDE.

**PCI DSS 1.3.1: Implement a DMZ to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.**

If your Amazon ES clusters contain cardholder data, the Amazon ES domains should be placed in a VPC, which enables secure communication between Amazon ES and other services within the VPC without the need for an internet gateway, NAT device, or VPN connection port.

This method is used to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

**PCI DSS 1.3.2: Limit inbound Internet traffic to IP addresses within the DMZ.**

If your Amazon ES clusters contain cardholder data, the Amazon ES domains should be placed in a VPC, which enables secure communication between Amazon ES and other services within the VPC without the need for an internet gateway, NAT device, or VPN connection port.

This method is used to limit inbound Internet traffic to IP addresses within the DMZ.

You can also use a resource-based policy and specify an IP condition for restricting access based on source IP addresses. See the blog post How to control access to your Amazon Elasticsearch Service domain.

**PCI DSS 1.3.4: Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.**

If your Amazon ES clusters contain cardholder data, the Amazon ES domains should be placed in a VPC, which enables secure communication between Amazon ES and other services within the VPC without the need for an internet gateway, NAT device, or VPN connection port.

This method is used to block unauthorized outbound traffic from the cardholder data environment to the Internet.

**PCI DSS 1.3.6: Place system components that store cardholder data (such as a database) in an internal network zone, segregated from the DMZ and other untrusted networks.**

If your Amazon ES clusters contain cardholder data, the Amazon ES domains should be placed in a VPC, which enables secure communication between Amazon ES and other services within the VPC without the need for an internet gateway, NAT device, or VPN connection port.
This method is used to place system components that store cardholder data in an internal network zone, segregated from the DMZ and other untrusted networks.

Remediation

If you create a domain with a public endpoint, you cannot later place it within a VPC. Instead, you must create a new domain and migrate your data.

The reverse is also true. If you create a domain within a VPC, it cannot have a public endpoint. Instead, you must either create another domain or disable this control.

See the information on migrating from public access to VPC access in the Amazon Elasticsearch Service Developer Guide.

[PCI.EC.2] Amazon Elasticsearch Service domains should have encryption at rest enabled

Severity: Medium

Resource: Amazon ES domain

AWS Config rule: elasticsearch-encrypted-at-rest

This control checks whether Amazon ES domains have encryption at rest configuration enabled.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 3.4: Render Primary Account Numbers (PAN) unreadable anywhere it is stored (including on portable digital media, backup media, and in logs).

- If you use Amazon ES to store credit card Primary Account Numbers (PAN), the PAN should be protected by enabling Amazon ES domain encryption at rest.
- If enabled, it encrypts the following aspects of a domain: Indices, automated snapshots, Amazon ES logs, swap files, all other data in the application directory.

This is a method used to render PAN unreadable.

Remediation

By default, domains do not encrypt data at rest, and you cannot configure existing domains to use the feature.

To enable the feature, you must create another domain and migrate your data. For information about creating domains, see the Amazon Elasticsearch Service Developer Guide.

Encryption of data at rest requires Amazon ES 5.1 or later. For more information about encrypting data at rest for Amazon ES, see the Amazon Elasticsearch Service Developer Guide.

[PCI.IAM.1] IAM root user access key should not exist

Severity: Critical

Resource: Account

AWS Config rule: iam-root-access-key-check
This control checks whether user access keys exist for the root user.

**Related PCI DSS requirements**

This control is related to the following PCI DSS requirements:

**PCI DSS 2.1:** Always change vendor-supplied defaults and remove or disable unnecessary default accounts before installing a system on the network.

The root user is the most privileged AWS user. AWS Access Keys provide programmatic access to a given account.

No access keys should be created for the root user, as this may violate the requirement to remove or disable unnecessary default accounts.

**PCI DSS 2.2:** Develop configuration standards for all system components. Assure that these standards address all known security vulnerabilities and are consistent with industry-accepted system hardening standards.

The root user is the most privileged AWS user. AWS Access Keys provide programmatic access to a given account.

No access keys should be created for the root user, as this may violate the requirement to implement system hardening configurations.

**PCI DSS 7.2.1:** Establish an access control system(s) for systems components that restricts access based on a user’s need to know, and is set to “deny all” unless specifically allowed. This access control system(s) must include the following: Coverage of all system components.

The root user is the most privileged AWS user. AWS Access Keys provide programmatic access to a given account.

No access keys should be created for the root user, as this may violate the requirement to ensure access to systems components is restricted to least privilege necessary, or a user’s need to know.

**Remediation**

**To deactivate or delete access keys**

1. Log in to your account using the root credentials.
2. Choose the account name near the top-right corner of the page and then choose **My Security Credentials**.
3. In the pop-up warning, choose **Continue to Security Credentials**.
4. Choose **Access keys (access key ID and secret access key)**.
5. For any existing keys, do one of the following:
   - To prevent the key from being used to authenticate the account, choose **Make Inactive**.
   - To permanently delete the key, choose **Delete** and then choose **Yes**. You can’t recover deleted keys.

**[PCI.IAM.2]** IAM users should not have IAM policies attached

**Severity:** Low

**Resource:** IAM user

**AWS Config rule:** `iam-user-no-policies-check`
This control checks that none of your IAM users have policies attached. IAM users must inherit permissions from IAM groups or roles.

It does not check whether least privileged policies are applied to IAM roles and groups.

**Related PCI DSS requirements**

This control is related to the following PCI DSS requirements:

**PCI DSS 7.2.1:** Establish an access control system(s) for systems components that restricts access based on a user's need to know, and is set to "deny all" unless specifically allowed. This access control system(s) must include the following: Coverage of all system components.

IAM policies are how privileges are granted to users, groups, or roles in AWS.

By default, IAM users, groups, and roles have no access to AWS resources until IAM policies are attached to them.

To manage least privileged access and reduce the complexity of access management for PCI DSS in-scope resources, you should assign IAM policies at the group or role level and not at the user level.

Reducing access management complexity reduces opportunity for a principal to inadvertently receive or retain excessive privileges.

This is a method used to ensure access to systems components that contain cardholder data is restricted to least privilege necessary, or a user's need to know.

**Remediation**

To resolve this issue, do the following:

1. Create an IAM group
2. Assign the policy to the group
3. Add the users to the group

The policy is applied to each user in the group.

**To create an IAM group**

2. Choose Groups and then choose Create New Group.
3. Enter a name for the group to create and then choose Next Step.
4. Select each policy to assign to the group and then choose Next Step.

The policies that you choose should include any policies currently attached directly to a user account. The next step to resolve a failed check is to add users to a group and then assign the policies to that group.

Each user in the group gets assigned the policies assigned to the group.
5. Confirm the details on the Review page and then choose Create Group.

For more information about creating IAM groups, see the IAM User Guide.

**To add users to an IAM group**

2. Choose **Groups**.
3. Choose **Group Actions** and then choose **Add Users to Group**.
4. Choose the users to add to the group and then choose **Add Users**.

For more information about adding users to groups, see the **IAM User Guide**.

**To remove a policy attached directly to a user**

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose **Users**.
3. For the user to detach a policy from, in the **User name** column, choose the name.
4. For each policy listed under **Attached directly**, to remove the policy from the user, choose the X on the right side of the page and then choose **Remove**.
5. Confirm that the user can still use AWS services as expected.

**[PCI.IAM.3] IAM policies should not allow full "*" administrative privileges**

**Severity:** High

**Resource:** IAM policy

**AWS Config rule:** [iam-policy-no-statements-with-admin-access](https://console.aws.amazon.com/iam/)

This control checks whether the default version of AWS Identity and Access Management policies (also known as customer managed policies) do not have administrator access with a statement that has "Effect": "Allow" with "Action": "*" over "Resource": "*".

It only checks for the customer managed policies that you created, but does not check for full access to individual services, such as "S3:*".

It does not check for inline and AWS managed policies.

**Related PCI DSS requirements**

This control is related to the following PCI DSS requirements:

**PCI DSS 7.2.1:** Establish an access control system(s) for systems components that restricts access based on a user’s need to know, and is set to "deny all" unless specifically allowed. This access control system(s) must include the following: Coverage of all system components.

Providing full administrative privileges instead of restricting to the minimum required may violate the requirement to ensure access to systems components is restricted to the least privilege necessary, or a user’s need to know.

**Remediation**

**To modify an IAM policy**

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose **Policies**.
3. Choose the radio button next to the policy to remove.
4. From **Policy actions**, choose **Detach**.
5. On the Detach policy page, choose the radio button next to each user to detach the policy from and then choose Detach policy.

6. Confirm that the user that you detached the policy from can still access AWS services and resources as expected.

[PCI.IAM.4] Hardware MFA should be enabled for the root user

Severity: Critical

Resource: Account

AWS Config rule: root-account-hardware-mfa-enabled

This control checks whether your AWS account is enabled to use multi-factor authentication (MFA) hardware device to sign in with root credentials.

It does not check whether you are using virtual MFA.

To address PCI DSS requirement 8.3.1, you can choose between hardware MFA (this control) or virtual MFA (the section called “[PCI.IAM.5] Virtual MFA should be enabled for the root user” (p. 202)).

Note
- This control is not supported in AWS GovCloud (US-East) or AWS GovCloud (US-West).

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 8.3.1: Incorporate multi-factor authentication for all non-console access into the Cardholder Data Environment (CDE) for personnel with administrative access.

The root user is the most privileged user in an account.

MFA adds an extra layer of protection on top of a user name and password. If users with administrative privileges are accessing the cardholder data environment over a network interface rather than via a direct, physical connection to the system component, and are not physically in front of the machine they are administering, MFA is required.

Enabling hardware MFA is a method used to incorporate multi-factor authentication (MFA) for all non-console administrative access

Remediation

To enable hardware-based MFA for the root account

1. Log in to your account using the root credentials.
2. Choose the account name at the top right of the page and then choose My Security Credentials.
3. In the warning, choose Continue to Security Credentials.
5. Choose Activate MFA.
6. Choose a hardware-based (not virtual) device to use for MFA and then choose Continue.
7. Complete the steps to configure the device type appropriate to your selection.

[PCI.IAM.5] Virtual MFA should be enabled for the root user

Severity: Critical
Resource: Account

AWS Config rule: root-account-mfa-enabled

This control checks whether users of your AWS account require a multi-factor authentication (MFA) device to sign in with root credentials.

It does not check whether you are using hardware MFA.

To address PCI DSS requirement 8.3.1, you can choose between virtual MFA (this control) or hardware MFA (the section called “[PCI.IAM.4] Hardware MFA should be enabled for the root user” (p. 202)).

Note
This control is not supported in AWS GovCloud (US-East) or AWS GovCloud (US-West).

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 8.3.1: Incorporate multi-factor authentication for all non-console access into the Cardholder Data Environment (CDE) for personnel with administrative access.

The root user is the most privileged user in an account.

MFA adds an extra layer of protection on top of a user name and password. If users with administrative privileges are accessing the cardholder data environment, and are not physically in front of the machine they are administering, MFA is required.

Enabling virtual MFA is a method used to incorporate multi-factor authentication (MFA) for all non-console administrative access.

Remediation

To enable MFA for the root account

1. Log in to your account using the root credentials.
2. Choose the account name at the top-right of the page and then choose My Security Credentials.
3. In the warning, choose Continue to Security Credentials.
5. Choose Activate MFA.
6. Choose the type of device to use for MFA and then choose Continue.
7. Complete the steps to configure the device type appropriate to your selection.

[PCI.IAM.6] MFA should be enabled for all IAM users

Severity: Medium

Resource: IAM user

AWS Config rule: iam-user-mfa-enabled

This control checks whether the IAM users have multi-factor authentication (MFA) enabled.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:
PCI DSS 8.3.1: Incorporate multi-factor authentication for all non-console access into the Cardholder Data Environment (CDE) for personnel with administrative access.

   Enabling MFA for all IAM users is a method used to incorporate multi-factor authentication (MFA) for all non-console administrative access.

Remediation

To configure MFA for a user

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Users.
3. Choose the user name of the user to configure MFA for.
4. Choose Security credentials and then choose Manage next to Assigned MFA device.
5. Follow the Manage MFA Device wizard to assign the type of device appropriate for your environment.

To learn how to delegate MFA setup to users, the AWS Security Blog post How to Delegate Management of Multi-Factor Authentication to AWS IAM Users.

[PCI.KMS.1] Customer master key (CMK) rotation should be enabled

Severity: Medium

Resource: AWS KMS key

AWS Config rule: cmk-backing-key-rotation-enabled

This control checks that key rotation is enabled for each customer master key (CMK). It does not check CMKs that have imported key material.

You should ensure keys that have imported material and those that are not stored in AWS KMS are rotated. AWS managed customer master keys are rotated once every 3 years.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 3.6.4: Cryptographic keys should be changed once they have reached the end of their cryptoperiod.

   While PCI DSS does not specify the time frame for cryptoperiods, if key rotation is enabled, rotation will occur annually by default.

   If you use customer master key (CMK) to encrypt cardholder data, you should enable key rotation.

   This is a method used to change cryptographic keys once they have reached the end of their cryptoperiod.

Remediation

To enable CMK rotation

2. To change the AWS Region, use the Region selector in the upper-right corner of the page.
3. Choose Customer managed keys.
4. In the Alias column, choose the alias of the key to update.
5. Choose Key rotation.
6. Select Automatically rotate this CMK every year and then choose Save.

[PCI.Lambda.1] Lambda functions should prohibit public access

Severity: Critical

Resource: Lambda function

AWS Config rule: lambda-function-public-access-prohibited

This control checks whether the Lambda function resource-based policy prohibits public access. It does not check for access to the Lambda function by internal principals, such as IAM roles. You should ensure that access to the Lambda function is restricted to authorized principals only by using least privilege Lambda resource-based policies.

For more information about using resource-based policies for AWS Lambda, see the AWS Lambda Developer Guide.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 1.2.1: Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.

If you use a Lambda function that is in scope for PCI DSS, the function should not be publicly accessible. A publicly accessible function might violate the requirement to allow only necessary traffic to and from the CDE.

PCI DSS 1.3.1: Implement a DMZ to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

If you use a Lambda function that is in scope for PCI DSS, the function should not be publicly accessible. A publicly accessible function might violate the requirement to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

PCI DSS 1.3.2: Limit inbound Internet traffic to IP addresses within the DMZ.

If you use a Lambda function that is in scope for PCI DSS, the function should not be publicly accessible. A publicly accessible function might violate the requirement to limit inbound Internet traffic to IP addresses within the DMZ.

PCI DSS 1.3.4: Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.

If you use a Lambda function that is in scope for PCI DSS, the function must not be publicly accessible. A publicly accessible function might violate the requirement to block unauthorized outbound traffic from the cardholder data environment to the Internet.

PCI DSS 7.2.1: Establish an access control system(s) for systems components that restricts access based on a user's need to know, and is set to "deny all" unless specifically allowed. This access control system(s) must include the following: Coverage of all system components.

If you use a Lambda function that is in scope for PCI DSS, the function should not be publicly accessible. A publicly accessible function might violate the requirement to ensure access to systems
components that contain cardholder data is restricted to the least privilege necessary, or a user’s need to know.

Remediation

To remediate this issue, you update the resource-based policy to change the publicly accessible Lambda function to a private Lambda function.

You can only update resource-based policies for Lambda resources within the scope of the AddPermission and AddLayerVersionPermission API actions.

You cannot author policies for your Lambda resources in JSON, or use conditions that don't map to parameters for those actions using the CLI or the SDK.

To use the AWS CLI to revoke function-use permission from an AWS service or another account

1. To get the ID of the statement from the output of GetPolicy, from the AWS CLI, run the following:

   ```bash
   aws lambda get-policy --function-name yourfunctionname
   ```

   This command returns the Lambda resource-based policy string associated with the publicly accessible Lambda function.

2. From the policy statement returned by the get-policy command, copy the string value of the Sid field.

3. From the AWS CLI, run

   ```bash
   aws lambda remove-permission --function-name yourfunctionname --statement-id youridvalue
   ```

To use the Lambda console to restrict access to the Lambda function

1. Open the AWS Lambda console at https://console.aws.amazon.com/lambda/.

2. Navigate to Functions and then select your publicly accessible Lambda function

3. Under Designer, choose the key icon at the top left. It has the tool-tip View permissions.

4. Under Function policy, if the policy allows actions for the principal element "*" or {"AWS": "*"}, it is publicly accessible.

   Consider adding the following IAM condition to scope access to your account only.

   ```json
   "Condition": {
   "StringEquals": {
   "AWS:SourceAccount": "<account_id>"
   }
   }
   }
   ```

For other Lambda resource-based policies examples that allow you to grant usage permission to other accounts on a per-resource basis, see the information on using resource-based policies for AWS Lambda in the AWS Lambda Developer Guide.

[PCI.Lambda.2] Lambda functions should be in a VPC

Severity: Low
Resource: Lambda function

AWS Config rule: lambda-inside-vpc

This control checks whether a Lambda function is in a VPC.

It does not evaluate the VPC subnet routing configuration to determine public reachability.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 1.2.1: Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.

By default, Lambda runs your functions in a secure default VPC with access to AWS services and the internet.

If you use a Lambda function that is in scope for PCI DSS, the function can be configured to use a VPC endpoint. This would allow you to connect to your Lambda function from within a VPC without internet access, which is a method used to allow only necessary traffic to and from the CDE.

PCI DSS 1.3.1: Implement a DMZ to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

By default, Lambda runs your functions in a secure default VPC with access to AWS services and the internet.

If you use a Lambda function that is in scope for PCI DSS, the function can be configured to use a VPC endpoint. This allows you to connect to your Lambda function from within a VPC without internet access, which is a method used to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

PCI DSS 1.3.2: Limit inbound Internet traffic to IP addresses within the DMZ.

By default, Lambda runs your functions in a secure default VPC with access to AWS services and the internet.

If you use a Lambda function that is in scope for PCI DSS, the function can be configured to use a VPC endpoint. This allows you to connect to your Lambda function from within a VPC without internet access, which is a method used to limit inbound Internet traffic to IP addresses within the DMZ.

PCI DSS 1.3.4: Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.

By default, Lambda runs your functions in a secure default VPC with access to AWS services and the internet.

If you use a Lambda function that is in scope for PCI DSS, the function can be configured to use a VPC endpoint. This allows you to connect to your Lambda function from within a VPC without internet access, which is a method used to block unauthorized outbound traffic from the cardholder data environment to the Internet.

Remediation

To configure a function to connect to private subnets in a virtual private cloud (VPC) in your account

1. Open the AWS Lambda console at https://console.aws.amazon.com/lambda/.
2. Navigate to Functions and then select your Lambda function.
3. Scroll to **Network** and then select a VPC with the connectivity requirements of the function.
4. To run your functions in high availability mode, Security Hub recommends that you choose at least 2 subnets.
5. Choose at least 1 security group that has the connectivity requirements of the function
6. Choose **Save**.

For more information see the section on configuring a Lambda function to access resources in a VPC in the *AWS Lambda Developer Guide*.

**[PCI.RDS.1] RDS snapshots should prohibit public access**

**Severity:** Critical

**Resource:** Amazon RDS DB snapshot

**AWS Config rule:** rds-snapshots-public-prohibited

This control checks whether Amazon RDS DB snapshots prohibit access by other accounts. You should also ensure that access to the snapshot and permission to change Amazon RDS configuration is restricted to authorized principals only.

To learn more about sharing DB snapshots in Amazon RDS, see the *Amazon RDS User Guide*.

Note that if the configuration is changed to allow public access, the AWS Config rule may not be able to detect the change for up to 12 hours. Until the AWS Config rule detects the change, the check passes even though the configuration violates the rule.

**Related PCI DSS requirements**

This control is related to the following PCI DSS requirements:

**PCI DSS 1.2.1:** Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.

- RDS snapshots are used to back up the data on your RDS instances at a specific point in time and can be used to restore previous states of RDS instances.

- If an RDS snapshot stores cardholder data, the RDS snapshot should not be shared by other accounts. Sharing the RDS snapshot would allow other accounts to restore an RDS instance from the snapshot, which may violate the requirement to allow only necessary traffic to and from the CDE.

**PCI DSS 1.3.1:** Implement a DMZ to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

- RDS snapshots are used to back up the data on your RDS instances at a specific point in time and can be used to restore previous states of RDS instances.

- If an RDS snapshot stores cardholder data, the RDS snapshot should not be shared by other accounts. Sharing the RDS snapshot would allow other accounts to restore an RDS instance from the snapshot, which may violate the requirement to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

**PCI DSS 1.3.4:** Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.

- RDS snapshots are used to back up the data on your RDS instances at a specific point in time and can be used to restore previous states of RDS instances.

- If an RDS snapshot stores cardholder data, the RDS snapshot should not be shared by other accounts. Sharing the RDS snapshot would allow other accounts to restore an RDS instance from
the snapshot, which may violate the requirement to block unauthorized outbound traffic from the cardholder data environment to the Internet.

PCI DSS 1.3.6: Place system components that store cardholder data (such as a database) in an internal network zone, segregated from the DMZ and other untrusted networks.

RDS snapshots are used to back up the data on your RDS instances at a specific point in time and can be used to restore previous states of RDS instances.

If an RDS snapshot stores cardholder data, the RDS snapshot should not be shared by other accounts. Sharing the RDS snapshot would allow other accounts to restore an RDS instance from the snapshot, which may violate the requirement to place system components that store cardholder data in an internal network zone, segregated from the DMZ and other untrusted networks.

PCI DSS 7.2.1: Establish an access control system(s) for systems components that restricts access based on a user's need to know, and is set to "deny all" unless specifically allowed. This access control system(s) must include the following: Coverage of all system components.

RDS snapshots are used to back up the data on your RDS instances at a specific point in time and can be used to restore previous states of RDS instances.

If an RDS snapshot stores cardholder data, the RDS snapshot should not be shared by other accounts. Sharing the RDS snapshot would allow other accounts to restore an RDS instance from the snapshot, which may violate the requirement to ensure access to systems components that contain cardholder data is restricted to least privilege necessary, or a user's need to know.

Remediation

To remove public access for Amazon RDS Snapshots

1. Open the Amazon RDS console at https://console.aws.amazon.com/rds/.
2. Navigate to Snapshots and then select the public Snapshot you want to modify
3. From the Actions list, choose Share Snapshots
4. From DB snapshot visibility, choose Private
5. Under DB snapshot visibility, select for all
6. Choose Save

[PCI.RDS.2] RDS DB Instances should prohibit public access

Severity: Critical

Resource: RDS DB instance

AWS Config rule: rds-instance-public-access-check

This control checks whether RDS instances are publicly accessible by evaluating the publiclyAccessible field in the instance configuration item. The value of publiclyAccessible indicates whether the DB instance is publicly accessible. When the DB instance is publicly accessible, it is an Internet-facing instance with a publicly resolvable DNS name, which resolves to a public IP address. When the DB instance isn't publicly accessible, it is an internal instance with a DNS name that resolves to a private IP address.

The control does not check VPC subnet routing settings or the Security Group rules. You should also ensure VPC subnet routing does not allow public access, and that the security group inbound rule associated with the RDS instance does not allow unrestricted access (0.0.0.0/0). You should also ensure that access to your RDS instance configuration is limited to only authorized users by restricting users' IAM permissions to modify RDS instances settings and resources.
For more information, see Hiding a DB instance in a VPC from the Internet in the Amazon RDS User Guide.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 1.2.1: Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.

If you use an RDS instance that is in scope for PCI DSS, the RDS instance should not be publicly accessible, as this might violate the requirement to allow only necessary traffic to and from the CDE.

PCI DSS 1.3.1: Implement a DMZ to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

If you use an RDS instance to store cardholder data, the RDS instance should not be publicly accessible as this might violate the requirement to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

PCI DSS 1.3.2: Limit inbound Internet traffic to IP addresses within the DMZ.

If you use an RDS instance to store cardholder data, the RDS instance should not be publicly accessible as this might violate the requirement to limit inbound Internet traffic to IP addresses within the DMZ.

PCI DSS 1.3.4: Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.

If you use an RDS instance to store cardholder data, the RDS instance should not be publicly accessible, as this might violate the requirement to block unauthorized outbound traffic from the cardholder data environment to the Internet.

PCI DSS 1.3.6: Place system components that store cardholder data (such as a database) in an internal network zone, segregated from the DMZ and other untrusted networks.

If you use an RDS instance to store cardholder data, the RDS instance should not be publicly accessible, as this may violate the requirement to place system components that store cardholder data in an internal network zone, segregated from the DMZ and other untrusted networks.

PCI DSS 7.2.1: Establish an access control system(s) for systems components that restricts access based on a user’s need to know, and is set to “deny all” unless specifically allowed. This access control system(s) must include the following: Coverage of all system components.

If you use an RDS instance to store cardholder data, the RDS instance should not be publicly accessible, as this may violate the requirement to ensure access to systems components that contain cardholder data is restricted to least privilege necessary, or a user’s need to know.

Remediation

To remove public access for Amazon RDS Databases

1. Open the Amazon RDS console at https://console.aws.amazon.com/rds/.
2. Navigate to Databases and then choose your public Database
3. Choose Modify
4. Scroll to Network & Security
5. For Public accessibility, choose No
6. Scroll to the bottom and then choose Continue
7. Under Scheduling of modifications, choose Apply immediately
8. Choose Modify DB Instance
[PCI.Redshift.1] Amazon Redshift clusters should prohibit public access

Severity: Critical

Resource: Amazon Redshift cluster

AWS Config rule: redshift-cluster-public-access-check

This control checks whether Amazon Redshift clusters are publicly accessible by evaluating the publiclyAccessible field in the cluster configuration item.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

- **PCI DSS 1.2.1: Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.**
  
  If you use an Amazon Redshift cluster to store cardholder data, the cluster should not be publicly accessible, as this might violate the requirement to allow only necessary traffic to and from the CDE.

- **PCI DSS 1.3.1: Implement a DMZ to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.**
  
  If you use an Amazon Redshift cluster to store cardholder data, the cluster should not be publicly accessible, as this might violate the requirement to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

- **PCI DSS 1.3.2: Limit inbound Internet traffic to IP addresses within the DMZ.**
  
  If you use an Amazon Redshift cluster to store cardholder data, the cluster should not be publicly accessible, as this may violate the requirement to limit inbound Internet traffic to IP addresses within the DMZ.

- **PCI DSS 1.3.4: Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.**
  
  If you use an Amazon Redshift cluster to store cardholder data, the cluster should not be publicly accessible, as this may violate the requirement to block unauthorized outbound traffic from the cardholder data environment to the Internet.

- **PCI DSS 1.3.6: Place system components that store cardholder data (such as a database) in an internal network zone, segregated from the DMZ and other untrusted networks.**
  
  If you use an Amazon Redshift cluster to store cardholder data, the cluster should not be publicly accessible as this may violate the requirement to place system components that store cardholder data in an internal network zone, segregated from the DMZ and other untrusted networks.

Remediation

**To disable public access for an Amazon Redshift cluster**

1. Open the Amazon Redshift console at https://console.aws.amazon.com/redshift/.
2. On the navigation pane, choose Clusters and then select your public Amazon Redshift cluster
3. From the Cluster drop-down menu, choose Modify cluster
4. In Publicly accessible, choose No
5. Choose Modify

For more information about creating a cluster in a VPC, see the Amazon Redshift Cluster Management Guide.

[PCI.S3.1] S3 buckets should prohibit public write access

Severity: Critical

Resource: S3 bucket

AWS Config rule: s3-bucket-public-write-prohibited

This control checks whether your S3 buckets allow public write access by evaluating the Block Public Access settings, the bucket policy, and the bucket access control list (ACL).

It does not check for write access to the bucket by internal principals, such as IAM roles. You should ensure that access to the bucket is restricted to authorized principals only.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 1.2.1: Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public write access. Allowing public write access might violate the requirement to allow only necessary traffic to and from the CDE.

PCI DSS 1.3.1: Implement a DMZ to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public write access. Allowing public write access might violate the requirement to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

Unless you explicitly require everyone on the internet to be able to write to your S3 bucket, you should ensure that your S3 bucket is not publicly writable.

PCI DSS 1.3.2: Limit inbound Internet traffic to IP addresses within the DMZ.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public write access. Allowing public write access might violate the requirement to limit inbound Internet traffic to IP addresses within the DMZ.

PCI DSS 1.3.4: Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public write access. Allowing public write access might violate the requirement to block unauthorized outbound traffic from the cardholder data environment to the Internet.

PCI DSS 1.3.6: Place system components that store cardholder data (such as a database) in an internal network zone, segregated from the DMZ and other untrusted networks.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public write access. Allowing public write access may violate the requirement to place system components that store cardholder data in an internal network zone, segregated from the DMZ and other untrusted networks.
PCI DSS 7.2.1: Establish an access control system(s) for systems components that restricts access based on a user's need to know, and is set to “deny all” unless specifically allowed. This access control system(s) must include the following: Coverage of all system components.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public write access. Allowing public write access might violate the requirement to ensure access to systems components is restricted to least privilege necessary, or a user's need to know.

Remediation

To remove public access for an S3 bucket

1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
2. Choose the name of the bucket identified in the finding.
3. Choose Permissions and then choose Public access settings.
4. Choose Edit, select all four options, and then choose Save.
5. If prompted, enter confirm and then choose Confirm.

[PCI.S3.2] S3 buckets should prohibit public read access

Severity: Critical

Resource: S3 bucket

AWS Config rule: a3-bucket-public-read-prohibited

This control checks whether your S3 buckets allow public read access by evaluating the Block Public Access settings, the bucket policy, and the bucket access control list (ACL).

Unless you explicitly require everyone on the internet to be able to write to your S3 bucket, you should ensure that your S3 bucket is not publicly writable.

It does not check for read access to the bucket by internal principals, such as IAM roles. You should ensure that access to the bucket is restricted to authorized principals only.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 1.2.1: Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public read access. Public read access might violate the requirement to allow only necessary traffic to and from the CDE.

PCI DSS 1.3.1: Implement a DMZ to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public read access. Public read access might violate the requirement to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.

PCI DSS 1.3.2: Limit inbound Internet traffic to IP addresses within the DMZ.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public read access. Public read access might violate the requirement to limit inbound Internet traffic to IP addresses within the DMZ.
PCI DSS 1.3.6: Place system components that store cardholder data (such as a database) in an internal network zone, segregated from the DMZ and other untrusted networks.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public read access. Public read access might violate the requirement to place system components that store cardholder data in an internal network zone, segregated from the DMZ and other untrusted networks.

PCI DSS 7.2.1: Establish an access control system(s) for systems components that restricts access based on a user’s need to know, and is set to "deny all" unless specifically allowed. This access control system(s) must include the following: Coverage of all system components.

If you use an S3 bucket to store cardholder data, the bucket should prohibit public read access. Public read access might violate the requirement to ensure access to systems components is restricted to least privilege necessary, or a user’s need to know.

Remediation

To remove public access for an S3 bucket

1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
2. Choose the name of the bucket identified in the finding.
3. Choose Permissions and then choose Public access settings.
4. Choose Edit, select all four options, and then choose Save.
5. If prompted, enter confirm and then choose Confirm.

[PCI.S3.3] S3 buckets should have cross-region replication enabled

Severity: Low

Resource: S3 bucket

AWS Config rule: s3-bucket-replication-enabled

This control checks whether S3 buckets have cross-region replication enabled.

PCI DSS does not require data replication or highly available configurations. However, this check aligns with AWS best practices for this control.

In addition to availability, you should consider other systems hardening settings.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 2.2: Develop configuration standards for all system components. Assure that these standards address all known security vulnerabilities and are consistent with industry-accepted system hardening standards.

Enabling cross-region replication on S3 buckets ensures that multiple versions of the data are available in different distinct Regions. This allows you to store data at even greater distances, minimize latency, increase operational efficiency, and protect against DDoS and data corruption events.

This is one method used to implement system hardening configuration.
Remediation

To enable S3 bucket replication
1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
2. Choose the S3 bucket that does not have cross-region replication enabled.
3. Choose Management, then choose Replication
4. Choose Add rule. If versioning is not already enabled, you are prompted to enable it.
5. Choose your source bucket - Entire bucket
6. Choose your destination bucket. If versioning is not already enabled on the destination bucket for your account, you are prompted to enable it.
7. Choose an IAM role. For more information on setting up permissions for replication, see the Amazon Simple Storage Service Developer Guide.
8. Enter a rule name, choose Enabled for the status, then choose Next
9. Choose Save

For more information about replication, see the Amazon Simple Storage Service Developer Guide.

[PCI.S3.4] S3 buckets should have server-side encryption enabled

Severity: Medium

Resource: S3 bucket

AWS Config rule: s3-bucket-server-side-encryption-enabled

This control checks that your Amazon S3 bucket either has Amazon S3 default encryption enabled or that the S3 bucket policy explicitly denies put-object requests without server-side encryption.

When you set default encryption on a bucket, all new objects stored in the bucket are encrypted when they are stored, including clear text PAN data.

Server-side encryption for all of the objects stored in a bucket can also be enforced using a bucket policy. For more information about server-side encryption, see the Amazon Simple Storage Service Developer Guide.

Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

PCI DSS 3.4: Render Primary Account Numbers (PAN) unreadable anywhere it is stored (including on portable digital media, backup media, and in logs).

If you use an S3 bucket to store credit card Primary Account Numbers (PAN), then to render the PAN unreadable, the bucket default encryption should be enabled and/or the S3 bucket policy should explicitly deny put-object requests without server-side encryption.

Remediation

To enable default encryption on an S3 bucket
1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
2. Choose the bucket from the list.
3. Choose Properties.
4. Choose **Default encryption**.
5. For the encryption, choose either **AES-256** or **AWS-KMS**.
   - To use keys that are managed by Amazon S3 for default encryption, choose **AES-256**. For more information about using Amazon S3 server-side encryption to encrypt your data, see the *Amazon Simple Storage Service Developer Guide*.
   - To use keys that are managed by AWS KMS for default encryption, choose **AWS-KMS**, and then choose a master key from the list of the AWS KMS master keys that you have created.

Type the Amazon Resource Name (ARN) of the AWS KMS key to use. You can find the ARN for your AWS KMS key in the IAM console, under **Encryption keys**. Or, you can choose a key name from the drop-down list.

**Important**

If you use the AWS KMS option for your default encryption configuration, you are subject to the RPS (requests per second) limits of AWS KMS. For more information about AWS KMS limits and how to request a limit increase, see the *AWS Key Management Service Developer Guide*.

For more information about creating an AWS KMS key, see the *AWS Key Management Service Developer Guide*.

For more information about using AWS KMS with Amazon S3, see the *Amazon Simple Storage Service Developer Guide*.

When enabling default encryption, you might need to update your bucket policy. For more information about moving from bucket policies to default encryption, see the *Amazon Simple Storage Service Developer Guide*.

6. Choose **Save**.

For more information about default S3 bucket encryption, see the *Amazon Simple Storage Service Console User Guide*.

**[PCI.SSM.1] Amazon EC2 instances managed by Systems Manager should have a patch compliance status of COMPLIANT after a patch installation**

**Severity:** Medium

**Resource:** SSM patch compliance and Amazon EC2 instance

**AWS Config rule:** `ec2-managedinstance-patch-compliance-status-check`

This control checks whether the compliance status of the Amazon EC2 Systems Manager patch compliance is COMPLIANT or NON_COMPLIANT after the patch installation on the instance.

It only checks instances that are managed by AWS Systems Manager Patch Manager.

It does not check whether the patch was applied within the 30-day limit prescribed by PCI DSS requirement 6.2.

It also does not validate whether the patches applied were classified as security patches.

You should create patching groups with the appropriate baseline settings and ensure in-scope systems are managed by those patch groups in Systems Manager. For more information about patch groups, see the *AWS Systems Manager User Guide*. 
Related PCI DSS requirements

This control is related to the following PCI DSS requirements:

**PCI DSS 6.2**: Ensure that all system components and software are protected from known vulnerabilities by installing applicable vendor-supplied security patches. Install critical security patches within one month of release.

Patches released by the vendor for systems that are in-scope for PCI DSS should be tested and validated prior to installation in production environment. Once deployed, security settings and controls should be validated to ensure that deployed patches have not impacted the security of the Card Data Environment (CDE).

If you use Amazon EC2 instances managed by AWS Systems Manager Patch Manager to patch managed instances in your CDE, ensure that the patches are successfully applied. To do this, check that the compliance status of the Amazon EC2 Systems Manager patch compliance is "COMPLIANT". Patch Manager can apply both operating systems and applications, and this is a method used to protect system components and software from known vulnerabilities.

Remediation

**To remediate non-compliant patches**

This rule checks whether the compliance status of the Amazon EC2 Systems Manager patch compliance is COMPLIANT or NON_COMPLIANT. To find out more about patch compliance states, see the [AWS Systems Manager User Guide](https://docs.aws.amazon.com/systems-manager/user-guide/)

2. In the navigation pane, under **Instances & Nodes**, choose **Run Command**.
3. Choose **Run command**.
4. Choose the radio button next to **AWS-RunPatchBaseline** and then change the **Operation** to **Install**.
5. Choose **Choose instances manually** and then choose the non-compliant instance(s).
6. Scroll to the bottom and then choose **Run**.
7. After the command has completed, to monitor the new compliance status of your patched instances, in the navigation pane, choose **Compliance**.

See the [AWS Systems Manager User Guide](https://docs.aws.amazon.com/systems-manager/user-guide/) for more information about the following

- Using Systems Manager documents to patch a managed instance
- Running commands using the Systems Manager Run command

AWS Foundational Security Best Practices standard

The AWS Foundational Security Best Practices standard is a set of controls that detect when your deployed accounts and resources deviate from security best practices.

The standard allows you to continuously evaluate all of your AWS accounts and workloads to quickly identify areas of deviation from best practices. It provides actionable and prescriptive guidance on how to improve and maintain your organization’s security posture.

The controls include best practices from across multiple AWS services. Each control belongs to one of the following categories, which are based on the functions described in the NIST Cybersecurity Framework.
• Identify
• Protect
• Detect
• Recover

Topics
• Required AWS Config resources (p. 218)
• AWS Foundational Security Best Practices controls (p. 219)

Required AWS Config resources

For AWS Security Hub to accurately report findings for all of the AWS Foundational Security Best Practices controls, you must enable the following resources in AWS Config.

Note
In Regions where a control is not available, the corresponding resource is not available in AWS Config.

• ACM Certificate
• Amazon EBS volume
• Application Load Balancer
• Amazon EFS file system
• CloudFront distribution
• CloudTrail trail
• CodeBuild project
• Amazon EC2 instance
• Amazon EC2 security group
• Amazon EC2 volume
• Elastic Load Balancing load balancer
• Elasticsearch domain
• GuardDuty detector
• IAM group
• IAM policy
• IAM role
• IAM user
• AWS KMS key
• Lambda function
• Amazon RDS DB cluster snapshot
• Amazon RDS DB instance
• Amazon RDS snapshot
• Amazon S3 Block Public Access
• S3 bucket
• Systems Manager managed instance inventory
• Systems Manager patch compliance
• Subnet
AWS Foundational Security Best Practices controls

The AWS Foundational Security Best Practices standard contains the following controls. For each control, the information includes the following information.

- The category and subcategory that the control applies to
- The severity
- The applicable resource
- The required AWS Config rule, and any specific parameter values set by AWS Security Hub
- Remediation steps

[ACM.1] Imported ACM certificates should be renewed within 90 days of expiration

**Category:** Protect > Data protection > Encryption of data in transit  
**Severity:** Medium  
**Resource:** ACM certificate  
**AWS Config rule:** `acm-certificate-expiration-check`  
**Parameters:**

- `daysToExpiration`: 90

This control checks whether ACM certificates in your account are marked for expiration within 90 days. It checks both imported certificates and certificates provided by AWS Certificate Manager.

Certificates provided by ACM are automatically renewed. If you're using certificates provided by ACM, you do not need to rotate SSL/TLS certificates. ACM manages certificate renewals for you.

ACM does not automatically renew certificates that you import. You must renew imported certificates manually.

For more information, see Managed renewal in the *AWS Certificate Manager User Guide*.

**Remediation**

ACM provides managed renewal for your Amazon issued SSL/TLS certificates. This includes both public and private certificates issued by using ACM. If possible, ACM renews your certificates automatically with no action required from you. A certificate is eligible for renewal if it is associated with another AWS service, such as Elastic Load Balancing or Amazon CloudFront. It can also be renewed if it has been exported since being issued or last renewed.

If ACM cannot automatically validate one or more domain names in a certificate, ACM notifies the domain owner that the domain must be validated manually. A domain can require manual validation for the following reasons:

- ACM cannot establish an HTTPS connection with the domain.
- The certificate that is returned in the response to the HTTPS requests does not match the one that ACM is renewing.

When a certificate is 45 days from expiration and one or more domain names in the certificate requires manual validation, ACM notifies the domain owner:
By email (for email-validated certificates)

If the certificate was last validated by email, ACM sends to the domain owner an email for each domain name that requires manual validation. To ensure that this email can be received, the domain owner must correctly configure email for each domain.

For more information, see (Optional) Configure email for your domain. The email contains a link that performs the validation. This link expires after 72 hours. If necessary, you can use the ACM console, AWS CLI, or API to request that ACM resend the domain validation email. For more information, see Request a domain validation email for certificate renewal.

Important
Email-validated certificates are automatically renewed up to 825 days after their last manual validation date. After 825 days, to proceed with the renewal, the domain owner or an authorized representative must manually revalidate ownership of the domain. To avoid this issue, Security Hub recommends that you create a new certificate and use DNS validation if possible. If they are properly configured, DNS-validated certificates are revalidated indefinitely.

By notification in your AWS Personal Health Dashboard

ACM sends notifications to your Personal Health Dashboard to notify you that one or more domain names in the certificate require validation before the certificate can be renewed. ACM sends these notifications when your certificate is 45 days, 30 days, 15 days, 7 days, 3 days, and 1 day from expiration. These notifications are informational only.

[CloudTrail.1] CloudTrail should be enabled and configured with at least one multi-Region trail

Category: Identify > Logging
Severity: High
Resource: Account
AWS Config rule: multi-region-cloud-trail-enabled
Parameters:

• readWriteType: ALL

This control checks that there is at least one multi-Region CloudTrail trail.

AWS CloudTrail records AWS API calls for your account and delivers log files to you. The recorded information includes the following information.

• Identity of the API caller
• Time of the API call
• Source IP address of the API caller
• Request parameters
• Response elements returned by the AWS service

CloudTrail provides a history of AWS API calls for an account, including API calls made from the AWS Management Console, AWS SDKs, command-line tools. The history also includes API calls from higher-level AWS services such as AWS CloudFormation.
The AWS API call history produced by CloudTrail enables security analysis, resource change tracking, and compliance auditing. Multi-Region trails also provide the following benefits.

- A multi-Region trail helps to detect unexpected activity occurring in otherwise unused Regions.
- A multi-Region trail ensures that global service event logging is enabled for a trail by default. Global service event logging records events generated by AWS global services.
- For a multi-Region trail, management events for all read and write operations ensure that CloudTrail records management operations on all of an AWS account’s resources.

Remediation

To create a new trail in CloudTrail

1. Open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. If you haven’t used CloudTrail before, choose Get Started Now.
3. Choose Trails and then choose Create trail.
4. Enter a name for the trail.
5. For Apply trail to all regions, choose Yes.
6. Under Storage location, do one of the following:
   a. To create a new S3 bucket for CloudTrail logs, for Create a new S3 bucket, choose Yes, then enter a name for the new S3 bucket.
   b. To use an existing S3 bucket, for Create a new S3 bucket, choose No, then select the S3 bucket to use.
7. Choose Advanced. For Enable log file validation, choose Yes.
8. Choose Create.

To update an existing trail in CloudTrail

1. Open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. Choose Trails.
3. In the Name column, choose the name of the trail.
4. For Trail settings, choose the pencil icon.
5. For Apply trail to all regions, choose Yes, and then choose Save.
6. For Management events, choose the pencil icon.
7. For Read/Write events, choose All, and then choose Save.
8. For Storage location, choose the pencil icon.
9. For Enable log file validation, choose Yes, and then choose Save.

[CloudTrail.2] CloudTrail should have encryption at-rest enabled

Category: Protect > Data protection > Encryption of data at rest

Severity: Medium

Resource: CloudTrail trail

AWS Config rule: cloud-trail-encryption-enabled

Parameters: None
This control checks whether CloudTrail is configured to use the server-side encryption (SSE) AWS Key Management Service customer master key (CMK) encryption. The check passes if the KmsKeyId is defined.

For an added layer of security for your sensitive CloudTrail log files, you should use server-side encryption with AWS KMS–managed keys (SSE-KMS) for your CloudTrail log files for encryption at rest. Note that by default, the log files delivered by CloudTrail to your buckets are encrypted by Amazon server-side encryption with Amazon S3-managed encryption keys (SSE-S3).

Remediation

To enable encryption for CloudTrail logs

1. Open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. Choose Trails.
3. Choose the trail to update.
4. Under Storage location, choose the pencil icon to edit the settings.
5. For Encrypt log files with SSE-KMS, choose Yes.
6. For Create a new KMS key, do one of the following:
   - To create a key, choose Yes and then enter an alias for the key in the KMS key field. The key is created in the same Region as the bucket.
   - To use an existing key, choose No and then select the key from the KMS key list.

   Note
   The AWS KMS key and S3 bucket must be in the same Region.
7. Choose Save.

   You might need to modify the policy for CloudTrail to successfully interact with your CMK. For more information, see Encrypting CloudTrail log files with AWS KMS–managed keys (SSE-KMS) in the AWS CloudTrail User Guide.

[CodeBuild.1] CodeBuild GitHub or Bitbucket source repository URLs should use OAuth

Category: Protect > Secure development

Severity: Critical

Resource: CodeBuild project

AWS Config rule: codebuild-project-source-repo-url-check

Parameters: None

This control checks whether the GitHub or Bitbucket source repository URL contains either personal access tokens or a user name and password.

Note
This control is not supported in AWS GovCloud (US-East) or AWS GovCloud (US-West).

Authentication credentials should never be stored or transmitted in clear text or appear in the repository URL. Instead of personal access tokens or user name and password, you should use OAuth to grant authorization for accessing GitHub or Bitbucket repositories. Using personal access tokens or a user name and password could expose your credentials to unintended data exposure and unauthorized access.
Remediation

To remove basic authentication / (GitHub) Personal Access Token from CodeBuild project source

2. Choose the build project that contains personal access tokens or a user name and password.
3. From Edit, choose Source.
4. Choose Disconnect from GitHub / Bitbucket.
5. Choose Connect using OAuth, then choose Connect to GitHub / Bitbucket.
6. When prompted, choose authorize as appropriate.
7. Reconfigure your repository URL and additional configuration settings, as needed.
8. Choose Update source.

For more information, refer to CodeBuild use case-based samples in the AWS CodeBuild User Guide.

[CodeBuild.2] CodeBuild project environment variables should not contain clear text credentials

Category: Protect > Secure development

Severity: Critical

Resource: CodeBuild project

AWS Config rule: codebuild-project-envvar-awscrd-check

Parameters: None

This control checks whether the project contains the environment variables AWS_ACCESS_KEY_ID and AWS_SECRET_ACCESS_KEY.

Authentication credentials AWS_ACCESS_KEY_ID and AWS_SECRET_ACCESS_KEY should never be stored in clear text, as this could lead to unintended data exposure and unauthorized access.

Note
This control is not supported in AWS GovCloud (US-East) or AWS GovCloud (US-West).

Remediation

To remove your environmental variable

2. Expand Build.
3. Choose Build project, and then choose the build project that contains plaintext credentials.
4. From Edit, choose Environment.
5. Expand Additional configuration.
6. Choose Remove next to the environment variables.
7. Choose Update environment.

To store sensitive values in the Amazon EC2 Systems Manager Parameter Store and then retrieve them from your build spec

2. Expand **Build**.
3. Choose **Build project**, and then choose the build project that contains plaintext credentials.
4. From **Edit**, choose **Environment**.
5. Expand **Additional configuration** and scroll to **Environment variables**.
6. Follow this tutorial to create a Systems Manager parameter that contains your sensitive data.
7. After you create the parameter, copy the parameter name.
8. Back in the CodeBuild Console, choose **Create environmental variable**.
9. Enter the name of your variable as it appears in your build spec.
10. For **Value**, paste the name of your parameter.
11. For **Type**, choose **Parameter**.
12. To remove your noncompliant environmental variable that contains plaintext credentials, choose **Remove**.
13. Choose **Update environment**.

For more information, see **Environment variables in build environments** in the *AWS CodeBuild User Guide*.

**[Config.1] AWS Config should be enabled**

**Category:** Identify > Inventory  
**Severity:** Medium  
**Resource:** Account  
**AWS Config rule:** None  
**Parameters:** None

This control checks whether AWS Config is enabled in the account for the local Region and is recording all resources.

The AWS Config service performs configuration management of supported AWS resources in your account and delivers log files to you. The recorded information includes the configuration item (AWS resource), relationships between configuration items, and any configuration changes between resources.

Security Hub recommends that you enable AWS Config in all Regions. The AWS configuration item history that AWS Config captures enables security analysis, resource change tracking, and compliance auditing.

**Note**  
Because Security Hub is a Regional service, the check performed for this control checks only the current Region for the account. It does not check all Regions. To allow security checks against global resources in each Region, you also must record global resources.

To learn more, see **Getting started with AWS Config** in the *AWS Config Developer Guide*.

**Remediation**

**To configure AWS Config settings**

2. Choose the Region to configure AWS Config in.
3. If you have not used AWS Config before, choose **Get started**.
4. On the **Settings** page, do the following:
   a. Under **Resource types to record**, choose **Record all resources supported in this region** and **Include global resources (e.g. AWS IAM resources)**.
   b. Under **Amazon S3 bucket**, specify the bucket to use or create a bucket and optionally include a prefix.
   c. Under **Amazon SNS topic**, choose an Amazon SNS topic from your account or create one. For more information about Amazon SNS, see the [Amazon Simple Notification Service Getting Started Guide](https://docs.aws.amazon.com/sns/latest/dg/).
   d. Under **AWS Config role**, either choose **Create AWS Config service-linked role** or **Choose a role from your account** and then choose the role to use.
5. Choose **Next**.
6. On the **AWS Config rules** page, choose **Skip**.
7. Choose **Confirm**.

For more information about using AWS Config from the AWS CLI, see [Turning on AWS Config](https://docs.aws.amazon.com/config/latest/developerguide/toggle-config.html) in the *AWS Config Developer Guide*.

You can also use an AWS CloudFormation template to automate this process. For more information, see the [AWS CloudFormation StackSets sample template](https://docs.aws.amazon.com/cloudformation/latest/userguide/template-aws.html) in the *AWS CloudFormation User Guide*.

**[EC2.1] Amazon EBS snapshots should not be public, determined by the ability to be restorable by anyone**

**Category:** Protect > Secure network configuration  
**Severity:** Critical  
**Resource:** Account

**AWS Config rule:** `ebs-snapshot-public-restorable-check`

**Parameters:** None

This control checks that Amazon Elastic Block Store snapshots are not public, as determined by the ability to be restorable by anyone.

EBS snapshots are used to back up the data on your EBS volumes to Amazon S3 at a specific point in time. You can use the snapshots to restore previous states of EBS volumes. EBS snapshots should not be publicly restorable by everyone unless you explicitly allow it. Restricting this ability avoids accidental exposure of your company's sensitive data.

**Remediation**

**To make a public EBS snapshot private**

1. Open the Amazon EC2 console at [https://console.aws.amazon.com/ec2/](https://console.aws.amazon.com/ec2/).
2. On the navigation pane, choose **Snapshots** menu and then choose your public snapshot.
3. From **Actions**, choose **Modify permissions**.
4. Choose **Private**.
5. Optionally, add the AWS account numbers of the authorized accounts to share your snapshot with.
6. Choose **Save**.
[EC2.2] The VPC default security group should not allow inbound and outbound traffic

**Category:** Protect > Secure network configuration  
**Severity:** Medium  
**Resource:** EC2 security group  
**AWS Config rule:** vpc-default-security-group-closed  
**Parameters:** None

This control checks that the default security group of a VPC does not allow inbound or outbound traffic. The rules for the default security group allow all outbound and inbound traffic from network interfaces (and their associated instances) that are assigned to the same security group.

We do not recommend using the default security group. Because the default security group cannot be deleted, you should change the default security group rules setting to restrict inbound and outbound traffic. This prevents unintended traffic if the default security group is accidentally configured for resources such as EC2 instances.

**Remediation**

**To update the default security group to restrict all access**

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. View the default security groups details to see the resources that are assigned to them.
3. Create a set of least-privilege security groups for the resources.
4. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
5. On the Amazon EC2 console, change the security group for the resources that use the default security groups to the least-privilege security group you created.
6. For each default security group, choose the **Inbound** tab and then delete all inbound rules.
7. For each default security group, choose the **Outbound** tab and then delete all outbound rules.

For more information, see Working with security groups in the Amazon VPC User Guide.

[EC2.3] Attached EBS volumes should be encrypted at-rest

**Category:** Protect > Data protection > Encryption of data at rest  
**Severity:** Medium  
**Resource:** EC2 volume  
**AWS Config rule:** encrypted-volumes  
**Parameters:** None

This control checks whether the EBS volumes that are in an attached state are encrypted. To pass this check, EBS volumes must be in use and encrypted. If the EBS volume is not attached, then it is not subject to this check.

For an added layer of security of your sensitive data in EBS volumes, you should enable EBS encryption at rest. Amazon EBS encryption offers a straightforward encryption solution for your EBS resources that
doesn't require you to build, maintain, and secure your own key management infrastructure. It uses AWS KMS customer master keys (CMK) when creating encrypted volumes and snapshots.

To learn more about Amazon EBS encryption, see Amazon EBS encryption in the Amazon EC2 User Guide for Linux Instances.

Remediation

There is no direct way to encrypt an existing unencrypted volume or snapshot. You can only encrypt a new volume or snapshot when you create it.

If you enabled encryption by default, Amazon EBS encrypts the resulting new volume or snapshot using your default key for Amazon EBS encryption. Even if you have not enabled encryption by default, you can enable encryption when you create an individual volume or snapshot. In both cases, you can override the default key for Amazon EBS encryption and choose a symmetric customer managed CMK.

For more information, see Creating an Amazon EBS volume and Copying an Amazon EBS snapshot in the Amazon EC2 User Guide for Linux Instances.

[EFS.1] Amazon EFS should be configured to encrypt file data at-rest using AWS KMS

Category: Protect > Data protection > Encryption of data at rest

Severity: Medium

Resource: EFS file system

AWS Config rule: efs-encrypted-check

Parameters: None

This control checks whether Amazon Elastic File System is configured to encrypt the file data using AWS KMS. The check fails in the following cases.

- Encrypted is set to false in the DescribeFileSystems response.
- The KmsKeyId key in the DescribeFileSystems response does not match the KmsKeyId parameter for efs-encrypted-check.

Note that this control does not use the KmsKeyId parameter for efs-encrypted-check. It only checks the value of Encrypted.

For an added layer of security for your sensitive data in Amazon EFS, you should create encrypted file systems. Amazon EFS supports encryption for file systems at-rest. You can enable encryption of data at-rest when you create an Amazon EFS file system. To learn more about Amazon EFS encryption, see Data encryption in Amazon EFS in the Amazon Elastic File System User Guide.

Remediation

For details on how to encrypt a new Amazon EFS file system, see Encrypting data at rest in the Amazon Elastic File System User Guide.

[ELBv2.1] Application Load Balancer should be configured to redirect all HTTP requests to HTTPS

Category: Protect > Data protection > Encryption of data in transit
Severity: Medium

Resource: Elbv2 load balancer

AWS Config rule: alb-http-to-https-redirection-check

Parameters: None

This control checks whether HTTP to HTTPS redirection is configured on all HTTP listeners of Application Load Balancers. The check fails if one or more HTTP listeners of Application Load Balancers do not have HTTP to HTTPS redirection configured.

Before you start to use your Application Load Balancer, you must add one or more listeners. A listener is a process that uses the configured protocol and port to check for connection requests. Listeners support both HTTP and HTTPS protocols. You can use an HTTPS listener to offload the work of encryption and decryption to your Application Load Balancer. You should use redirect actions with Application Load Balancer to redirect client HTTP request to an HTTPS request on port 443 to enforce encryption in-transit.

To learn more, see Listeners for your Application Load Balancers in User Guide for Application Load Balancers.

Remediation

To redirect HTTP requests to HTTPS on an Application Load Balancer

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the navigation pane, choose Load balancers.
3. Choose an Application Load Balancer.
4. Choose the Listeners tab.
5. Choose an HTTP listener (port 80 TCP) and then choose Edit.
6. If there is an existing rule, you must delete it. Otherwise, choose Add action and then choose Redirect to...
7. Choose HTTPS and then enter 443.
8. Choose the check mark in a circle symbol and then choose Update.

[ES.1] Elasticsearch domains should have encryption at-rest enabled

Category: Protect > Data protection > Encryption of data at rest

Severity: Medium

Resource: Elasticsearch domain

AWS Config rule: elasticsearch-encrypted-at-rest

Parameters: None

This control checks whether Amazon Elasticsearch Service (Amazon ES) domains have encryption at rest configuration enabled. The check fails if encryption at rest is not enabled.

For an added layer of security for your sensitive data in Elasticsearch, you should configure your Elasticsearch to be encrypted at rest. Elasticsearch domains offer encryption of data at rest. The feature uses AWS KMS to store and manage your encryption keys. To perform the encryption, it uses the Advanced Encryption Standard algorithm with 256-bit keys (AES-256).
To learn more about Elasticsearch encryption at-rest, see Encryption of data at rest for Amazon Elasticsearch Service in the Amazon Elasticsearch Service Developer Guide.

**Note**
Certain instance types, such as t.small and t.medium, do not support encryption of data at rest. For details, see Supported instance types in the Amazon Elasticsearch Service Developer Guide.

**Remediation**

By default, domains do not encrypt data at rest, and you cannot configure existing domains to use the feature.

To enable the feature, you must create another domain and migrate your data. For information about creating domains, see the Amazon Elasticsearch Service Developer Guide.

Encryption of data at rest requires Amazon ES 5.1 or later. For more information about encrypting data at rest for Amazon ES, see the Amazon Elasticsearch Service Developer Guide.

**[GuardDuty.1] GuardDuty should be enabled**

**Category:** Detect > Detection services  
**Severity:** Medium  
**Resource:** Account  
**AWS Config rule:** guardduty-enabled-centralized  
**Parameters:** None

This control checks whether Amazon GuardDuty is enabled in your GuardDuty account and Region. It is highly recommended that you enable GuardDuty in all supported AWS Regions. Doing so allows GuardDuty to generate findings about unauthorized or unusual activity, even in Regions that you do not actively use. This also allows GuardDuty to monitor CloudTrail events for global AWS services such as IAM.

**Note**
This control is not supported in the following Regions.

- Asia Pacific (Hong Kong)
- Middle East (Bahrain)
- AWS GovCloud (US-East)
- AWS GovCloud (US-West)

**Remediation**

**To enable GuardDuty**

2. Choose Get Started.
3. Choose Enable GuardDuty.

**[IAM.1] IAM policies should not allow full "*" administrative privileges**

**Category:** Protect > Secure access management
Severity: High

Resource: IAM policy

AWS Config rule: iam-policy-no-statements-with-admin-access

Parameters: None

This control checks whether the default version of IAM policies (also known as customer managed policies) has administrator access that includes a statement with "Effect": "Allow" with "Action": "*" over "Resource": "*".

The control only checks the customer managed policies that you create. It does not check inline and AWS managed policies.

IAM policies define a set of privileges that are granted to users, groups, or roles. Following standard security advice, AWS recommends that you grant least privilege, which means to grant only the permissions that are required to perform a task. When you provide full administrative privileges instead of the minimum set of permissions that the user needs, you expose the resources to potentially unwanted actions.

Instead of allowing full administrative privileges, determine what users need to do and then craft policies that let the users perform only those tasks. It is more secure to start with a minimum set of permissions and grant additional permissions as necessary. Do not start with permissions that are too lenient and then try to tighten them later.

You should remove IAM policies that have a statement with "Effect": "Allow" with "Action": "*" over "Resource": "*".

Remediation

To modify an IAM policy

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Policies.
3. Choose the button next to the policy to remove.
4. From Policy actions, choose Detach.
5. For each user to detach the policy from, choose the button next to the user, then choose Detach policy.

Confirm that the user that you detached the policy from can still access AWS services and resources as expected.

[IAM.2] IAM users should not have IAM policies attached

Category: Protect > Secure access management

Severity: Low

Resource: IAM user

AWS Config rule: iam-user-no-policies-check

Parameters: None

This control checks that none of your IAM users have policies attached. Instead, IAM users must inherit permissions from IAM groups or roles.
By default, IAM users, groups, and roles have no access to AWS resources. IAM policies grant privileges to users, groups, or roles. We recommend that you apply IAM policies directly to groups and roles but not to users. Assigning privileges at the group or role level reduces the complexity of access management as the number of users grows. Reducing access management complexity might in turn reduce the opportunity for a principal to inadvertently receive or retain excessive privileges.

**Remediation**

To resolve this issue, create an IAM group, assign the policy to the group, and then add the users to the group. The policy is applied to each user in the group.

**To create an IAM group**

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose **Groups** and then choose **Create New Group**.
3. Enter a name for the group to create and then choose **Next Step**.
4. Select each policy to assign to the group and then choose **Next Step**. The policies that you choose should include any policies currently attached directly to a user account.
5. Add users to a group and then assign the policies to that group. Each user in the group is then assigned the policies that are assigned to the group.
6. Confirm the details on the **Review** page and then choose **Create Group**.

For more information about creating groups, see Creating IAM groups in the IAM User Guide.

**To add users to an IAM group**

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose **Groups**.
3. Choose **Group Actions** and then choose **Add Users to Group**.
4. Select the users to add to the group and then choose **Add Users**.

For more information about adding users to groups, see Adding and removing users in an IAM group in the IAM User Guide.

**To remove a policy attached directly to a user**

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose **Users**.
3. For the user to detach a policy from, choose the name in the **User name** column.
4. For each policy listed under **Attached directly**, choose the X on the right side of the page to remove the policy from the user and then choose **Remove**.
5. Confirm that the user can still use AWS services as expected.

**[IAM.3] IAM users access keys should be rotated every 90 days or less**

**Category:** Protect > Secure access management

**Severity:** Medium

**Resource:** IAM user
AWS Config rule: `access-keys-rotated`

Parameters:

- `maxAccessKeyAge`: 90

This control checks whether the active access keys are rotated within 90 days.

We highly recommend that you do not generate and remove all access keys in your account. Instead, the recommended best practice is to either create one or more IAM roles or to use federation. You can use these methods to allow your users to use their existing corporate credentials to log into the AWS Management Console and AWS CLI.

Each approach has its use cases. Federation is generally better for enterprises that have an existing central directory or plan to need more than the current limit IAM users. Applications that run outside of an AWS environment need access keys for programmatic access to AWS resources.

However, if the resources that need programmatic access run inside AWS, the best practice is to use IAM roles. Roles allow you to grant a resource access without hardcoding an access key ID and secret access key into the configuration.

To learn more about protecting your access keys and account, see Best practices for managing AWS access keys in the AWS General Reference. Also see the blog post Guidelines for protecting your AWS account while using programmatic access.

If you already have an access key, Security Hub recommends that you rotate the access keys every 90 days. Rotating access keys reduces the chance that an access key that is associated with a compromised or terminated account is used. It also ensures that data cannot be accessed with an old key that might have been lost, cracked, or stolen. Always update your applications after you rotate access keys.

Access keys consist of an access key ID and a secret access key. They are used to sign programmatic requests that you make to AWS. AWS users need their own access keys to make programmatic calls to AWS from the AWS CLI, Tools for Windows PowerShell, the AWS SDKs, or direct HTTP calls using the APIs for individual AWS services.

If your organization uses AWS Single Sign-On (AWS SSO), your users can sign in to Active Directory, a built-in AWS SSO directory, or another iDP connected to AWS SSO. They can then be mapped to an IAM role that enables them to run AWS CLI commands or call AWS APIs without the need for IAM user access keys. To learn more, see Configuring the AWS CLI to use AWS Single Sign-On in the AWS Command Line Interface User Guide.

**Remediation**

**To ensure that access keys aren’t more than 90 days old**

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Users.
3. For each user that shows an Access key age that is greater than 90 days, choose the User name to open the settings for that user.
5. Create a new key for the user:
   a. Choose Create access key.
   b. To save the key content, either download the secret access key, or choose Show and then copy it from the page.
   c. Store the key in a secure location to provide to the user.
   d. Choose Close.
6. Update all applications that were using the previous key to use the new key.
7. For the previous key, choose **Make inactive** to make the access key inactive. The user now cannot use that key to make requests.
8. Confirm that all applications work as expected with the new key.
9. After confirming that all applications work with the new key, delete the previous key. After you delete the access key, you cannot recover it.

   To delete the previous key, choose the X at the end of the row and then choose **Delete**.

**[IAM.4] IAM root user access key should not exist**

*Category:* Protect > Secure access management  
*Severity:* Critical  
*Resource:* Account  

**AWS Config rule:** iam-root-access-key-check  
*Parameters:* None  

This control checks whether the root user access key is available.

The root account is the most privileged user in an AWS account. AWS access keys provide programmatic access to a given account.

Security Hub recommends that you remove all access keys that are associated with the root account. This limits the vectors that can be used to compromise the account. It also encourages the creation and use of role-based accounts that are least privileged.

**Remediation**

**To deactivate or delete access keys**

1. Log in to your account using the AWS account root user credentials.
2. Choose the account name near the top-right corner of the page and then choose **My Security Credentials**.
3. In the pop-up warning, choose **Continue to Security Credentials**.
4. Choose **Access keys (access key ID and secret access key)**.
5. For any existing keys, do one of the following:
   - To prevent the key from being used to authenticate the account, choose **Make Inactive**.
   - To permanently delete the key, choose **Delete** and then choose **Yes**. You cannot recover deleted keys.

**[IAM.5] MFA should be enabled for all IAM users that have a console password**

*Category:* Protect > Secure access management  
*Severity:* Medium  
*Resource:* IAM user  

**AWS Config rule:** mfa-enabled-for-iam-console-access
**Parameters:** None

This control checks whether AWS Multi-Factor Authentication (MFA) is enabled for all IAM users that use a console password.

Multi-factor authentication (MFA) adds an extra layer of protection on top of a user name and password. With MFA enabled, when a user signs in to an AWS website, they are prompted for their user name and password. In addition, they are prompted for an authentication code from their AWS MFA device.

We recommend that you enable MFA for all accounts that have a console password. MFA is designed to provide increased security for console access. The authenticating principal must possess a device that emits a time-sensitive key and must have knowledge of a credential.

**Remediation**

**To configure MFA for a user**

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Users.
3. Choose the **User name** of the user to configure MFA for.
4. Choose **Security credentials**.
5. Next to **Assigned MFA Device**, choose **Manage**.
6. Follow the **Manage MFA Device** wizard to assign the type of device appropriate for your environment.

To learn how to delegate MFA setup to users, see the blog post How to delegate management of multi-factor authentication to AWS IAM users.

**[IAM.6] Hardware MFA should be enabled for the root user**

**Category:** Protect > Secure access management

**Severity:** Critical

**Resource:** Account

**AWS Config rule:** root-account-hardware-mfa-enabled

**Parameters:** None

This control checks whether your AWS account is enabled to use a hardware multi-factor authentication (MFA) device to sign in with root user credentials.

Virtual MFA might not provide the same level of security as hardware MFA devices. We recommend that you use only a virtual MFA device while you wait for hardware purchase approval or for your hardware to arrive. To learn more, see Enabling a virtual multi-factor authentication (MFA) device (console) in the *IAM User Guide*.

**Note**

This control is not supported in AWS GovCloud (US-East) or AWS GovCloud (US-West).

**Remediation**

**To enable hardware-based MFA for the root account**

1. Log in to your account using the root user credentials.
2. Choose the account name near the top-right corner of the page and then choose **My Security Credentials**.
3. In the pop-up warning, choose **Continue to Security Credentials**.
4. Choose **Multi-Factor Authentication (MFA)**.
5. Choose **Activate MFA**.
6. Choose a hardware-based (not virtual) device to use for MFA and then choose **Continue**.
7. Complete the steps to configure the device type appropriate to your selection.

**[IAM.7]** Password policies for IAM users should have strong configurations

**Category:** Protect > Secure access management

**Severity:** Medium

**Resource:** Account

**AWS Config rule:** `iam-password-policy`

**Parameters:**

- `RequireUppercaseCharacters: true`
- `RequireLowercaseCharacters: true`
- `RequireSymbols: true`
- `RequireNumbers: true`
- `MinimumPasswordLength: 14 or greater`
- `PasswordReusePrevention: 24`
- `MaxPasswordAge: 90`

This control checks whether the account password policy for IAM users uses the following recommended configurations.

- `RequireUppercaseCharacters: true`
- `RequireLowercaseCharacters: true`
- `RequireSymbols: true`
- `RequireNumbers: true`
- `MinimumPasswordLength: 14 or greater`
- `PasswordReusePrevention: 24`
- `MaxPasswordAge: 90`

We highly recommend that you do not generate and remove all access keys in your account. Instead, the recommended best practice is to either create one or more IAM roles, or to use federation. You can use these methods to allow your users to use their existing corporate credentials to log into the AWS Management Console and AWS CLI.

Each approach has its use cases. Federation is generally better for enterprises that have an existing central directory or plan to need more than the current limit IAM users. Applications running outside of an AWS environment need access keys for programmatic access to AWS resources.

However, if the resources that need programmatic access run inside AWS, the best practice is to use IAM roles. Roles allow you to grant a resource access without hardcoding an access key ID and secret access key into the configuration.
To learn more about protecting your access keys and account, see Best practices for managing AWS access keys in the AWS General Reference. Also see the blog post Guidelines for protecting your AWS account while using programmatic access.

If you already have an access key, Security Hub recommends that you enforce the creation of strong user passwords. When you create or change a password policy, the change is enforced immediately for new users. It does not require existing users to change their passwords.

Remediation

To modify the password policy

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. Choose Account settings.
3. Select Prevent password reuse. For Number of passwords to remember, enter 24.
4. Select Requires at least one uppercase letter.
5. Select Requires at least one lowercase letter.
6. Select Requires at least one non-alphanumeric character.
7. Select Requires at least one number.
8. For Minimum password length, enter 14.
9. Choose Enable password expiration. For Password expiration period (in days), enter 90.
10. Choose Apply password policy.

[Lambda.1] Lambda functions should prohibit public access by other accounts

Category: Protect > Secure network configuration

Severity: Critical

Resource: Lambda function

AWS Config rule: lambda-function-public-access-prohibited

Parameters: None

This control checks whether the Lambda function resource-based policy prohibits public access outside of your account.

The Lambda function should not be publicly accessible, as this may allow unintended access to your code stored in the function.

Remediation

You can only update resource-based policies for Lambda resources within the scope of the AddPermission and AddLayerVersionPermission API operations. You cannot author policies for your Lambda resources in JSON. Neither can you use conditions that don't map to parameters for those actions using the AWS CLI or the SDK. The following steps use the AWS CLI.

To create a private Lambda function

1. Get the ID of the statement from the output of GetPolicy.
   a. From the AWS CLI, run aws lambda get-policy --function-name yourfunctionname. This command returns the Lambda resource-based policy string associated with the publicly accessible Lambda.
b. Copy the string value of the Sid field in the policy statement.

2. From the AWS CLI, run `aws lambda remove-permission --function-name yourfunctionname --statement-id youridvalue`

[Lambda.2] Lambda functions should use latest runtimes

**Category:** Protect > Secure development

**Severity:** Medium

**Resource:** Lambda function

**AWS Config rule:** [lambda-function-settings-check](https://docs.aws.amazon.com/config/latest/developerguide/config rules-lambda-function-settings-check.html)

**Parameters:**

- `runtime`: nodejs12.x, nodejs10.x, python3.8, python3.7, python3.6, python2.7, ruby2.5, ruby2.7, java11, java8, go1.x, dotnetcore2.1, dotnetcore3.1

This control checks that the Lambda function settings for runtimes match the expected values set for the latest runtimes for each supported language. This control checks for the following runtimes: nodejs12.x, nodejs10.x, python3.8, python3.7, python3.6, python2.7, ruby2.5, ruby2.7, java11, java8, go1.x, dotnetcore2.1, dotnetcore3.1

Lambda runtimes are built around a combination of operating system, programming language, and software libraries that are subject to maintenance and security updates. When a runtime component is no longer supported for security updates, Lambda deprecates the runtime. Even though you cannot create functions that use the deprecated runtime, the function is still available to process invocation events. Make sure that your Lambda functions are current and do not use out-of-date runtime environments.

To learn more about the latest runtimes this control checks for all supported languages, see [AWS Lambda runtimes](https://docs.aws.amazon.com/lambda/latest/dg/lambda-run-time.html) in the *AWS Lambda Developer Guide*.

**Remediation**

For more information on supported runtimes and deprecation schedules, see the Runtime support policy section of the *AWS Lambda Developer Guide*. When you migrate your runtimes to the latest version, follow the syntax and guidance from the publishers of the language.

[RDS.1] RDS snapshots should be private

**Category:** Protect > Secure network configuration

**Severity:** Critical

**Resource:** RDS DB snapshot

**AWS Config rule:** [rds-snapshots-public-prohibited](https://docs.aws.amazon.com/config/latest/developerguide/config-rds-snapshots-public-prohibited.html)

**Parameters:** None

This control checks whether Amazon RDS snapshots are public.

RDS snapshots are used to back up the data on your RDS instances at a specific point in time. They can be used to restore previous states of RDS instances.
An RDS snapshot must not be public unless intended. If you share an unencrypted manual snapshot as public, this makes the snapshot available to all AWS accounts. This may result in unintended data exposure of your RDS instance.

Note that if the configuration is changed to allow public access, the AWS Config rule may not be able to detect the change for up to 12 hours. Until the AWS Config rule detects the change, the check passes even though the configuration violates the rule.

To learn more about sharing a DB snapshot, see Sharing a DB snapshot in the Amazon RDS User Guide.

Remediation

To remove public access for RDS snapshots
1. Open the Amazon RDS console at https://console.aws.amazon.com/rds/.
2. Navigate to Snapshots and then choose the public snapshot you want to modify.
3. From Actions, choose Share Snapshots.
4. From DB snapshot visibility, choose Private.
5. Under DB snapshot visibility, choose all.
6. Choose Save.

[RDS.2] RDS DB instances should prohibit public access, determined by the PubliclyAccessible configuration

Category: Protect > Secure network configuration

Severity: Critical

Resource: RDS DB instance

AWS Config rule: rds-instance-public-access-check

Parameters: None

This control checks whether Amazon RDS instances are publicly accessible by evaluating the PubliclyAccessible field in the instance configuration item.

The PubliclyAccessible value in the RDS instance configuration indicates whether the DB instance is publicly accessible. When the DB instance is configured with PubliclyAccessible, it is an Internet-facing instance with a publicly resolvable DNS name, which resolves to a public IP address. When the DB instance isn't publicly accessible, it is an internal instance with a DNS name that resolves to a private IP address.

Unless you intend for your RDS instance to be publicly accessible, the RDS instance should not be configured with PubliclyAccessible value, as this may allow unnecessary traffic to your database instance.

Remediation

To remove public access from RDS DB instances
1. Open the Amazon RDS console at https://console.aws.amazon.com/rds/.
2. Navigate to Databases and then choose your public database.
3. Choose Modify.
5. Choose Continue.
6. Under Scheduling of modifications choose Apply immediately.
7. Choose Modify DB Instance.

For more information, see Working with a DB instance in a VPC in the Amazon RDS User Guide.

[RDS.3] RDS DB instances should have encryption at-rest enabled

Category: Protect > Data protection > Encryption of data at rest

Severity: Medium

Resource: RDS DB instance

AWS Config rule: rds-storage-encrypted

Parameters: None

This control checks whether storage encryption is enabled for your Amazon RDS DB instances.

For an added layer of security for your sensitive data in RDS DB instances, you should configure your RDS DB instances to be encrypted at rest. To encrypt your RDS DB instances and snapshots at rest, enable the encryption option for your RDS DB instances. Data that is encrypted at rest includes the underlying storage for DB instances, its automated backups, read replicas, and snapshots.

RDS encrypted DB instances use the open standard AES-256 encryption algorithm to encrypt your data on the server that hosts your RDS DB instances. After your data is encrypted, Amazon RDS handles authentication of access and decryption of your data transparently with a minimal impact on performance. You do not need to modify your database client applications to use encryption.

Amazon RDS encryption is currently available for all database engines and storage types. Amazon RDS encryption is available for most DB instance classes. To learn about DB instance classes that do not support Amazon RDS encryption, see Encrypting Amazon RDS resources in the Amazon RDS User Guide.

Remediation

For information about encrypting DB instances in Amazon RDS, see Encrypting Amazon RDS resources in the Amazon RDS User Guide.

[S3.1] S3 Block Public Access setting should be enabled

Category: Protect > Secure network configuration

Severity: Medium

Resource: Account

AWS Config rule: s3-account-level-public-access-blocks

Parameters:
- ignorePublicAcls: true
- blockPublicPolicy: true
- blockPublicAcls: true
- restrictPublicBuckets: true
This control checks whether the following Amazon S3 public access block settings are configured at the account level:

- ignorePublicAcls: true
- blockPublicPolicy: true
- blockPublicAcls: true
- restrictPublicBuckets: true

Amazon S3 public access block is designed to provide controls across an entire AWS account or at the individual S3 bucket level to ensure that objects never have public access. Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, or both.

Unless you intend to have your S3 buckets be publicly accessible, you should configure the account level Amazon S3 Block Public Access feature.

To learn more, see Using Amazon S3 Block Public Access in the Amazon Simple Storage Service Developer Guide.

**Note**

This control is not supported in Middle East (Bahrain).

**Remediation**

**To enable Amazon S3 Block Public Access**

1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
2. Choose **Block public access (account settings)**.
3. Select **Block all public access**.
4. Choose **Save changes**.

For more information, see Using Amazon S3 block public access in the Amazon Simple Storage Service Developer Guide.

[S3.2] S3 buckets should prohibit public read access

**Category:** Protect > Secure network configuration  
**Severity:** Critical  
**Resource:** S3 bucket

**AWS Config rule:**  

a3-bucket-public-read-prohibited  

**Parameters:** None

This control checks whether your S3 buckets allow public read access. It evaluates the Block Public Access settings, the bucket policy, and the bucket access control list (ACL).

Some use cases require that everyone on the internet be able to read from your S3 bucket. However, those situations are rare. To ensure the integrity and security of your data, your S3 bucket should not be publicly readable.

**Remediation**

**To remove public access for an S3 bucket**

1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.  

240
2. Choose the name of the bucket where your CloudTrail logs are stored.
3. Choose Permissions and then choose Public access settings.
4. Choose Edit.
5. Select all four options and then choose Save.
6. If prompted, enter confirm and then choose Confirm.

[S3.3] S3 buckets should prohibit public write access

Category: Protect > Secure network configuration

Severity: Critical

Resource: S3 bucket

AWS Config rule: s3-bucket-public-write-prohibited

Parameters: None

This control checks whether your S3 buckets allow public write access. It evaluates the block public access settings, the bucket policy, and the bucket access control list (ACL).

Some use cases require that everyone on the internet be able to write to your S3 bucket. However, those situations are rare. To ensure the integrity and security of your data, your S3 bucket should not be publicly writable.

Remediation

To remove public access for an S3 bucket
1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
2. Choose the name of the bucket where your CloudTrail logs are stored.
3. Choose Permissions and then choose Public access settings.
4. Choose Edit.
5. Select all four options and then choose Save.
6. If prompted, enter confirm and then choose Confirm.

[S3.4] S3 buckets should have server-side encryption enabled

Category: Protect > Data protection > Encryption of data at rest

Severity: Medium

Resource: S3 bucket

AWS Config rule: s3-bucket-server-side-encryption-enabled

Parameters: None

This control checks that your S3 bucket either has Amazon S3 default encryption enabled or that the S3 bucket policy explicitly denies put-object requests without server-side encryption.

For an added layer of security for your sensitive data in S3 buckets, you should configure your buckets with server-side encryption to protect your data at rest. Amazon S3 encrypts each object with a unique key. As an additional safeguard, it encrypts the key itself with a master key that it rotates regularly.
Amazon S3 server-side encryption uses one of the strongest block ciphers available to encrypt your data, 256-bit Advanced Encryption Standard (AES-256).

To learn more, see Protecting data using server-side encryption with Amazon S3-managed encryption keys (SSE-S3) in the Amazon Simple Storage Service Developer Guide.

Remediation

**To enable default encryption on an S3 bucket**

1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
2. Choose the bucket from the list.
3. Choose Properties.
4. Choose Default encryption.
5. For the encryption, choose either AES-256 or AWS-KMS.
   - To use keys that are managed by Amazon S3 for default encryption, choose AES-256. For more information about using Amazon S3 server-side encryption to encrypt your data, see the Amazon Simple Storage Service Developer Guide.
   - To use keys that are managed by AWS KMS for default encryption, choose AWS-KMS. Then choose a master key from the list of the AWS KMS master keys that you have created.

Type the Amazon Resource Name (ARN) of the AWS KMS key to use. You can find the ARN for your AWS KMS key in the IAM console, under Encryption keys. Or, you can choose a key name from the drop-down list.

**Important**
If you use the AWS KMS option for your default encryption configuration, you are subject to the RPS (requests per second) quotas of AWS KMS. For more information about AWS KMS quotas and how to request a quota increase, see the AWS Key Management Service Developer Guide.

For more information about creating an AWS KMS key, see the AWS Key Management Service Developer Guide.

For more information about using AWS KMS with Amazon S3, see the Amazon Simple Storage Service Developer Guide.

When enabling default encryption, you might need to update your bucket policy. For more information about moving from bucket policies to default encryption, see the Amazon Simple Storage Service Developer Guide.

6. Choose Save.

For more information about default S3 bucket encryption, see the Amazon Simple Storage Service Console User Guide.

[SSM.1] EC2 instances should be managed by AWS Systems Manager

**Category:** Identify > Inventory

**Severity:** Medium

**Resource:** EC2 instance

**AWS Config rule:** ec2-instance-managed-by-systems-manager
Parameters: None

This control checks whether the EC2 instances in your account are managed by AWS Systems Manager. Systems Manager is an AWS service that you can use to view and control your AWS infrastructure.

To help you to maintain security and compliance, Systems Manager scans your managed instances. A managed instance is a machine that is configured for use with Systems Manager. Systems Manager then reports or takes corrective action on any policy violations that it detects. Systems Manager also helps you to configure and maintain your managed instances.

To learn more, see AWS Systems Manager User Guide.

Remediation

To ensure that EC2 instances are managed by Systems Manager

2. Choose Quick setup.
3. On the configuration screen, keep the default options.
4. Choose Set up Systems Manager.

To determine whether your instances support Systems Manager associations, see Systems Manager prerequisites in the AWS Systems Manager User Guide.

[SSM.2] All EC2 instances managed by Systems Manager should be compliant with patching requirements

Category: Detect > Detection services

Severity: Medium

Resource: SSM patch compliance

AWS Config rule: ec2-managedinstance-patch-compliance-status-check

Parameters: None

This control checks whether the compliance status of the Amazon EC2 Systems Manager patch compliance is COMPLIANT or NON_COMPLIANT after the patch installation on the instance. It only checks instances that are managed by Systems Manager Patch Manager.

Having your EC2 instances fully patched as required by your organization reduces the attack surface of your AWS accounts.

Note
This control is not supported in Middle East (Bahrain).

Remediation

To remediate noncompliant patches

2. Under Instances & Nodes, choose Run Command and then choose Run command.
3. Choose the button next to AWS-RunPatchBaseline.
4. Change the Operation to Install.
5. Choose Choose instances manually and then choose the noncompliant instances.
6. At the bottom of the page, choose Run.
7. After the command is complete, to monitor the new compliance status of your patched instances, in the navigation pane, choose Compliance.

For more information about using Systems Manager Documents to patch a managed instance, see About SSM documents for patching instances and Running commands using Systems Manager Run command in the AWS Systems Manager User Guide.
Logging AWS Security Hub API calls with AWS CloudTrail

AWS Security Hub is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Security Hub. CloudTrail captures API calls for Security Hub as events. The captured calls include calls from the Security Hub console and code calls to the Security Hub API operations. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Security Hub. 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 that CloudTrail collects, you can determine the request that was made to Security Hub, the IP address that the request was made from, who made the request, when it was made, and additional details.

To learn more about CloudTrail, including how to configure and enable it, see the AWS CloudTrail User Guide.

Security Hub information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When supported event activity occurs in Security Hub, 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 account. For more information, see Viewing events with CloudTrail event history.

For an ongoing record of events in your account, including events for Security Hub, 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

Security Hub supports logging all of the Security Hub API actions as events in CloudTrail logs. To view a list of Security Hub operations, see the Security Hub API Reference.

When activity for the following actions is logged to CloudTrail, the value for responseElements is set to null. This ensures that sensitive information isn't included in CloudTrail logs.

- BatchImportFindings
- GetFindings
- GetInsights
- GetMembers
- UpdateFindings
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 the CloudTrail userIdentity element.

Example: Security Hub 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, request parameters, and so on. 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 CreateInsight action. In this example, an insight called Test Insight is created. The ResourceId attribute is specified as the Group by aggregator, and no optional filters for this insight are specified. For more information about insights, see Insights in AWS Security Hub (p. 31).

```json
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAJK6U5DS22IAVUI7BW",
        "arn": "arn:aws:iam::012345678901:user/TestUser",
        "accountId": "012345678901",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "TestUser"
    },
    "eventTime": "2018-11-25T01:02:18Z",
    "eventSource": "securityhub.amazonaws.com",
    "eventName": "CreateInsight",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.251.233.179",
    "userAgent": "aws-cli/1.11.76 Python/2.7.10 Darwin/17.7.0 botocore/1.5.39",
    "requestParameters": {
        "Filters": {},
        "ResultField": "ResourceId",
        "Name": "Test Insight"
    },
    "responseElements": {
        "InsightArn": "arn:aws:securityhub:us-west-2:012345678901:insight/custom/f4c4890b-ae6c-4c26-95f9-e62cc46f3055"
    },
    "requestID": "c0fffccd-f04d-11e8-93fc-ddcd14710066",
    "eventID": "3dabcbebf-35b0-443f-a1a2-26e186ce23bf",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "recipientAccountId": "012345678901"
}
```
Automating AWS Security Hub with CloudWatch Events

Amazon CloudWatch Events enables you to automate your AWS services and respond automatically to system events such as application availability issues or resource changes. Events from AWS services are delivered to CloudWatch Events in near real time. You can write simple rules to indicate which events you’re interested in and what automated actions to take when an event matches a rule. The actions that can be automatically triggered include the following:

- Invoking an AWS Lambda function
- Invoking Amazon EC2 Run Command
- Relaying the event to Amazon Kinesis Data Streams
- Activating an AWS Step Functions state machine
- Notifying an Amazon SNS topic or an Amazon SQS queue
- Sending a finding to a third-party ticketing, chat, SIEM, or incident response and management tool

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

**Note**
As a best practice, make sure that the permissions granted to your users to access CloudWatch Events use least-privilege IAM policies, and that only the required permissions are granted.
For more information, see Authentication and Access Control for Amazon CloudWatch Events.

Types of Security Hub integration with CloudWatch Events

Security Hub supports the following types of integration with CloudWatch Events, using the following CloudWatch event types.

On the CloudWatch Events dashboard for Security Hub, All Events includes all of these event types.

**All findings (Security Hub Findings - Imported)**

Security Hub automatically sends all findings to CloudWatch Events as Security Hub Findings - Imported events. Security Hub Findings - Imported events are triggered by updates from both BatchImportFindings and BatchUpdateFindings.

You can define rules in CloudWatch Events that automatically route findings to an Amazon S3 bucket, a remediation workflow, or a third-party tool.

Use this method to automatically send all findings, or all findings with specific characteristics, to a response or remediation workflow.

See the section called “Configuring a CloudWatch Events rule for Security Hub findings that are automatically sent to CloudWatch Events” (p. 248).
Findings for custom actions (Security Hub Findings - Custom Action)

Security Hub also sends findings associated with custom actions to CloudWatch Events as Security Hub Findings - Custom Action events.

This is useful for analysts working with the Security Hub console who want to send a specific finding, or a small set of findings, to a response or remediation workflow. You can select a custom action for up to 20 findings at a time. The set of findings is sent to CloudWatch as a single CloudWatch event.

When you create a custom action, you specify a custom action ID for the custom action. You can use the custom action ID to create a rule in CloudWatch Events that defines a specific action to take when a finding is received that is associated with the custom action ID.

See the section called “Creating a custom action and associating it with a CloudWatch Events rule” (p. 249).

For example, you can create a custom action in Security Hub called send_to_ticketing. Then in CloudWatch Events, you create a rule that is triggered when CloudWatch Events receives a finding that includes the send_to_ticketing custom action ID. The rule includes logic to send the finding to your ticketing system. You can then select findings within Security Hub and use the custom action in Security Hub to manually send findings to your ticketing system.

For examples of how to send Security Hub findings to CloudWatch Events for further processing, see How to Integrate AWS Security Hub Custom Actions with PagerDuty and How to Enable Custom Actions in AWS Security Hub on the AWS Partner Network (APN) Blog.

Insight results for custom actions (Security Hub Insight Results)

You can also use custom actions to send a set of insight results to CloudWatch Events as Security Hub Insight Results events. Insight results are the set of resources associated with an insight. Note that when you send insight results to CloudWatch Events, you are not sending the findings to CloudWatch Events. You are only sending the resource identifiers associated with the insight results. You can send up to 100 resource identifiers at a time.

Similar to custom actions for findings, you first create the custom action in Security Hub, then create a rule in CloudWatch Events.

See the section called “Creating a custom action and associating it with a CloudWatch Events rule” (p. 249).

For example, if you see a particular insight result of interest that you want to share with a colleague, you can use custom actions to send that insight result to the colleague via a chat or ticketing system.

Configuring a CloudWatch Events rule for Security Hub findings that are automatically sent to CloudWatch Events

You can create a rule in CloudWatch Events that defines an action to take when a Security Hub Findings - Imported event is received. A Security Hub Findings - Imported events are triggered by updates from both BatchImportFindings and BatchUpdateFindings.
Creating a custom action and associating it with a CloudWatch Events rule

To create a CloudWatch Events rule for a Security Hub finding

2. In the navigation pane, choose Rules.
3. Choose Create rule.
4. For Event source, confirm that Event Pattern is selected.
5. Choose Edit for Event Pattern Preview.
6. Copy the following example pattern and paste it into the preview window. Be sure to replace the existing brackets.

```json
{
    "source": [  
        "aws.securityhub"
    ],
    "detail-type": [  
        "Security Hub Findings - Imported"
    ]
}
```
7. Choose Save to close the window.
8. Choose Add target, then select the target to invoke when this rule is matched.

You might need to configure the settings for the selected target.

Creating a custom action and associating it with a CloudWatch Events rule

To use Security Hub custom actions to send findings or insight results to CloudWatch Events, you first create the custom action in Security Hub, then define the rule in CloudWatch Events.

The rule in CloudWatch Events uses the ARN from the custom action.

To create a custom action in Security Hub

2. In the navigation pane, choose Settings and then choose Custom actions.
3. Choose Create custom action.
4. Provide a Name, Description, and Custom action ID for the action.

   The Name must be fewer than 20 characters.

   The Custom action ID must be unique per AWS account.
5. Choose Create custom action.
6. Make a note of the Custom action ARN. You need to use the ARN when you create a rule to associate with this action in CloudWatch Events.

To define a rule in CloudWatch Events

2. In the navigation pane, choose Rules.
3. Choose **Create rule**.
4. For **Event source**, confirm that **Event Pattern** is selected.
5. Choose **Edit** for **Event Pattern Preview**.
6. Copy one of the following example patterns, and paste it into the preview window. Be sure to replace the existing brackets.

   For an event associated with a finding custom action (**Security Hub Findings - Custom Action** event type), use the following format. Replace the placeholder ARN with the **Custom Action ARN** for the custom action you created.

   ```json
   {  
       "source": [  
           "aws.securityhub"
       ],  
       "detail-type": [  
           "Security Hub Findings - Custom Action"
       ],  
       "resources": [  
           "arn:aws:securityhub:us-west-2:123456789012:action/custom/test-action1"
       ]
   }
   ``

   For an event associated with an insight custom action (**Security Hub Insight Results** event type), use the following format. Replace the placeholder ARN with the **Custom Action ARN** for the custom action you created.

   ```json
   {  
       "source": [  
           "aws.securityhub"
       ],  
       "detail-type": [  
           "Security Hub Insight Results"
       ],  
       "resources": [  
           "arn:aws:securityhub:us-west-2:123456789012:action/custom/test-action1"
       ]
   }
   ``

   7. Choose **Save** to close the window.

   8. Choose **Add target** and then select the target to invoke when this rule is matched.

   9. Choose **Configure details**.

   10. Enter a name and description for the rule.

       To enable the rule now, for **State**, choose **Enabled**. To save the rule without enabling it, clear **Enabled**.

   11. Choose **Create rule**.

   After this rule is created in CloudWatch Events, when you perform a custom action on finding or insight results in your account, events are generated in CloudWatch Events.

CloudWatch Events formats for Security Hub

The CloudWatch events for the **Security Hub Findings - Imported**, **Security Findings - Custom Action**, and **Security Hub Insight Results** event types use the following formats.
Security Hub Findings - Imported

The CloudWatch event for the **Security Hub Findings - Imported** event type is in the following format.

```json
{
    "version": "0",
    "id": "CWE-event-id",
    "detail-type": "Security Hub Findings - Imported",
    "source": "aws.securityhub",
    "account": "111122223333",
    "time": "2019-04-11T21:52:17Z",
    "region": "us-west-2",
    "resources": [
    ],
    "detail": {
        "findings": [AMAZON_FINDING_JSON]
    }
}
```

For a complete list of parameters included in **AMAZON_FINDING_JSON**, see AWS Security Finding Format (ASFF) (p. 46).

Security Hub Findings - Custom Action

The CloudWatch event for the **Security Hub Findings - Custom Action** event type is in the following format.

```json
{
    "version": "0",
    "id": "1a1111a1-b22b-3c33-444d-5555e5ee5555",
    "detail-type": "Security Hub Findings - Custom Action",
    "source": "aws.securityhub",
    "account": "111122223333",
    "time": "2019-04-11T18:43:48Z",
    "region": "us-west-1",
    "resources": [
    ],
    "detail": {
        "actionName": "custom-action-name",
        "actionDescription": "description of the action",
        "findings": [AMAZON_FINDING_JSON for each specified finding]
    }
}
```

For a complete list of parameters included in **AMAZON_FINDING_JSON**, see AWS Security Finding Format (ASFF) (p. 46).

Security Hub Insight Results

The CloudWatch Events event for the **Security Hub Insight Results** event type has the following format.

```json
{
    "version": "0",
    "id": "1a1111a1-b22b-3c33-444d-5555e5ee5555",
    "detail-type": "Security Hub Insight Results",
    "source": "aws.securityhub",
    "account": "111122223333",
    "time": "2019-04-11T18:43:48Z",
    "region": "us-west-1",
    "resources": [
    ],
    "detail": {
        "actionName": "custom-action-name",
        "actionDescription": "description of the action",
        "findings": [AMAZON_FINDING_JSON for each specified finding]
    }
}
```
Use custom actions to send Security Hub findings to CloudWatch Events

After you've created one or more Security Hub custom actions and CloudWatch Events rules, you can send findings and insight results to CloudWatch Events for further management and processing.

Events are sent to CloudWatch Events only in the account in which they are viewed. If you are viewing a finding using a master account, the event is sent to CloudWatch Events in the master account.

For AWS API calls to be effective, the implementations of target code must switch roles into member accounts. This also means that the role you must switch into must be deployed to each member where action is needed.

To send findings to CloudWatch Events

2. On the Findings page, select one or more findings to send to CloudWatch Events. You can select up to 20 findings at a time.
3. From the Actions drop down, choose the custom action that aligns with the CloudWatch Events rule to apply.

   If successful, the message Successfully sent findings to Amazon CloudWatch Events is displayed.

To send insight results to CloudWatch Events

2. On the Insights page, choose the insight that includes the findings results to send to CloudWatch Events.
3. Select the findings from the insight to send to CloudWatch Events. You can select up to 20 findings at a time.
4. For Actions, choose the custom action that aligns with the CloudWatch Events rule to apply.

   If successful, the message Successfully sent findings to Amazon CloudWatch Events is displayed.
Disabling AWS Security Hub

To disable AWS Security Hub, you can use the Security Hub console or the `DisableSecurityHub` operation of the Security Hub API.

When you disable Security Hub, the following occurs.

- After 90 days, your existing findings and insights and any Security Hub configuration settings are deleted and cannot be recovered.
  
  If you want to save your existing findings, you must export them before you disable Security Hub. For more information, see Accounts and data retention in Security Hub (p. 29).
- Any enabled standards are disabled.
- Your master and member account relationships are removed.

To disable Security Hub (console)

2. In the navigation pane, under Settings, choose General.

When you disable Security Hub for an account, it is disabled only in the current Region. No new findings are processed for the account in that Region.
## Document history for the AWS Security Hub User Guide

The following table describes the updates to the documentation for AWS Security Hub.

<table>
<thead>
<tr>
<th>update-history-change</th>
<th>update-history-description</th>
<th>update-history-date</th>
</tr>
</thead>
<tbody>
<tr>
<td>New AWS Foundational Security Best Practices standard</td>
<td>Added the new AWS Foundational Security Best Practices standard, which is a set of controls that detect when your deployed accounts and resources deviate from security best practices.</td>
<td>April 22, 2020</td>
</tr>
<tr>
<td>New console option to update the workflow status for a finding</td>
<td>Added information for using the Security Hub console or API to set the workflow status for findings.</td>
<td>April 16, 2020</td>
</tr>
<tr>
<td>New BatchUpdateFindings API for customer updates to findings</td>
<td>Added information on using BatchUpdateFindings to update information related to the process of investigating a finding. BatchUpdateFindings replaces UpdateFindings, which is deprecated.</td>
<td>April 16, 2020</td>
</tr>
<tr>
<td>Updates to the AWS Security Finding Format (ASFF)</td>
<td>Added several new resource types. Added a new Label attribute to the Severity object. Label is intended to replace the Normalized field. Added a new Workflow object to track the process of an investigation into a finding. Workflow contains a Status attribute, which replaces the existing Workflowstate attribute.</td>
<td>March 12, 2020</td>
</tr>
<tr>
<td>Updates to the Integrations page</td>
<td>Updated to reflect the changes to the Integrations page. For each integration, the page now shows the integration category and whether each integration sends findings to or receives findings from Security Hub. It also provides the specific steps required to enable each integration.</td>
<td>February 26, 2020</td>
</tr>
<tr>
<td><strong>New third-party product integrations</strong></td>
<td>Added the following new product integrations: Cloud Custodian, FireEye Helix, Forcepoint CASB, Forcepoint DLP, Forcepoint NGFW, Rackspace Cloud Native Security, and Vectra.ai Cognito.</td>
<td>February 21, 2020</td>
</tr>
<tr>
<td><strong>New security standard for the Payment Card Industry Data Security Standard (PCI DSS)</strong></td>
<td>Added the Security Hub security standard for the Payment Card Industry Data Security Standard (PCI DSS). When this standard is enabled, Security Hub performs automated checks against controls related to PCI DSS requirements.</td>
<td>February 13, 2020</td>
</tr>
<tr>
<td><strong>Updates to the AWS Security Finding Format (ASFF)</strong></td>
<td>Added a field for related requirements for standards controls. Added new resource types and new resource details. The ASFF also now allows you to provide up to 32 resources.</td>
<td>February 5, 2020</td>
</tr>
<tr>
<td><strong>New option to disable individual security standard controls</strong></td>
<td>Added information on how to control whether each individual security standard control is enabled.</td>
<td>January 15, 2020</td>
</tr>
<tr>
<td><strong>Updates to Terminology and Concepts</strong></td>
<td>Updated some descriptions and added new terms to Terminology and Concepts.</td>
<td>September 21, 2019</td>
</tr>
<tr>
<td><strong>AWS Security Hub general availability release (p. 254)</strong></td>
<td>Content updates to reflect improvements made to Security Hub during the preview period.</td>
<td>June 25, 2019</td>
</tr>
<tr>
<td><strong>Added remediation steps for CIS AWS Foundations checks</strong></td>
<td>Added remediation steps to Security Standards Supported in AWS Security Hub.</td>
<td>April 15, 2019</td>
</tr>
<tr>
<td><strong>Preview release of AWS Security Hub (p. 254)</strong></td>
<td>Published the preview release version of the AWS Security Hub User Guide.</td>
<td>November 18, 2018</td>
</tr>
</tbody>
</table>