# AWS Security Hub

## User Guide

# AWS Security Hub: User Guide



The AWS Documentation website is getting a new look!

Try it now and let us know what you think. Switch to the new look >>

You can return to the original look by selecting English in the language selector above.

# Table of Contents


What Is AWS Security Hub? ... 1
- Benefits of Security Hub ... 1
- Getting Started with Security Hub ... 1
- Security Hub Free Trial ... 2
- Using Security Hub ... 2

Terminology and Concepts ... 3
Limits ... 5
Supported Regions ... 6
Setting Up Security Hub ... 7
- Enabling Security Hub ... 7

Security ... 9
- Data Protection ... 9
- Identity and Access Management ... 10
  - Audience ... 10
  - Authenticating with Identities ... 11
  - AWS Account Root User ... 11
  - IAM Users and Groups ... 11
  - IAM Roles ... 11
  - Managing Access Using Policies ... 12
  - How AWS Security Hub Works with IAM ... 14
- Compliance Validation ... 19
- Infrastructure Security ... 19

Managing Access to Security Hub ... 21
- Using IAM Policies to Delegate Security Hub Access to IAM Identities ... 21
  - AWS Managed (Predefined) Policies for Security Hub ... 21
  - Resources Defined by Security Hub ... 21
  - ... 22
- Using Service-Linked Roles ... 22
  - Service-Linked Role Permissions for Security Hub ... 22
  - Creating a Service-Linked Role for Security Hub ... 23
  - Editing a Service-Linked Role for Security Hub ... 24
  - Deleting a Service-Linked Role for Security Hub ... 24

Master and Member Accounts ... 25
- Designating Master and Member Accounts on the Security Hub Console ... 25
- Designating Master and Member Accounts Through Security Hub API Operations ... 27
- Accounts and Data Retention in Security Hub ... 27

Insights ... 29
- Custom Insights ... 29
  - Working with Insights ... 30
- Managed Insights ... 31

Findings ... 33
- Working with Findings in Security Hub ... 33
- Finding Format ... 35
  - Syntax of the AWS Security Finding Format ... 35
  - Attributes of the AWS Security Finding Format ... 37
  - Types Taxonomy of the AWS Security Finding ... 65

Product Integrations ... 68
- AWS Product Integrations ... 68
- Third-Party Partner Product Integrations ... 69
- Custom Product Integrations ... 72

Compliance Standards ... 74
- Enabling the CIS AWS Foundations Standard in Security Hub ... 74
- How the CIS AWS Foundations Standard in Security Hub Uses AWS Config ... 74
  - AWS Config Resources Required for CIS Checks ... 75

Results of Standards Checks in Security Hub ..... 76
CIS AWS Foundations Standard Checks Supported in Security Hub ..... 76
1.1 – Avoid the use of the "root" account ..... 77
Remediation ..... 77
1.2 – Ensure multi-factor authentication (MFA) is enabled for all IAM users that have a console password ..... 78
Remediation ..... 78
1.3 – Ensure credentials unused for 90 days or greater are disabled ..... 79
Remediation ..... 79
1.4 – Ensure access keys are rotated every 90 days or less ..... 79
Remediation ..... 79
1.5 – Ensure IAM password policy requires at least one uppercase letter ..... 80
Remediation ..... 80
1.6 – Ensure IAM password policy requires at least one lowercase letter ..... 80
Remediation ..... 81
1.7 – Ensure IAM password policy requires at least one symbol ..... 81
Remediation ..... 81
1.8 – Ensure IAM password policy requires at least one number ..... 81
Remediation ..... 81
1.9 – Ensure IAM password policy requires a minimum length of 14 or greater ..... 82
Remediation ..... 82
1.10 – Ensure IAM password policy prevents password reuse ..... 82
Remediation ..... 82
1.11 – Ensure IAM password policy expires passwords within 90 days or less ..... 82
Remediation ..... 83
1.12 – Ensure no root account access key exists ..... 83
Remediation ..... 83
1.13 – Ensure MFA is enabled for the "root" account ..... 84
Remediation ..... 84
1.14 – Ensure hardware MFA is enabled for the "root" account ..... 84
Remediation ..... 85
1.16 – Ensure IAM policies are attached only to groups or roles ..... 85
Remediation ..... 85
1.22 – Ensure IAM policies that allow full "*:*" administrative privileges are not created ..... 86
Remediation ..... 86
2.1 – Ensure CloudTrail is enabled in all Regions ..... 86
Remediation ..... 87
2.2. – Ensure CloudTrail log file validation is enabled ..... 88
Remediation ..... 88
2.3 – Ensure the S3 bucket CloudTrail logs to is not publicly accessible ..... 88
Remediation ..... 88
2.4 – Ensure CloudTrail trails are integrated with Amazon CloudWatch Logs ..... 89
Remediation ..... 89
2.5 – Ensure AWS Config is enabled in all Regions ..... 90
Remediation ..... 90
2.6 – Ensure S3 bucket access logging is enabled on the CloudTrail S3 bucket ..... 90
Remediation ..... 91
2.7 – Ensure CloudTrail logs are encrypted at rest using AWS KMS CMKs ..... 91
Remediation ..... 91
2.8 – Ensure rotation for customer created CMKs is enabled ..... 92
Remediation ..... 92
2.9 – Ensure VPC flow logging is enabled in all VPCs ..... 92
Remediation ..... 93
3.1 – Ensure a log metric filter and alarm exist for unauthorized API calls ..... 93
Remediation ..... 93
3.2 – Ensure a log metric filter and alarm exist for AWS Management Console sign-in without MFA ..... 94
Remediation ..... 94

3.3 – Ensure a log metric filter and alarm exist for usage of "root" account ..... 95
Remediation ..... 95
3.4 – Ensure a log metric filter and alarm exist for IAM policy changes ..... 96
Remediation ..... 96
3.5 – Ensure a log metric filter and alarm exist for CloudTrail configuration changes ..... 97
Remediation ..... 98
3.6 – Ensure a log metric filter and alarm exist for AWS Management Console authentication failures ... 98
Remediation ..... 99
3.7 – Ensure a log metric filter and alarm exist for disabling or scheduled deletion of customer created CMKs ..... 100
Remediation ..... 100
3.8 – Ensure a log metric filter and alarm exist for S3 bucket policy changes ..... 101
Remediation ..... 101
3.9 – Ensure a log metric filter and alarm exist for AWS Config configuration changes ..... 102
Remediation ..... 102
3.10 – Ensure a log metric filter and alarm exist for security group changes ..... 103
Remediation ..... 103
3.11 – Ensure a log metric filter and alarm exist for changes to Network Access Control Lists (NACL) ... 104
Remediation ..... 104
3.12 – Ensure a log metric filter and alarm exist for changes to network gateways ..... 105
Remediation ..... 105
3.13 – Ensure a log metric filter and alarm exist for route table changes ..... 106
Remediation ..... 107
3.14 – Ensure a log metric filter and alarm exist for VPC changes ..... 107
Remediation ..... 108
4.1 – Ensure no security groups allow ingress from 0.0.0.0/0 to port 22 ..... 109
Remediation ..... 109
4.2 – Ensure no security groups allow ingress from 0.0.0.0/0 to port 3389 ..... 109
Remediation ..... 109
4.3 – Ensure the default security group of every VPC restricts all traffic ..... 110
Remediation ..... 110
CIS AWS Foundations Standard Checks That Aren't Supported in Security Hub ..... 110
Security Hub with CloudTrail ..... 112
Security Hub Information in CloudTrail ..... 112
Example: Security Hub Log File Entries ..... 113
Security Hub with CloudWatch Events ..... 114
Configuring a CloudWatch Events Rule for Security Hub Findings That Are Automatically Sent to CloudWatch Events ..... 115
Creating a Custom Action and Associating It with a CloudWatch Events Rule ..... 115
CloudWatch Events Formats for Security Hub ..... 115
Use Custom Actions to Send Security Hub Findings to CloudWatch Events ..... 118
Disabling AWS Security Hub ..... 119
Document History ..... 120

# 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 compliance with the 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 Security Hub (p. 1)
- Getting Started with Security Hub (p. 1)
- Security Hub Free Trial (p. 2)
- Using Security Hub (p. 2)


## Benefits of Security Hub

- 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 findings 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.
- Security Hub automatically runs continuous, account-level configuration and compliance checks based on industry standards and best practices, such as the Center for Internet Security (CIS) AWS Foundations Benchmarks. The result of these checks is provided as a compliance score, and specific accounts and resources that require attention are identified.
- Security Hub consolidates your security findings across accounts and provider products and displays results on the Security Hub console pages. This lets you view the current status of the security and compliance checks to spot trends, identify potential issues, and take the necessary remediation steps.
- Security Hub supports integration with Amazon CloudWatch Events, which lets you automate remediation of specific findings by defining custom actions to take when a finding is received. You can configure custom actions to, for example, send findings to a ticketing system or to an automated remediation system.

## Getting Started with Security Hub

When you enable Security Hub, it immediately begins consuming, aggregating, organizing, and prioritizing findings from AWS services, such as Amazon GuardDuty, Amazon Inspector, and Amazon Macie, and from AWS partner security products. Security Hub also generates its own findings by running continuous, automated compliance checks based on AWS best practices and supported industry standards. Security Hub then correlates and consolidates findings across providers to help you to prioritize the most significant findings.

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.

> **Important**
> Security Hub detects and consolidates only those findings from the supported AWS and partner products that are generated after you enable Security Hub. It doesn't retroactively detect and consolidate security findings that were generated before you enabled it. Security Hub receives

and processes only those findings from the same Region where you enabled Security Hub in your account. For full compliance with CIS AWS Foundations Benchmark compliance checks, you must enable Security Hub in all AWS Regions.

# Security Hub Free Trial

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 compliance standards.

You can view your usage details during your free trial in the **Usage** tab of the **Settings** page of the Security Hub console. The usage details include the time remaining for the free trial and an estimated monthly cost for using Security Hub. The estimated monthly cost is based on your Security Hub usage during the free trial for findings and compliance checks projected over a 30-day period.

If you are using Security Hub from the master account, the estimated monthly cost includes the costs associated with all 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 charge is for only the current Region, not for all Regions in which Security Hub is enabled.

You are not charged for using Security Hub until your free trial ends. To learn more, see Security Hub Pricing.

# Using 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**

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

# 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. With the master account, you can view findings in member accounts.

An account can't 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. 25).

**Archived finding**

A finding that has a RecordState set to ARCHIVED. When you archive a finding in Security Hub it is excluded from the default view of the **Findings** page in the console. When you receive a finding for an issue or failed compliance check, you can archive it so that you see only active findings that you want to further investigate or take remediation steps for. Archived findings aren't deleted. You can modify the filter applied to the **Findings** page to display only the findings that you want to see. To view only archived findings, update or repalce the filter applied to the page to RecordState EQUALS ARCHIVED.

When you use the `GetFindings` operation of the Security Hub API, all findings are returned, both active and archived. Use filters in your request to return findings that match specific criteria. For example, to retrieve archived findings:

```
"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 compliance checks. For more information, see AWS Security Finding Format (p. 35).

**Compliance check**

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

**Compliance 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. Compliance standards can be based on regulatory frameworks, best practices, or internal company policies. To learn more about compliance standards in Security Hub, see Compliance Standards: CIS AWS Foundations (p. 74).

**Compliance rule**

The logic used to evaluate a control and conduct a compliance check. A single control can be evaluated by one or multiple rules. A rule may be reused across multiple controls. Security Hub leverages compliance rules powered by Config and has developed some native compliance rules run outside of AWS Config.

**Control**

A policy statement that, when implemented, reduces risk. A compliance standard consists of controls.

**Finding**

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

For more information about findings in Security Hub, see Findings in AWS Security Hub (p. 33).

**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 in AWS Security Hub (p. 29).

# Limits

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

| Resource | Default Limit | Comments |
| --- | --- | --- |
| Number of Security Hub member accounts | 1000 | The maximum number of Security Hub member accounts that can be added per account (Security Hub master account) per Region.<br><br>This is a hard limit. You can't request a limit increase of Security Hub member accounts. |
| Number of Security Hub outstanding invitations | 1000 | The maximum number of outstanding Security Hub member account invitations that can be sent per account (Security Hub master account) per Region.<br><br>This is a hard limit. You can't request a limit increase of Security Hub outstanding invitations. |
| Number of Security Hub custom insights | 100 | The maximum number of user-defined custom Security Hub insights that can be created per account per Region.<br><br>This is a hard limit. You can't request a limit increase of Security Hub custom insights. |
| Number of insight results | 100 | The maximum number of aggregated results returned for the `GetInsightsResults` API operation.<br><br>This is a hard limit. You can't request a limit increase of insight results. |

# 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**

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

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

## Enabling Security Hub

To use Security Hub, you must first enable it.

**Permissions required to enable Security Hub**

1. 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"
                }
            }
        }
    ]
}
```

2. Use the credentials of the IAM identity from step 1 to sign in to the Security Hub console. When you open the Security Hub console for the first time, choose **Get Started** and then choose **Enable Security Hub** .

When you enable Security Hub, it's 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 supported standards (in this release, CIS AWS Foundations) compliance checks

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. 22). For more information about service-linked roles, see Using Service-Linked Roles in the *IAM User Guide*.

By enabling Security Hub in a particular account, you also, by default, enable the supported CIS AWS Foundations standard in that account. For Security Hub to successfully run compliance checks against the rules included in the CIS AWS Foundations standard, you must have AWS Config enabled in the account where you enabled Security Hub. (If this is a Security Hub master account, enable AWS Config in each of this account's Security Hub member accounts.) Security Hub doesn't 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 by using the AWS CloudFormation "Enable AWS Config" template in AWS CloudFormation StackSets Sample Templates.

**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. 9)
- Identity and Access Management for AWS Security Hub (p. 10)
- Compliance Validation for AWS Security Hub (p. 19)
- Infrastructure Security in AWS Security Hub (p. 19)

# 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*.

# 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. 10)
- Authenticating with Identities (p. 11)
- AWS Account Root User (p. 11)
- IAM Users and Groups (p. 11)
- IAM Roles (p. 11)
- Managing Access Using Policies (p. 12)
- How AWS Security Hub Works with IAM (p. 14)

## 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. 17).

**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. 14).

**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. 16).

# 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. 14)
- Security Hub Resource-Based Policies (p. 15)
- Authorization Based on Security Hub Tags (p. 15)
- Security Hub IAM Roles (p. 15)
- Service-Linked Roles (p. 16)
- Service Roles (p. 16)
- AWS Security Hub Identity-Based Policy Examples (p. 16)


## 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

Resource-based policies are JSON policy documents that specify what actions a specified principal can perform on the Security Hub resource and under what conditions. Security Hub supports resource-based permissions policies for Security Hub the following resources:

- Hub

Resource-based policies let you grant usage permission to other accounts on a per-resource basis.

To enable cross-account access, you can specify an entire account or IAM entities in another account as the principal in a resource-based policy. Adding a cross-account principal to a resource-based policy is only half of establishing the trust relationship. When the principal and the resource are in different AWS accounts, you must also grant the principal entity permission to access the resource. Grant permission by attaching an identity-based policy to the entity. However, if a resource-based policy grants access to a principal in the same account, no additional identity-based policy is required. For more information, see How IAM Roles Differ from Resource-based Policies in the *IAM User Guide*.

## 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. 16)
- Using the Security Hub Console (p. 17)
- Troubleshooting AWS Security Hub Identity and Access (p. 17)

### 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*:

```
{
    "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. 18)
- I Am Not Authorized to Perform iam:PassRole (p. 18)
- I Want to View My Access Keys (p. 18)
- I'm an Administrator and Want to Allow Others to Access Security Hub (p. 18)
- I Want to Allow People Outside My AWS Account to Access My Security Hub Resources (p. 19)

### 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, `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`). Like a user name and password, you must use both the access key ID and secret access key together to authenticate your requests. Manage your access keys as securely as you do your user name and password.

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

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

### I'm an Administrator and Want to Allow Others to Access 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. 14).
- To learn how to provide access to your resources across AWS accounts that you own, see Providing Access to an IAM User in Another AWS Account That You Own in the *IAM User Guide*.
- To learn how to provide access to your resources to third-party AWS accounts, see Providing Access to AWS Accounts Owned by Third Parties in the *IAM User Guide*.
- To learn how to provide access through identity federation, see Providing Access to Externally Authenticated Users (Identity Federation) in the *IAM User Guide*.
- To learn the difference between using roles and resource-based policies for cross-account access, see How IAM Roles Differ from Resource-based Policies in the *IAM User Guide*.

# Compliance Validation for AWS Security Hub

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

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

## Using IAM Policies to Delegate Security Hub Access to IAM Identities

This section describes how to delegate Security Hub access to various IAM identities (users, groups, and roles).

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. Each action in the table identifies the resource types that can be specified with that action. A resource type can also define which condition keys you can include in a policy. These keys are displayed in the last column of the table.

**Resources Available in Security Hub**

| Resource Types | ARN | |
|---|---|---|
| action-target | `arn:${Partition}:securityhub:${Region}:${Account}:action/custom/${Id}` | |
| hub | `arn:${Partition}:securityhub:${Region}:${Account}:hub/default` | |
| insight | `arn:${Partition}:securityhub:${Region}:${Account}:insight/${Company}/${ProductId}/${UniqueId}` | |
| standard | `arn:${Partition}:securityhub:::ruleset/${StandardsName}/v/${StandardsVersion}` | |

| Resource Types | ARN | |
|---|---|---|
| standards-subscription | `arn:${Partition}:securityhub:${Region}:${Account}:subscription/${StandardsName}/v/${StandardsVersion}` | |
| product-subscription | `arn:${Partition}:securityhub:${Region}:${Account}:product-subscription/${Company}/${ProductId}` | |
| product | `arn:${Partition}:securityhub:${Region}:${Account}:product/${Company}/${ProductId}` | |

Security Hub defines the following condition key 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.

| Condition Keys | Description | Type |
|---|---|---|
| securityhub:TargetAccount | The ID of the AWS account to import findings in to. In the AWS Security Finding format, this field is called AwsAccountId | String |

# 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. 6).

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: `GetComplianceDetailsByConfigRule`
- 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.

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

# 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. 119).

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 it 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**

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.

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

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
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

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
2. Do one of the following:
   - If Security Hub isn't enabled, on the Security Hub first-run experience page, in the **AWS Security Hub Setup** section, choose **Enable Security Hub**. On the **Welcome to AWS Security Hub** page, choose **Enable AWS Security Hub**. Back on the first-run experience page, choose **Go to Security Hub**.

     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. The CIS AWS Foundations compliance standard is 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, compliance 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 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 or delete. You can also create custom insights to track security issues that are unique to your AWS environment and usage.

**Security Hub supports the following insight types:**

- Custom Insights (p. 29)
- Managed Insights (p. 31)

> **Important**
> You can create your own custom insights. You can't edit or delete Security Hub managed insights.

## Custom Insights

In Security Hub, an insight is a collection of related findings defined by an aggregation statement and optional filters. A custom insight is an insight that you create to track security issues and risks that are specific to your environment.

**To create an insight**

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
2. In the navigation pane, choose **Insights**.
3. Choose **Create insight**.
4. Choose the **Add filter** field and then select **Group by**.
5. Select the attribute to use to group the findings associated with this insight and then choose **Apply**.
6. (Optional) Choose any additional filters to use for this insight, define the filter criteria, and then choose **Apply** after adding each filter.

   > **Note**
   > For optional filters, AND logic is applied to your specified collection of filters to query your findings. However, OR logic is applied to multiple filters that use the same attribute that is set to different values.
7. Choose **Create insight**.
8. Enter an **Insight name** and then choose **Create insight**.

You can choose only one **Group by** aggregator (one attribute/value pair) in a Security Hub insight. Attributes available for filtering and grouping insights and findings include the following:

- **Aws account Id**
- **Company name**
- **Compliance status**
- **Generator ID**
- **Malware name**
- **Process name**

- **Threat intel type**
- **Product ARN**
- **Product name**
- **Record state**
- **EC2 instance image ID**
- **EC2 instance IPv4**
- **EC2 instance IPv6**
- **EC2 instance key name**
- **EC2 instance subnet ID**
- **EC2 instance type**
- **EC2 instance VPC ID**
- **IAM access key user name**
- **S3 bucket owner name**
- **Container image ID**
- **Container image name**
- **Container name**
- **Resource ID**
- **Resource type**
- **Severity label**
- **Source URL**
- **Type**
- **Verification state**
- **Workflow state**

For the complete list of AWS Security Finding Format attributes and their descriptions, see AWS Security Finding Format (p. 35).

# Working with Insights

You can modify an existing insight and then save the updates, or you can choose to save it as a new insight.

### To modify an insight

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
2. In the navigation pane, choose **Insights**.
3. Choose the insight to modify and then do one or more of the following:
   - To remove a filter from the insight, choose the circled X next to the filter.
   - To add a new filter, choose the **Add filter** field, select the attribute to use as a filter, then choose **Apply**.
   - To change the attribute used to group findings in the insight, first choose the circled X next to **Group by** to remove the existing grouping. Then choose the **Add filter** field, select the attribute to use for the **Group by** aggregator, then choose **Apply**.
4. When you've finished updating the insight, choose **Save insight** and then do one of the following:
   - To replace the existing insight with 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**.

If you modify the filters and the **Group by** aggregator of a managed insight, you can only save your changes as a new insight. You can't update the filters and the **Group by** aggregator of a managed insight.

When you no longer want an insight, you can delete it. You can delete your custom insights. You can't delete managed insights.

### To delete a custom insight

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
2. In the navigation pane, choose **Insights**.
3. Locate the insight to delete, choose the more options icon (the three dots in the top-left corner of the card) for the insight, and then choose **Delete**.

After you create an insight, you can use it to apply an action to associated findings. The default action is to archive findings. You can also define your own custom actions.

### To apply an action to the findings in an insight

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
2. In the navigation pane, choose **Insights**.
3. Choose the insight that includes findings to apply an action to.
4. Select all of the findings to apply the action to.
5. Choose **Actions** and then choose the action to apply. Only **Archive** appears if you haven't defined any custom actions.

   **Note**
   You can create Security Hub custom actions to automate Security Hub with Amazon CloudWatch Events. For more information and detailed steps on creating custom actions, see Automating AWS Security Hub with CloudWatch Events (p. 114).

# Managed Insights

In the current release, AWS Security Hub offers the following managed (default) insights:

**Important**
You can't edit or delete Security Hub managed insights.

- AWS resources with the most findings
- Amazon S3 buckets with sensitive data and public read permissions
- Resources that have a vulnerability or configuration issue and are involved in potential malicious behavior
- Amazon EC2 instances with vulnerabilities and open to the internet
- Amazon Machine Images (AMIs) that are generating the most findings
- AWS resources that don't meet security standards or best practices
- AWS resources associated with potential data exfiltration
- AWS resources associated with unauthorized resource consumption
- AWS users with the most suspicious activity
- S3 buckets with public write or read permissions
- S3 buckets that don't meet security standards or best practices
- S3 buckets with sensitive data

- Credentials that might have leaked
- EC2 instances that allow password authentication on SSH and SSH ports and are open to the internet
- EC2 instances involved in known Tactics, Techniques, and Procedures (TTPs)
- EC2 instances that have missing security patches for important vulnerabilities
- EC2 instances with general unusual behavior
- EC2 instances that have ports accessible from the internet
- EC2 instances that don't meet security standards or best practices
- EC2 instances with anonymized connections
- EC2 instances that are open to the internet
- EC2 instances associated with adversary reconnaissance
- AWS resources associated with malware
- AWS resources associated with cryptocurrency issues
- AWS resources with unauthorized access attempts
- Threat intel indicators with the most hits in the last week

# Findings in AWS Security Hub

AWS provides a highly secure cloud computing environment where you can run your workloads. When you use AWS services, you can also access various security, identity, and compliance tools from AWS and its partners. These tools include firewalls, endpoint, and intrusion detection applications, as well as database security, vulnerability, and compliance scanners. These tools can generate thousands of security findings daily. Findings from these tools might have different finding formats and might be stored and viewed across different platforms.

In this context, it can be difficult to get a complete understanding of your overall security and compliance state. To do so, you would have to either continuously and manually process the output from all of these tools or develop ways to aggregate and analyze the generated findings. With large workloads and environments, processing and analyzing this data can take hundreds of hours of building parsers, transformers, custom compliance rules, and data enrichment pipelines. Even then, the volume of the findings can sometimes be more than you can effectively process. Therefore, it can be difficult to separate potential security issues from noise, to prioritize the findings that matter most to you, and to ensure that you aren't missing any critical findings. AWS Security Hub eliminates this complexity and reduces the effort required to manage and improve the security and compliance of all of your AWS accounts, resources, and workloads.

Security Hub imports findings from AWS security services and from the third-party product integrations that you enable. Security Hub consumes these findings using a standard findings format called AWS Security Finding Format, which eliminates the need for time-consuming data conversion efforts. Security Hub then correlates the findings across integrated products to prioritize the most important ones. For more information about the findings format, see AWS Security Finding Format (p. 35).

**Topics**

- Working with Findings in Security Hub (p. 33)
- AWS Security Finding Format (p. 35)

# Working with Findings in Security Hub

After findings are imported in to Security Hub, you can filter finding results and create insights based on findings.

> **Note**
> Findings are kept for 90 days from the last update to the finding. After 90 days without an update, findings are deleted.

### To view and manage findings

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
2. In the navigation pane, choose **Findings**.

   By default, the **Findings** page lists all of your active findings that Security Hub has processed and generated. Archived findings are not displayed by default.

   The **Record state** filter attribute is preselected by default, and its value is `ACTIVE`. You can update the value of the **Record state** filter attribute to `ARCHIVED` to view only your archived findings. You can also remove this filter attribute to view all of your active and archived findings.
3. To query your findings, use the **Filter** field to select one attribute for the **Group by** aggregator and one or more filter attributes from the available attribute list.

   You can use one of the following attributes as the **Group by** aggregator:

- **Aws account Id**
- **Company name**
- **Compliance status**
- **Generator ID**
- **Malware name**
- **Process name**
- **Threat intel type**
- **Product ARN**
- **Product name**
- **Record state**
- **EC2 instance image ID**
- **EC2 instance IPv4**
- **EC2 instance IPv6**
- **EC2 instance key name**
- **EC2 instance subnet ID**
- **EC2 instance type**
- **EC2 instance VPC ID**
- **IAM access key user name**
- **S3 bucket owner name**
- **Container image ID**
- **Container image name**
- **Container name**
- **Resource ID**
- **Resource type**
- **Severity label**
- **Source URL**
- **Type**
- **Verification state**
- **Workflow state**

You can use all of the AWS Security Finding format's attributes as filters to query through your findings.

> **Note**
> For optional filters, AND logic is applied to your specified collection of filters to query your findings. However, OR logic is applied to multiple filters that use the same attribute set to different values.

For the complete list of AWS Security Finding attributes and their descriptions, see AWS Security Finding Format (p. 35).

4. Choose a finding's title to view the finding's detail pane. In the detail pane, choose the finding ID to view the complete details JSON of that finding.

   > **Note**
   > You can apply an action to a maximum of 20 findings at a time.

5. To apply default (**Archive**) and custom actions to findings, select one or more findings' check boxes. Then expand the **Actions** menu and choose either **Archive** or one of the existing custom actions. When you **Archive** findings, the `RecordState` of the selected findings is set to `ARCHIVED`.

**Note**
You can create Security Hub custom actions to automate Security Hub with Amazon CloudWatch Events. For more information and detailed steps on creating custom actions, see Automating AWS Security Hub with CloudWatch Events (p. 114).

To update finding details, you can also use the UpdateFinding operation of the Security Hub API.

# AWS Security Finding Format

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, thus eliminating the need for time-consuming data conversion efforts. Then it correlates ingested findings across products to prioritize the most important ones.

**Topics**


- Syntax of the AWS Security Finding Format (p. 35)
- Attributes of the AWS Security Finding Format (p. 37)
- Types Taxonomy of the AWS Security Finding (p. 65)


## Syntax of the AWS Security Finding Format

The following is the syntax of the complete finding JSON in the AWS Security Finding Format.

```
"Findings": [
     {
        "AwsAccountId": "string",
  "Compliance": {
     "Status": "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": {
         "AwsEc2Instance": {
            "IamInstanceProfileArn": "string",
            "ImageId": "string",
            "IpV4Addresses": [ "string" ],
            "IpV6Addresses": [ "string" ],
            "KeyName": "string",
            "LaunchedAt": "string",
            "SubnetId": "string",
            "Type": "string",
            "VpcId": "string"
         },
         "AwsIamAccessKey": {
            "CreatedAt": "string",
            "Status": "string",
            "UserName": "string"
         },
         "AwsS3Bucket": {
            "OwnerId": "string",
            "OwnerName": "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": {
     "Normalized": number,
     "Product": number
  },
  "SourceUrl": "string",
  "ThreatIntelIndicators": [
     {
        "Category": "string",
        "LastObservedAt": "string",
        "Source": "string",
        "SourceUrl": "string",
        "Type": "string",
        "Value": "string"
     }
  ],
  "Title": "string",
  "Types": [ "string" ],
  "UpdatedAt": "string",
  "UserDefinedFields": {
     "string" : "string"
  },
  "VerificationState": "string",
  "WorkflowState": "string"
      }
    ]
```

# Attributes of the AWS Security Finding Format

The following table provides descriptions and examples for the AWS Security Finding Format attributes.

| Attribute | Required | Description |
|---|---|---|
| `AwsAccountId` | Yes | The AWS account ID where a finding is generated.<br><br>Type: string (12 digits max)<br><br>Example:<br><br>`"AwsAccountId": "111111111111"` |
| `Compliance` | No | 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 compliance-related finding details.<br><br>Type: object<br><br>Example:<br><br>`"Compliance": { "Status": "PASSED"` |

| Attribute | Required | Description |
| --- | --- | --- |
| | | <pre>}</pre> |
| `Compliance.Status` | No | The result of a compliance check.<br><br>Type: enum<br><br>• Allowed values are the following:<br>  • `PASSED` – Compliance check passed for all evaluated resources.<br>  • `WARNING` – Some information is missing or this check is not supported given your configuration.<br>  • `FAILED` – Compliance check failed for at least one evaluated resource.<br>  • `NOT_AVAILABLE` – Check could not be performed due to a service outage or API error.<br><br>Example:<br><br><pre>"Status": "PASSED"</pre> |
| `Confidence` | No | A finding's confidence. Confidence is defined as the likelihood that a finding accurately identifies the behavior or issue that it was intended to identify. 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.<br><br>Type: integer (range 0–100)<br><br>Example:<br><br><pre>"Confidence": 42</pre> |

| Attribute | Required | Description |
| --- | --- | --- |
| `CreatedAt` | Yes | 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 created.<br><br>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.<br><br>This timestamp *must* be provided on the first generation of the finding and *can't* be changed upon subsequent updates to the finding.<br><br>Type: timestamp<br><br>Example:<br><br>`"CreatedAt": "2017-03-22T13:22:13.933Z"`<br><br>**Note**<br>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. |

| Attribute | Required | Description |
| --- | --- | --- |
| `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.<br><br>Type: integer (range 0–100)<br><br>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.<br><br>When assessing criticality of a finding, consider the following:<br><br>• Does the impacted resource contain sensitive data (e.g., an S3 bucket with PII)?<br>• Does the impacted resource enable an adversary to deepen their access or extend their capabilities to carry out additional malicious activity (e.g., a compromised sysadmin account)?<br>• Is the resource a business-critical asset (e.g., a key business system that if compromised could have significant revenue impact)?<br><br>You can use the following guidelines:<br><br>• A resource powering mission-critical systems or containing highly sensitive data can be scored in the 75–100 range<br>• A resource powering important (but not critical systems) or containing moderately important data can be scored in the 25–75 range<br>• A resource powering non-important systems or containing non-sensitive data *should* be scored in the 0–24 range<br><br>Example:<br><br>`"Criticality": 99` |
| `Description` | Yes | A finding's description. This field can be nonspecific boilerplate text or details that are specific to the instance of the finding.<br><br>Type: string (1,024 characters max)<br><br>Example:<br><br>`"Description": "The version of openssl found on instance i-abcd1234 is known to contain a vulnerability."` |

| Attribute | Required | Description |
| --- | --- | --- |
| `FirstObservedAt` | No | 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.<br><br>Type: timestamp<br><br>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.<br><br>This timestamp should be immutable between updates of the finding record, but can be updated if a more accurate timestamp has been determined.<br><br>Example:<br><br>`"FirstObservedAt": "2017-03-22T13:22:13.933Z"` |
| `GeneratorId` | Yes | 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.<br><br>Type: string (512 characters max) or Amazon Resource Name (ARN)<br><br>Example:<br><br>`"GeneratorId": "acme-vuln-9ab348"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Id` | Yes | The product-specific identifier for a finding.<br><br>Type: string (512 characters max) or ARN<br><br>The finding ID must comply with the following constraints:<br><br>• 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.<br>• You *can't* recycle identifiers regardless of whether the previous finding no longer exists.<br>• 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.<br>• For non-AWS services, the ID *can't* be prefixed with the literal string "arn:".<br>• For AWS services, the ID *must* be the ARN of the finding if one is available. Otherwise, you can use any other unique identifier.<br><br>These constraints are expected to hold within a findings product, but aren't required to hold across findings products.<br><br>Example:<br><br>`"Id": "us-west-2/111111111111/98aebb2207407c87f51e89943f12b1ef"` |
| `LastObservedAt` | No | 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.<br><br>Type: timestamp<br><br>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.<br><br>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.<br>You should update this field to reflect the last or most recently observed timestamp each time a finding is observed.<br><br>Example:<br><br>`"LastObservedAt": "2017-03-23T13:22:13.933Z"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Malware` | No | A list of malware related to a finding.<br><br>Type: array of up to five malware objects<br><br>Example:<br><br>`"Malware": [ { "Name": "Stringler", "Type": "COIN_MINER", "Path": "/usr/sbin/stringler", "State": "OBSERVED" } ]` |
| `Malware.Name` | Yes | The name of the malware that was observed.<br><br>Type: string (64 characters max)<br><br>Example:<br><br>`"Name": "Stringler"` |
| `Malware.Path` | No | The filesystem path of the malware that was observed.<br><br>Type: string (512 characters max)<br><br>Example:<br><br>`"Path": "/usr/sbin/stringler"` |
| `Malware.State` | No | The state of the malware that was observed. Valid values are OBSERVED \| REMOVAL_FAILED \| REMOVED.<br><br>Type: enum<br><br>Example:<br><br>`"State": "OBSERVED"` |
| `Malware.Type` | No | The type of the malware that was observed. Valid values are ADWARE \| BLENDED_THREAT \| BOTNET_AGENT \| COIN_MINER \| EXPLOIT_KIT \| KEYLOGGER \| MACRO \| POTENTIALLY_UNWANTED \| SPYWARE \| RANSOMWARE \| REMOTE_ACCESS \| ROOTKIT \| TROJAN \| VIRUS \| WORM.<br><br>Type: enum<br><br>Example:<br><br>`"Type": "COIN_MINER"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Network` | No | The details of network-related information about a finding.<br><br>Type: object<br><br>Example:<br><br>`"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" }` |
| `Network.DestinationDomain` | No | The destination domain of network-related information about a finding.<br><br>Type: string (128 characters max)<br><br>Example:<br><br>`"DestinationDomain": "there.com"` |
| `Network.DestinationIpV4` | No | The destination IPv4 address of network-related information about a finding.<br><br>Type: IPv4<br><br>Example:<br><br>`"DestinationIpV4": "2.3.4.5"` |
| `Network.DestinationIpV6` | No | The destination IPv6 address of network-related information about a finding.<br><br>Type: IPv6<br><br>Example:<br><br>`"DestinationIpV6": "FE80:CD00:0000:0CDE:1257:0000:211E:729C"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Network.DestinationPort` | No | The destination port of network-related information about a finding.<br><br>Type: number (range of 0–65535)<br><br>Example:<br><br>`"DestinationPort": "80"` |
| `Network.Direction` | No | The direction of network traffic associated with a finding. Valid values are IN \| OUT.<br><br>Type: enum<br><br>Example:<br><br>`"Direction": "IN"` |
| `Network.Protocol` | No | The protocol of network-related information about a finding.<br><br>Type: string (16 characters max)<br><br>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.<br><br>Example:<br><br>`"Protocol": "TCP"` |
| `Network.SourceDomain` | No | The source domain of network-related information about a finding.<br><br>Type: string (128 characters max)<br><br>Example:<br><br>`"SourceDomain": "here.com"` |
| `Network.SourceIpV4` | No | The source IPv4 address of network-related information about a finding.<br><br>Type: IPv4<br><br>Example:<br><br>`"SourceIpV4": "1.2.3.4"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Network.SourceIpV6` | No | The source IPv6 address of network-related information about a finding.<br><br>Type: IPv6<br><br>Example:<br><br>`"SourceIpV6": "FE80:CD00:0000:0CDE:1257:0000:211E:729C"` |
| `Network.SourceMac` | No | The source media access control (MAC) address of network-related information about a finding.<br><br>Type: string (must match `MM:MM:MM:SS:SS:SS`)<br><br>Example:<br><br>`"SourceMac": "00:0d:83:b1:c0:8e"` |
| `Network.SourcePort` | No | The source port of network-related information about a finding.<br><br>Type: number (range of 0–65535)<br><br>Example:<br><br>`"SourcePort": "80"` |
| `Note` | No | A user-defined note that is added to a finding.<br><br>Type: object<br><br>Example:<br><br>`"Note": { "Text": "Don't forget to check under the mat.", "UpdatedBy": "jsmith", "UpdatedAt": "2018-08-31T00:15:09Z" }` |
| `Note.Text` | Yes | The text of a finding note.<br><br>Type: string (512 characters max)<br><br>Example:<br><br>`"Text": "Example text."` |
| `Note.UpdatedAt` | Yes | The timestamp of when the note was updated.<br><br>Type: timestamp<br><br>Example:<br><br>`"UpdatedAt": "2018-08-31T00:15:09Z"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Note.UpdatedBy` | Yes | The principal that created a note.<br><br>Type: string (512 characters max) or ARN<br><br>Example:<br><br>`"UpdatedBy": "jsmith"` |
| `Process` | No | The details of process-related information about a finding.<br><br>Type: object<br><br>Example:<br><br>`"Process": { "Name": "syslogd", "Path": "/usr/sbin/syslogd", "Pid": 12345, "ParentPid": 56789, "LaunchedAt": "2018-09-27T22:37:31Z", "TerminatedAt": "2018-09-27T23:37:31Z" }` |
| `Process.LaunchedAt` | No | The timestamp for the date and time when the process was launched.<br><br>Type: timestamp<br><br>Example:<br><br>`"LaunchedAt": "2018-09-27T22:37:31Z"` |
| `Process.Name` | No | The name of the process.<br><br>Type: string (64 characters max)<br><br>Example:<br><br>`"Name": "syslogd"` |
| `Process.ParentPid` | No | The parent process ID.<br><br>Type: number<br><br>Example:<br><br>`"ParentPid": 56789` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Process.Path` | No | The path to the process executable.<br>Type: string (512 characters max)<br>Example:<br>`"Path": "/usr/sbin/syslogd"` |
| `Process.Pid` | No | The process ID.<br>Type: number<br>Example:<br>`"Pid": 12345` |
| `Process.TerminatedAt` | No | The timestamp for the date and time when the process was terminated.<br>Type: timestamp<br>Example:<br>`"TerminatedAt": "2018-09-27T23:37:31Z"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `ProductArn` | Yes | The ARN generated by Security Hub that uniquely identifies a third-party findings product after the product is registered with Security Hub.<br><br>Type: ARN<br><br>The format of this field is `arn:`*`partition`*`:securityhub:`*`region`*`:`*`account-id`*`:product/`*`company-id`*`/`*`product-id`*.<br><br>• For AWS services that are integrated with Security Hub, the `company-id` must be "aws", 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.<br>• For public products, the `company-id` and `product-id` must be the ID values specified at the time of registration.<br>• For private products, the `company-id` must be the account ID. The `product-id` must be the reserved word "default" or the ID that was specified at the time of registration.<br><br>Example:<br><br>`// Private ARN`<br>`"ProductArn": "arn:aws:securityhub:us-east-1:111111111111:product/111111111111/default"`<br><br>`// Public ARN`<br>`"ProductArn": "arn:aws:securityhub:us-west-2::product/aws/guardduty"`<br>`"ProductArn": "arn:aws:securityhub:us-west-2:222222222222:product/generico/secure-pro"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `ProductFields` | No | A data type where security findings products can include additional solution-specific details that aren't part of the defined AWS Security Finding Format.<br><br>Type: map of up to 50 key/value pairs<br><br>This field shouldn't contain redundant data and must not contain data that conflicts with AWS Security Finding Format fields. The "aws/" prefix represents a reserved namespace for AWS products and services only and must not be submitted with findings from partner products. 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. Fields names can include alphanumeric characters, white space, and the following symbols: _ . / = + \ - @<br><br>Example:<br><br>`"ProductFields": { "generico/secure-pro/Count": "6", "generico/secure-pro/Action.Type", "AWS_API_CALL", "API", "DeleteTrail", "Service_Name": "cloudtrail.amazonaws.com", "aws/inspector/AssessmentTemplateName": "My daily CVE assessment", "aws/inspector/AssessmentTargetName": "My prod env", "aws/inspector/RulesPackageName": "Common Vulnerabilities and Exposures" }` |
| `RecordState` | No | The record state of a finding. Valid values are `ACTIVE` and `ARCHIVED`.<br><br>Type: enum<br><br>By default, findings when initially generated by a service are considered `ACTIVE`. The `ARCHIVED` state indicates that a finding should be hidden from view. Archived findings aren't deleted and remain in the service historically. You can search, review, and report against them at any time.<br><br>Example:<br><br>`"RecordState": "ACTIVE"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `RelatedFindings` | No | A list of related findings.<br><br>Type: array of up to 10 `RelatedFinding` objects<br><br>Example:<br><br>`"RelatedFindings": [ { "ProductArn": "arn:aws:securityhub:us-west-2::product/aws/guardduty", "Id": "123e4567-e89b-12d3-a456-426655440000" }, { "ProductArn": "arn:aws:securityhub:us-west-2::product/aws/guardduty", "Id": "AcmeNerfHerder-111111111111-x189dx7824" } ]` |
| `RelatedFindings.Id` | Yes | The product-generated identifier for a related finding.<br><br>Type: string (512 characters max) or ARN<br><br>Example:<br><br>`"Id": "123e4567-e89b-12d3-a456-426655440000"` |
| `RelatedFindings.ProductArn` | Yes | The ARN of the product that generated a related finding.<br><br>Type: ARN<br><br>Example:<br><br>`"ProductArn": "arn:aws:securityhub:us-west-2::product/aws/guardduty"` |
| `Remediation` | No | The remediation options for a finding.<br><br>Type: object<br><br>Example:<br><br>`"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" } }` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Remediation.Recommendation` | No | A recommendation on how to remediate the issue identified within a finding.<br><br>If the recommendation object is present, either the `Text` or `Url` field must be present and populated, though both can be present and populated. The `Recommendation` field is meant to facilitate manual instructions or details to resolve a finding.<br><br>Type: object<br><br>Example:<br><br>`"Recommendation": { "Text": "Example text.", "Url": "http://myfp.com/recommendations/dangerous_things_and_how_to_fix_them.html" }` |
| `Recommendation.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.<br><br>Type: string (512 characters max)<br><br>Example:<br><br>`"Text": "Example text."` |
| `Recommendation.Url` | No | A URL to link to general remediation information for the finding type of a finding.<br><br>This URL must not require credentials to access. It must be accessible from the public internet and must not expect any context or session.<br><br>Type: URL<br><br>Example:<br><br>`"Url": "http://myfp.com/recommendations/example_domain.html"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Resources` | Yes | A set of resource data types that describe the resources that the finding refers to.<br><br>Type: array of up to 10 resource objects<br><br>Example:<br><br>`"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::111111111111:instance-profile/AdminRole", "VpcId": "vpc-11112222", "SubnetId": "subnet-56f5f633", "LaunchedAt": "2018-05-08T16:46:19.000Z" } } } ]` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Resource.Details` | No | This field provides additional details about the resource through one (and only one) of its subfields. The `Resource.Type` field indicates which of the subfields should contain data. Because this field is optional, there might be a `Resource.Type` value set but no details in any subfields. If a subfield is defined, the `Resource.Type` must match the subfield name. For example, if the `AwsS3Bucket` subfield is populated, the `Resource.Type` must be set to `AwsS3Bucket`.<br><br>You can define only one subfield in a finding. All other subfields that contain finding details (except the subfield defined in `Resource.Type`) are ignored by all processors.<br><br>If multiple resources are present, such as an Amazon EC2 instance and an Amazon S3 bucket, you must create two separate `Resource` objects in the `Resources` field.<br><br>If a resource doesn't fit into one of the existing subfields, you can use the `Other` field to provide the resource details. You must set the `Resource.Type` to `Other` when providing details in the `Other` subfield. You must *not* use the `Other` field when an existing subfield is more appropriate for the resource that you're defining. You must *not* use the `Other` field with any other subfield. For example, when you use the `AwsEc2Instance` field, you must *not* provide additional details about an EC2 instance in the `Other` field. Additionally, you can't use the `Other` field instead of the `AwsEc2Instance` field when providing details about an EC2 instance.<br><br>Type: object<br><br>Example:<br><br>`"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": "da4d66eac431652a4d44d490a00500bded52c97d235b7b4752f9f688566fe6de", "OwnerName": "acmes3bucketowner" }, "Other": [ { "Key": "LightPen", "Value": "blinky" }, { "Key": "SerialNo", "Value": "1234abcd" } ] }` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Resource.Details.AwsEc2Instance` | No | The details of an Amazon EC2 instance.<br><br>Type: object |
| `Resource.Details.AwsEc2Instance.IamInstanceProfileArn` | No | The IAM profile ARN of the instance. |
| `Resource.Details.AwsEc2Instance.ImageId` | No | The Amazon Machine Image (AMI) ID of the instance.<br><br>Type: string (64 characters max) |
| `Resource.Details.AwsEc2Instance.IpV4Addresses` | No | The IPv4 addresses that are associated with the instance.<br><br>Type: array of up to 10 IPv4 addresses |
| `Resource.Details.AwsEc2Instance.IpV6Addresses` | No | The IPv6 addresses that are associated with the instance.<br><br>Type: array of up to 10 IPv6 addresses |
| `Resource.Details.AwsEc2Instance.KeyName` | No | The key name that is associated with the instance.<br><br>Type: string (128 characters max) |
| `Resource.Details.AwsEc2Instance.LaunchedAt` | No | The date and time when the instance was launched.<br><br>Type: timestamp |
| `Resource.Details.AwsEc2Instance.SubnetId` | No | The identifier of the subnet where the instance was launched.<br><br>Type: string (32 characters max) |
| `Resource.Details.AwsEc2Instance.Type` | No | The instance type of the instance. This must be a valid EC2 instance type.<br><br>Type: string (16 characters max) |
| `Resource.Details.AwsEc2Instance.VpcId` | No | The identifier of the VPC where the instance was launched.<br><br>Type: string (32 characters max) |
| `Resource.Details.AwsIamAccessKey` | No | IAM access key details that are related to a finding.<br><br>Type: object |
| `Resource.Details.AwsIamAccessKey.CreatedAt` | No | The creation date and time of the IAM access key that is related to a finding.<br><br>Type: timestamp |
| `Resource.Details.AwsIamAccessKey.Status` | No | The status of the IAM access key that is related to a finding. Valid values are `ACTIVE` and `INACTIVE`.<br><br>Type: enum |
| `Resource.Details.AwsIamAccessKey.UserName` | No | The user associated with the IAM access key that is related to a finding.<br><br>Type: string (128 char max) |

| Attribute | Required | Description |
| --- | --- | --- |
| `Resource.Details.AwsS3Bucket` | No | The details of an Amazon S3 bucket.<br><br>Type: object |
| `Resource.Details.AwsS3Bucket.OwnerId` | No | The canonical user ID of the owner of the Amazon S3 bucket.<br><br>Type: string (64 char max) |
| `Resource.Details.AwsS3Bucket.OwnerName` | No | The display name of the owner of the Amazon S3 bucket.<br><br>Type: string (128 char max) |
| `Resource.Details.Container` | No | Container details that are related to a finding.<br><br>Type: object<br><br>Example:<br><br>`"Container": {`<br>`"Name": "Secret Service Container",`<br>`"ImageId": "image12",`<br>`"ImageName": "SecSvc v1.2 Image",`<br>`"LaunchedAt": "2018-09-29T01:25:54Z"`<br>`}` |
| `Resource.Details.Container.ImageId` | No | The identifier of the image that is related to a finding.<br><br>Type: string (128 characters max) |
| `Resource.Details.Container.ImageName` | No | The name of the image that is related to a finding.<br><br>Type: string (128 characters max) |
| `Resource.Details.Container.LaunchedAt` | No | The date and time that the container was started.<br><br>Type: timestamp |
| `Resource.Details.Container.Name` | No | The name of the container that is related to a finding.<br><br>Type: string (128 characters max) |
| `Resource.Details.Other` | No | The details of a resource that doesn't have a specific subfield for the resource type that is defined under `Resource.Details`. To populate this field, you must set `Resource.Type.Other`.<br><br>Type: map of up to 50 key/value pairs<br><br>For each key/value pair, they key must be less than 128 characters, and the value must be less than 1,024 characters. |

<table>
<tr><th>Attribute</th><th>Required</th><th>Description</th></tr>
<tr><td>Resource.Id</td><td>Yes</td><td>The canonical identifier for the given resource type. For AWS resources that are identified by ARNs, this must be the ARN. For all other AWS resource types that lack ARNs, this must be the identifier as defined by the AWS service that created the resource. For non-AWS resources, this should be a unique identifier associated with the resource.<br><br>Type: string (512 characters max) or ARN<br><br>Example:<br><br><pre>"Id": "arn:aws:s3:::example-bucket"</pre></td></tr>
<tr><td>Resource.Partition</td><td>No</td><td>The canonical AWS partition name that the Region is assigned to.<br><br>Type: enum<br><br>Valid values include the following:<br><br><table><tr><th>Partition</th><th>Description</th></tr><tr><td>aws</td><td>Commercial</td></tr><tr><td>aws-cn</td><td>China</td></tr><tr><td>aws-us-gov</td><td>AWS GovCloud (US)</td></tr></table><br>Example:<br><br><pre>"Partition": "aws"</pre></td></tr>
<tr><td>Resource.Region</td><td>No</td><td>The canonical AWS external Region name where this resource is located.<br><br>Type: string (16 characters max)<br><br>Example:<br><br><pre>"Region": "us-west-2"</pre></td></tr>
</table>

| Attribute | Required | Description |
| --- | --- | --- |
| `Resource.Tags` | No | A list of AWS tags that are associated with a resource at the time the finding was processed. Include the `Resource.Tags` attribute only for resources that have an associated tag. If a resource has no associated tag, don't include a `Resource.Tags` attribute in the finding.<br><br>Type: map of up to 50 tags (values are limited to 256 characters max)<br><br>The following basic restrictions apply to tags:<br><br>• 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 `Resource.Detail.Other` field.<br>• Values are limited to: alphanumeric characters, white space, +, -, =, ., _, :, /, and @.<br>• Values are limited to the AWS Tag value length of 256 characters max.<br><br>Example:<br><br>`"Tags": { "billingCode": "Lotus-1-2-3", "needsPatching": "true" }` |
| `Resource.Type` | Yes | The type of the resource that you're providing details for.<br><br>Type: string (32 characters max)<br><br>Valid values are:<br><br>• `AwsEc2Instance`<br>• `AwsS3Bucket`<br>• `Container`<br>• `AwsIamAccessKey`<br>• `AwsIamUser`<br>• `AwsAccount`<br>• `AwsIamPolicy`<br>• `AwsCloudTrailTrail`<br>• `AwsKmsKey`<br>• `AwsEc2Vpc`<br>• `AwsEc2SecurityGroup`<br>• `Other`<br><br>Example:<br><br>`"Type": "AwsS3Bucket"` |

| Attribute | Required | Description |
| --- | --- | --- |
| `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.<br><br>Type: string (10 characters max, conforms to `YYYY-MM-DD`)<br><br>Example:<br><br>`"SchemaVersion": "2018-10-08"` |
| `Severity` | Yes | A finding's severity.<br><br>Type: object<br><br>Example:<br><br>`"Severity": {`<br>`  "Product": 8.3,`<br>`  "Normalized": 25`<br>`}` |

| Attribute | Required | Description |
| --- | --- | --- |
| `Severity.Normalized` | Yes | The normalized severity of a finding. For findings that supported AWS services generate, Security Hub automatically translates the native severity into the normalized severity based on the following guidance. For findings supported third-party partner products generate, partners can use this guidance to determine the normalized severity required by the AWS Security Finding Format before sending these findings to Security Hub.<br><br>In the AWS Security Finding Format, a finding severity doesn't include consideration of the criticality of the assets that are involved in the activity that resulted in this finding. Findings that are associated with actual data loss or denial of service are considered most severe. Findings that are associated with an active compromise but that don't indicate that data loss or other negative effects have occurred are considered second-most severe. Findings associated with issues that indicate potential for a future compromise are considered third-most severe.<br><br>Severity is scored on a 0–100 basis, using a ratio scale that supports only full integers. This means that when determining the normalized severity, you should assess not only which findings are more severe than others but also how more severe one finding is than another. Zero means that no severity applies (e.g., the severity is "Informational"), and 100 means that the finding has the maximum possible severity. We recommend that you use the following guidance when translating findings' native severity scores to normalized severity for the AWS Security Finding Format:<br><br>• Informational findings (e.g., a finding that is associated with a “Passed” compliance check or a sensitive data identification). Suggested score: 0. These findings should receive a `Low` severity label.<br>• Findings that are associated with issues that could result in future compromises (e.g., vulnerabilities, configuration weaknesses, exposed passwords). This generally aligns to the Software and Configuration Checks namespace under a finding's type. Suggested score: 1–39. These findings should receive a `Low` severity label.<br>• Findings that are associated with issues that indicate an active compromise, but no indication that an adversary has completed their objectives (e.g., 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. These findings should receive a `Medium` severity label.<br>• Findings that are associated with an adversary completing 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. These findings should receive a `High` or `Critical` severity label.<br><br>In Security Hub, the normalized severity scores are available both in their numeric form and in a translated severity label using the following translation table. You can use the severity labels in |

<table>
<tr><th>Attribute</th><th>Required</th><th>Description</th></tr>
<tr><td></td><td></td><td>Filters and Group By statements when managing insights using <code>Severity.Label</code>.<br><br><table><tr><th>Severity Label</th><th>Severity Score Range</th></tr><tr><td><code>Informational</code></td><td>0</td></tr><tr><td><code>Low</code></td><td>1–39</td></tr><tr><td><code>Medium</code></td><td>40–69</td></tr><tr><td><code>High</code></td><td>70–89</td></tr><tr><td><code>Critical</code></td><td>90–100</td></tr></table></td></tr>
<tr><td><code>Severity.Product</code></td><td>No</td><td>The native severity as defined by the finding product that generated the finding.<br><br>Type: number (single-precision 32-bit IEEE 754 floating point number, restricted to finite values)</td></tr>
<tr><td><code>SourceUrl</code></td><td>No</td><td>A URL that links to a page about the current finding in the finding product.<br><br>Type: URL</td></tr>
<tr><td><code>ThreatIntelIndicators</code></td><td>No</td><td>Threat intel details that are related to a finding.<br><br>Type: array of up to five threat intel indicator objects<br><br>Example:<br><br><pre>"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"
  }
]</pre></td></tr>
<tr><td><code>ThreatIntelIndicators.Category</code></td><td>No</td><td>The category of a threat intel indicator. Valid values are BACKDOOR | CARD_STEALER | COMMAND_AND_CONTROL | DROP_SITE | EXPLOIT_SITE | KEYLOGGER.<br><br>Type: enum</td></tr>
<tr><td><code>ThreatIntelIndicators.LastObservedAt</code></td><td>No</td><td>The date and time of the last observation of a threat intel indicator.<br><br>Type: timestamp</td></tr>
<tr><td><code>ThreatIntelIndicators.Source</code></td><td>No</td><td>The source of the threat intel.<br><br>Type: string (64 characters max)</td></tr>
</table>

| Attribute | Required | Description |
| --- | --- | --- |
| `ThreatIntelIndicators.SourceUrl` | No | The URL for more details from the source of the threat intel.<br><br>Type: URL |
| `ThreatIntelIndicators.Type` | No | The type of a threat intel indicator. Valid values are DOMAIN \| EMAIL_ADDRESS \| HASH_MD5 \| HASH_SHA1 \| HASH_SHA256 \| HASH_SHA512 \| IPV4_ADDRESS \| IPV6_ADDRESS \| MUTEX \| PROCESS \| URL.<br><br>Type: enum |
| `ThreatIntelIndicators.Value` | No | The value of a threat intel indicator.<br><br>Type: string (512 characters max) |
| `Title` | Yes | A finding's title. This field can contain nonspecific boilerplate text or details that are specific to this instance of the finding.<br><br>Type: string (256 characters max) |
| `Types` | Yes | One or more finding types in the format of `namespace/category/classifier` that classify a finding.<br><br>Type: array of 50 strings max<br><br>Valid namespace values are Software and Configuration Checks \| TTPs \| Effects \| Unusual Behaviors \| Sensitive Data Identifications.<br><br>• Namespace *must* be a value from the predefined set of namespace values<br>• Category *might* be any value, but it's *recommended* that finding products use categories from the finding type taxonomy in Types Taxonomy of the AWS Security Finding (p. 65)<br>• Classifier *might* be any value, but it's *recommended* that FPs use the identifier verbatim defined by published standards whenever possible<br><br>Namespaces are required for all finding types, but categories and classifiers are optional. If you specify a classifier is specified, you must also specify a category. The '/' character is reserved and must *not* be used in a category or classifier. Escaping the '/' character isn't supported.<br><br>Example:<br><br>`"Types": [ "Software and Configuration Checks/Vulnerabilities/CVE" ]` |

| Attribute | Required | Description |
| --- | --- | --- |
| `UpdatedAt` | Yes | An ISO8601-formatted timestamp (as defined in RFC-3339 Date and Time on the Internet: Timestamps) 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.<br><br>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.<br><br>Type: timestamp<br><br>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. |
| `UserDefinedFields` | No | 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.<br><br>Type: map of up to 50 key/value pairs<br><br>Example:<br><br>`"UserDefinedFields": { "reviewedByCio": "true", "comeBackToLater": "Check this again on Monday" }` |

| Attribute | Required | Description |
| --- | --- | --- |
| `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 have investigated a finding.<br><br>Type: enum<br><br>Valid values are as follows:<br><br>• `UNKNOWN` – The default disposition of a security finding unless a user changes it<br>• `TRUE_POSITIVE` – A user sets this value if the security finding has been confirmed<br>• `FALSE_POSITIVE` – A user sets this value if the security finding has been determined to be a false alarm<br>• `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 |

| Attribute | Required | Description |
|---|---|---|
| `WorkflowState` | No | 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.<br><br>Type: enum<br><br>Valid values are as follows:<br><br>• `NEW` – This can be associated with findings in the `Active` record state. This is the default workflow state for any new finding.<br>• `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.<br>• `IN_PROGRESS` – This can be associated with findings in the `Active` record state. Team members are actively working on the finding.<br>• `RESOLVED` – This can be associated with findings in the `Archive` 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.<br>• `DEFERRED` – This can be associated with findings in the `Archive` 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.<br>• `DUPLICATE` – This can be associated with findings in the `Archive` record state, and it means that the finding is a duplicate of another finding.<br><br>Example:<br><br>`"WorkflowState": "NEW"` |

# Types Taxonomy of the AWS Security Finding

The following information describes the first three levels of the `Types` path. The top-level bullets are namespaces, the second-level bullets are categories, and the third-level bullets (shown only for Software and Configuration Checks) are `Classifiers`.

- Namespaces
  - Categories
    - Classifiers

Findings products can define classifiers. A findings product might define a partial path. For example, the following finding types are all valid: TTPs, TTPs/Defense Evasion, and TTPs/Defense Evasion/CloudTrailStopped.

TTPs stands for Tactics, Techniques, and Procedures. The TTP categories in the following list align to the MITRE ATT&CK MatrixTM. Unusual Behaviors are different from TTPs because they reflect general unusual behavior (e.g., general statistical anomalies) and aren't 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
    - 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)
    - C5 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

This section lists the AWS services and third-party products that you can integrate with AWS Security Hub and instructions for enabling them. It also provides instructions for importing findings that are generated from your own custom security products.

> **Important**
> Security Hub detects 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 doesn't retroactively detect and consolidate security findings that were generated before you enabled Security Hub.

**Topics**


- AWS Product Integrations (p. 68)
- Third-Party Partner Product Integrations (p. 69)
- Custom Product Integrations (p. 72)


## AWS Product Integrations

Security Hub consolidates security findings generated by the following AWS services:

- Amazon GuardDuty
- Amazon Inspector
- Amazon Macie

To integrate these services with Security Hub, you just need to enable them in your account on the console for each service. A resource policy that allows Security Hub to get findings from these services is automatically created and applied. You do not need to configure any settings to start getting findings from them. After you enable them, Security Hub immediately starts collecting findings in that account from these services. If you don't have a supported AWS product enabled, or the integration is not enabled in Security Hub, no findings are sent to Security Hub. You can verify whether a product integration is enabled on the **Integrations** page of the Security Hub console.

With GuardDuty, Security Hub imports GuardDuty findings of all of the supported finding types. New findings from GuardDuty are sent to Security Hub within 5 minutes. Updates to findings are sent based on the **Updated findings** setting for CloudWatch Events in GuardDuty settings. When you generate GuardDuty sample findings using the GuardDuty **Setting** page, Security Hub ingests the sample findings and omits the prefix '[Sample]' in the finding type. For example, the sample finding type in GuardDuty "[SAMPLE] Recon:IAMUser/ResourcePermissions" is displayed as "Recon:IAMUser/ResourcePermissions" in Security Hub. For more information about GuardDuty findings, see Amazon GuardDuty Findings.

With Amazon Inspector, Security Hub imports Amazon Inspector findings that are generated through assessment runs based on all supported rules packages. For more information about Amazon Inspector rules packages and rules, see Amazon Inspector Rules Packages and Rules.

With Macie, a finding (currently known as an alert) can be one of the following indices: **CloudTrail data**, **S3 bucket properties**, and **S3 objects**. For more information, see Locating and Analyzing Macie Alerts.

Security Hub imports Macie basic and custom alerts (findings) only from the **S3 bucket properties** and **S3 objects** indices. Macie does not send data classifications. Security Hub does *not* import Macie findings from the **CloudTrail data** index.

# Third-Party Partner Product Integrations

After you enable Security Hub, you can configure it to import (via automatic or manual importing) findings from the following third-party product integrations.

| Company name | Product name | Product ARN | Product description |
|---|---|---|---|
| Alert Logic | SIEMless ThreatManagement | arn:aws:securityhub:<REGION>::product/alertlogic/althreatmanagement | Get the right level of coverage: vulnerability and asset visibility, threat detection and incident management, WAF, and assigned SOC analyst options. |
| ARMOR | Armor Anywhere | arn:aws:securityhub:<REGION>::product/armordefense/armoranywhere | Armor Anywhere is a managed security and compliance for AWS. |
| Barracuda Networks | Cloud Security Guardian | arn:aws:securityhub:<REGION>::product/barracuda/cloudsecurityguardian | Barracuda Cloud Security Guardian helps organizations stay secure while building applications in, and moving workloads to, the public cloud. |
| Checkpoint | CloudGuard IaaS | arn:aws:securityhub:<REGION>::product/checkpoint/cloudguard-iaas | CloudGuard IaaS transparently extends comprehensive threat prevention security to AWS while protecting assets in the cloud. |
| Checkpoint | Dome9 Arc | arn:aws:securityhub:<REGION>::product/checkpoint/dome9-arc | Dome9 Arc delivers verifiable cloud network security, advanced IAM protection, and comprehensive compliance and governance. |
| CrowdStrike | CrowdStrike Falcon | arn:aws:securityhub:<REGION>::product/crowdstrike/crowdstrike-falcon | CrowdStrike Falcon's single lightweight sensor unifies next-generation antivirus, endpoint detection and response, and 24/7 managed hunting via the cloud. |
| CyberArk | Privileged Threat Analytics | arn:aws:securityhub:<REGION>::product/cyberark/cyberark-pta | Privileged Threat Analytics collect, detect, alert, and respond to high-risk activity and behavior of privileged accounts to contain in-progress attacks. |
| F5 Networks | Advanced WAF | arn:aws:securityhub:<REGION>::product/f5networks/f5-advanced-waf | Advanced WAF provides malicious bot protection, L7 DoS mitigation, API inspection, behavior analytics, and more to defend against web app attacks. |
| GuardiCore | Centra 4.0 | arn:aws:securityhub:<REGION>::product/guardicore/guardicore | Centra provides flow visualization, micro-segmentation, and breach detection for workloads in modern data centers and clouds. |
| GuardiCore | Infection Monkey | arn:aws:securityhub:<REGION>::product/guardicore/aws-infection-monkey | Infection Monkey is an attack/simulation tool designed to test networks against attackers. |

| Company name | Product name | Product ARN | Product description |
|---|---|---|---|
| IBM | QRadar SIEM | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/ibm/qradar-siem | IBM QRadar SIEM provides security teams with the ability to quickly and accurately detect, prioritize, investigate, and respond to threats. |
| Imperva | Attack Analytics | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/imperva/imperva-attack-analytics | Imperva Attack Analytics correlates and distills thousands of security events into a few readable security incidents. |
| McAfee | MVISION Cloud for AWS | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/mcafee-skyhigh/mcafee-mvision-cloud-aws | MVISION Cloud for Amazon Web Services is a comprehensive monitoring, auditing, and remediation solution for your AWS environment. |
| Palo Alto Networks | Redlock | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/paloaltonetworks/redlock | Protect your AWS deployment with cloud security analytics, advanced threat detection, and compliance monitoring. |
| Qualys | Vulnerability Management | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/qualys/qualys-vm | Qualys Vulnerability Management (VM) continuously scans and identifies vulnerabilities, protecting your assets. |
| Rapid7 | InsightVM | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/rapid7/insightvm | Rapid7 InsightVM provides vulnerability management for modern environments, allowing you to efficiently find, prioritize, and remediate vulnerabilities. |
| Sophos | Server Protection | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/sophos/sophos-server-protection | Sophos Server Protection defends the critical applications and data at the core of your organization, using comprehensive defense-in-depth techniques. |
| Splunk | Splunk Enterprise | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/splunk/splunk-enterprise | Splunk uses Amazon CloudWatch Events as a consumer of Security Hub findings. Send your data to Splunk for advanced security analytics and SIEM. |
| Sumo Logic | Machine Data Analytics | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/sumologicinc/sumologic-mda | Sumo Logic's hosted, machine data analytics platform that enables DevSecOps teams build, run, and secure their AWS applications. |
| Symantec | Cloud Workload Protection | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/symantec-corp/symantec-cwp | Cloud Workload Protection provides complete protection for your Amazon EC2 instances with anti-malware, intrusion prevention, and file integrity monitoring. |
| Tenable | Tenable.io | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/tenable/tenable-io | Understand, investigate, and prioritize vulnerabilities. Managed in the Cloud. |
| Turbot | Turbot | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/turbot/turbot | Turbot ensures that your cloud infrastructure is secure, compliant, scalable, and cost optimized. |
| Twistlock | Enterprise Edition | arn:aws:securityhub:<REGION>:<PRODUCT_ID>:product/twistlock/twistlock-enterprise | Twistlock is a cloud native cybersecurity platform that protects VMs, containers, and serverless platforms. |

The following partner products only receive findings and do not have a Product ARN:

| Company name | Product name | Product description |
|---|---|---|
| Palo Alto Networks | Demisto Enterprise AMI | Demisto is a Security Orchestration, Automation, and Response (SOAR) platform that integrates with your entire security product stack to accelerate incident response and security operations. |
| PagerDuty | PagerDuty | 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. 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. |
| Splunk | Splunk Phantom | 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. |
| Rapid7 | InsightConnect | Rapid7's InsightConnect is a security orchestration and automation solution that enables your team to optimize SOC operations with little to no code. |
| Atlassian | Ops Genie | Opsgenie is a modern incident management solution for operating always-on services, empowering Dev & Ops teams to plan for service disruptions and stay in control during incidents. Integrating with Security Hub will ensure mission critical security related incidents are routed to the appropriate teams for immediate resolution. |
| ServiceNow | ITSM | The ServiceNow Security Hub integration allows security findings from Security Hub to be viewed within ServiceNow ITSM. |
| ServiceNow | SecOps | The ServiceNow Security Hub integration allows both automated and manual forwarding of security findings from Security Hub to ServiceNow Security Operations. |
| Slack | Slack | Slack is a layer of the business technology stack that brings together people, data, and applications – 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. |

### To subscribe to a partner product

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
2. Choose **Integrations** and then locate the product to integrate with Security Hub.
3. Choose **Purchase** to open AWS Marketplace. In AWS Marketplace, choose **Continue to Subscribe**.

**Note**
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.

4. Choose **Subscribe**.

After you subscribe to a product, you need to enable the integration with Security Hub. 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 import findings from that product.

**To enable Security Hub integration with the partner product**

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
2. Locate the product to enable and then choose **Enable**.

   You must have a subscription to the product to successfully integrate it with Security Hub.
3. To review the configuration information from the company that creates the product, choose **Configuration instructions**.
4. Review the policy that is assigned to the product subscription and then choose **Enable**.

# Custom Product Integrations

In addition to findings generated by the integrated AWS service and third-party product, Security Hub can also consume findings that are generated by various custom security products you may use. You can import these findings into Security Hub manually using the BatchImportFindings API operation.

Follow these instructions when invoking the `BatchImportFindings` API operation to import findings generated by custom security products:

- You must provide the finding details using the AWS Security Finding format.
- You must enable Security Hub before you can successfully invoke the `BatchImportFindings` API operation.
- 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.
- We recommend that you use the ProductFields attribute to define the name of the product that generates the findings that you're importing. For example, if you're integrating Cloud Custodian with Security Hub, you could use the following values.

```
"ProductFields":
  {
    "ProviderName": "CloudCustodian",
    "ProviderVersion": "0.8.32.1",
  }
```

**Note**
Cloud Custodian is a flexible rules engine that is commonly used as a solution for automated security, compliance, and cost management in the cloud. For more information about

integrating Cloud Custodian with Security Hub, see Announcing Cloud Custodian Integration with AWS Security Hub on the AWS Open Source Blog.

- You must supply, manage, and increment your own finding IDs, using the Id attribute. Each new finding must have a unique finding ID.
- You must specify your own account ID, using the AwsAccountId attribute.
- You must supply your own timestamps for the CreatedAt and UpdatedAt attributes.
- In addition to importing new findings from custom products, you can also update existing findings from custom products using the `BatchImportFindings` API operation. To update existing findings, use the existing finding ID (via the Id attribute) while resending the full finding with the appropriate information updated in the request, including a modified UpdatedAt timestamp.

# Compliance Standards: CIS AWS Foundations

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 compliance rules in the supported security standards. These checks provide a compliance score and identify specific accounts and resources that require attention.

In this release, 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

## Enabling the CIS AWS Foundations Standard in Security Hub

After you enable Security Hub in a particular AWS account and Region, the CIS AWS Foundations standard in that account and Region is automatically enabled.

After the CIS AWS Foundations standard is enabled in Security Hub in a particular account and Region, it begins running checks on your environment's resources in that account and Region against the compliance rules in the standard. Then Security Hub generates findings based on the results of these checks.

**Important**
Cross-Region processing isn't supported for the CIS AWS Foundations standard in Security Hub. In other words, if you enable Security Hub (and consequently this standard in Security Hub) in one Region and a resource that it checks is located in another Region, the return value for such check is Failed. For example, if you're storing your AWS CloudTrail logs in an Amazon S3 bucket in the us-east-2 Region and the CIS AWS Foundations standard is running in Security Hub enabled in us-west-2, checks 2.3 (Ensure the S3 bucket CloudTrail logs to is not publicly accessible) and 2.6 (Ensure S3 bucket access logging is enabled on the CloudTrail S3 bucket) are returned as Failed.
You must enable Security Hub is all Regions to be fully compliant with CIS AWS Foundations Benchmark checks.

## How the CIS AWS Foundations Standard in Security Hub Uses AWS Config

To run the CIS AWS Foundations standard's compliance checks 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. Therefore, for the CIS AWS Foundations standard

to be functional in Security Hub, when you enable it in a particular account, you must also enable AWS Config in that account.

Security Hub doesn't 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 by using the AWS CloudFormation "Enable AWS Config" template in AWS CloudFormation StackSets Sample Templates.

When you enable Security Hub in a particular account, you also, by default, enable the supported CIS AWS Foundations standard in that account. In other words, after you enable Security Hub, it immediately begins running checks on your environment's resources against the compliance rules in the now-enabled CIS AWS Foundations standard. Then Security Hub generates findings based on the results of these checks.

To run CIS AWS Foundations standard's compliance checks on your environment's resources, Security Hub uses AWS Config rules to evaluate the configuration settings of your AWS resources. AWS Config rules represent your ideal resource configuration settings. Therefore, when you enable Security Hub in a particular account, you must also enable AWS Config in that account. After Security Hub and AWS Config are enabled, Security Hub automatically creates the requisite infrastructure of AWS Config rules that it needs to run the CIS AWS Foundations standard's compliance checks.

> **Note**
> If you're working with a Security Hub master account, enable AWS Config in each of this master account's Security Hub member accounts.

> **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*.

> **Important**
> If you enable AWS Config in your Security Hub master account, this doesn't automatically enable AWS Config in the Security Hub member accounts for this master account. If you want Security Hub to generate findings against the compliance rules in the CIS AWS Foundations standard for the resources in a Security Hub member account, you must enable AWS Config in that member account.

After the CIS AWS Foundations standard is enabled, Security Hub automatically creates the requisite infrastructure of AWS Config rules that it needs to run the standard's compliance checks. For every check that uses a specific AWS Config managed rule or rules, Security Hub creates an instance of that rule or rules specific to Security Hub (even if another instance of this rule already exists) in your AWS environment. For information about which specific AWS Config managed rules the CIS AWS Foundations standard in Security Hub uses, see CIS AWS Foundations Standard Checks Supported in Security Hub (p. 76).

> **Note**
> The limit for the AWS Config managed rules is 150 rules per account per Region. However, when you enable the CIS AWS Foundations standard in Security Hub, the service-linked AWS Config rules that are automatically created do not count towards the 150 rule limit. You can enable these compliance checks even if you already have 150 AWS Config rules in your account.

# AWS Config Resources Required for CIS Checks

If you don't enable all resources in AWS Config, a finding is generated for the check 2.5 – Ensure AWS Config is enabled in all Regions (p. 90). For other checks, you must enable the following resources in AWS Config for Security Hub to accurately report findings based on the CIS AWS Foundation standard checks:

- AwsEc2Instance
- AwsS3Bucket

- Container
- AwsIamAccessKey
- AwsIamUser
- AwsAccount
- AwsIamPolicy
- AwsCloudTrailTrail
- AwsKmsKey
- AwsEc2Vpc
- AwsEc2SecurityGroup

When you view the details of a compliance standards rule that is based on an AWS Config rule, you can choose **Compliance rules** to open the AWS Config rule associated with the compliance check. Only compliance checks that are based on AWS Config rules provide links to the AWS Config rule.

# Results of Standards Checks in Security Hub

Security Hub uses the AWS Security Finding Format (p. 35) format for the findings that it generates as the result of running checks against the compliance rules included in the enabled standards. For these findings, the AWS Security Finding format includes a special **Compliance** field that contains the standard's compliance-related findings details, including the results of the checks that Security Hub ran. The possible return values for a standard check are Passed, Failed, Warning (if Security Hub or AWS Config can't complete the check), and Not available (if the service whose resources are being checked isn't available). If all resources in the Security Hub master account and across all member accounts passed a given check, the rule that this check used is considered **Compliant**. If one or more resources in the Security Hub master account, across member accounts, or both failed a given check or received a warning about it, the rule that this check used is considered **Noncompliant**.

The **Compliance** field displays the result of the most recent check that Security Hub ran against a given rule. The results of the previous checks are kept in an archived state for 90 days. If a subsequent check against a given rule generates a new result (for example, the status of "Avoid the use of the root account" changed from Failed to Passed), a new finding that contains the most recent result is generated. 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 starts running the standards checks within 2 hours after the CIS AWS Foundations standard is enabled. The checks run again automatically within 12 hours from the latest check.

Security Hub supports both periodic and change-triggered compliance checks. Periodic checks are automatically run again within 12 hours after the latest run. Change-triggered checks are run when the resource associated with the check has any state changes. For any Security Hub compliance check based on a managed AWS Config rule, you can click through to that rule to see whether it is change triggered or periodic. In general, Security Hub leverages change triggered rules whenever possible, but there must be Config Configuration Item support for the resource to use a change triggered rule. Security Hub's compliance checks that leverage Security Hub's own custom lambda functions are always periodic. Periodicity cannot currently be changed.

# CIS AWS Foundations Standard Checks Supported in Security Hub

The following are the CIS AWS Foundations standard's compliance checks that are supported in this release of Security Hub.

**Important**
You can disable the entire CIS AWS Foundations standard and thus stop Security Hub from running checks against its rules and generating findings based on those checks. You *can't* disable individual rules in the CIS AWS Foundations standard.

# 1.1 – Avoid the use of the "root" account

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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 fingings for 3.3 – Ensure a log metric filter and alarm exist for usage of "root" account (p. 95).

### To create an Amazon SNS topic

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

### To create a metric filter and alarm

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{$.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

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.

To run this check, Security Hub uses the mfa-enabled-for-iam-console-access AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

> **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 CIS Standards checks may not be accurate. It may also take up to 4 hours after remediating this issue for the check to report compliance.

## 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, see How to Delegate Management of Multi-Factor Authentication to AWS IAM Users on the AWS Security Blog.

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

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.

To run this check, Security Hub uses the iam-user-unused-credentials-check AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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.

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

1. Open the IAM console at https://console.aws.amazon.com/iam/.
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

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.

To run this check, Security Hub uses the access-keys-rotated AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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.
4. Choose **Security credentials**.
5. To create a new key for the user, choose **Create access key**. Then either download the secret access key or choose **Show** and then copy it from the page. Store it in a secure location to provide to the user and then 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. To delete it, choose the **X** at the end of the row and then choose **Delete**. After you delete the access key, you can't recover it.

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

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.

To run this check, Security Hub uses the iam-password-policy AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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.

To run this check, Security Hub uses the iam-password-policy AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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.

To run this check, Security Hub uses the iam-password-policy AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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.

To run this check, Security Hub uses the iam-password-policy AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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.

To run this check, Security Hub uses the iam-password-policy AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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.

To run this check, Security Hub uses the iam-password-policy AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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. Additionally, requiring regular password changes 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.

To run this check, Security Hub uses the iam-password-policy AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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. Additionally, removing the root access keys encourages the creation and use of role-based accounts that are least privileged.

To run this check, Security Hub uses the iam-root-access-key-check AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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, we recommend 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.

To run this check, Security Hub uses the root-account-mfa-enabled AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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**.
4. Choose **Multi-factor authentication (MFA)**.
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. 84).

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

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, we recommend 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.

> **Note**
> Using hardware MFA for many, many accounts might create a logistical device management issue. If this is the case, consider implementing this Level 2 recommendation selectively to the highest security accounts and the Level 1 recommendation applied to the remaining accounts.

To run this check, Security Hub uses the root-account-hardware-mfa-enabled AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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**.
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.

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

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.

To run this check, Security Hub uses the iam-user-no-policies-check AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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.

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

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": "*".

To run this check, Security Hub uses the iam-policy-no-statements-with-admin-access AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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

To run this check, Security Hub uses the multi-region-cloud-trail-enabled AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

# 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. 88) .
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

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.

To run this check, Security Hub uses the cloud-trail-log-file-validation-enabled AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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 and being used. For example, if you are using Security Hub in the us-east-2 Region, and you are storing CloudTrail logs in a bucket in the us-west-2 Region, Security Hub cannot find the bucket in a the us-west-2 Region. 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 fails for all accounts except the account that owns the bucket.

To run this check, Security Hub uses the s3-bucket-public-read-prohibited and s3-bucket-public-write-prohibited AWS Config managed rules. After the CIS AWS Foundations standard is enabled, an instance of each of these rules, specific to Security Hub, is created in your AWS environment.

## 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

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.

To run this check, Security Hub uses the cloud-trail-cloud-watch-logs-enabled AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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. 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 in all Regions

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.

To run this check, Security Hub runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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

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 and being used. For example, if you are using Security Hub in the us-east-2 Region, and you are storing CloudTrail logs in a bucket in the us-west-2 Region, Security Hub cannot find the bucket in a the us-west-2 Region. 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 fails for all accounts except the account that owns the bucket.

To run this check, Security Hub uses the s3-bucket-logging-enabled AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## Remediation

**To enable S3 bucket access logging**

1. Open the Amazon S3 console at 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

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.

To run this check, Security Hub uses the cloud-trail-encryption-enabled AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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.

To run this check, Security Hub uses the cmk-backing-key-rotation-enabled AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## Remediation

**To enable CMK rotation**

1. Open the AWS KMS console at https://console.aws.amazon.com/kms.
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

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.

To run this check, Security Hub uses the vpc-flow-logs-enabled AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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**

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.

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.

```
{($.errorCode="*UnauthorizedOperation") || ($.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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{($.eventName="ConsoleLogin") && ($.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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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**

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

Make a note of the associated log group name.

**To create a metric filter and alarm**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{$.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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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**

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{($.eventName=DeleteGroupPolicy) || ($.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**.

   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-3.4-IAMPolicyChanges`**.
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**.

# 3.5 – Ensure a log metric filter and alarm exist for CloudTrail configuration changes

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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**

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

# 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

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

### To create a metric filter and alarm

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

   ```
   {($.eventName=ConsoleLogin) && ($.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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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**

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{($.eventSource=kms.amazonaws.com) && (($.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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

### To create a metric filter and alarm

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{($.eventSource=s3.amazonaws.com) && (($.eventName=PutBucketAcl)
 || ($.eventName=PutBucketPolicy) || ($.eventName=PutBucketCors) ||
 ($.eventName=PutBucketLifecycle) || ($.eventName=PutBucketReplication)
 || ($.eventName=DeleteBucketPolicy) || ($.eventName=DeleteBucketCors) ||
 ($.eventName=DeleteBucketLifecycle) || ($.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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

### To create a metric filter and alarm

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{($.eventSource=config.amazonaws.com) && (($.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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

### To create a metric filter and alarm

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.

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.

```
{($.eventName=AuthorizeSecurityGroupIngress) ||
 ($.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)

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{($.eventName=CreateNetworkAcl) || ($.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**.

   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-3.11-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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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**

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{($.eventName=CreateCustomerGateway) || ($.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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

## 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**

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

**To create a metric filter and alarm**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{($.eventName=CreateRoute) || ($.eventName=CreateRouteTable) ||
 ($.eventName=ReplaceRoute) || ($.eventName=ReplaceRouteTableAssociation)
 || ($.eventName=DeleteRouteTable) || ($.eventName=DeleteRoute) ||
 ($.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

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 runs through the exact audit steps prescribed for it in Securing Amazon Web Services. No AWS Config managed rules are created in your AWS environment for this check.

# 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

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
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. 86).

   Make a note of the associated log group name.

### To create a metric filter and alarm

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{($.eventName=CreateVpc) || ($.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

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.

To run this check, Security Hub uses the restricted-ssh AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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.

To run this check, Security Hub uses the restricted-common-ports AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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

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, 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.

To run this check, Security Hub uses the vpc-default-security-group-closed AWS Config managed rule. After the CIS AWS Foundations standard is enabled, an instance of this rule, specific to Security Hub, is created in your AWS environment.

## 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 Standard Checks That Aren't Supported in Security Hub

The following are the compliance rules that are *not* supported in the CIS AWS Foundations standard in Security Hub:

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

# 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. 29).


```
{
    "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-2:0123456789010:insight/custom/f4c4890b-
ac6b-4c26-95f9-e62cc46f3055"
    },
    "requestID": "c0fffccd-f04d-11e8-93fc-ddcd14710066",
    "eventID": "3dabcebf-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 AWS SMS queue

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

Security Hub supports three types of integration with CloudWatch Events. First, Security Hub automatically sends all findings to CloudWatch Events as events. You can define rules in CloudWatch Events that automatically route generated 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.

Second, Security Hub also sends findings associated with custom actions to CloudWatch 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. 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. For example, you can create a custom action in Security Hub that sends findings to a ticketing system with a custom action ID set to "send_to_ticketing". Then in CloudWatch Events, create a rule that is triggered for any finding received that includes a custom action ID of "send_to_ticketing". The rule in CloudWatch Events includes logic to send the finding to your ticketing system. You can then also select findings within Security Hub and use the custom action in Security Hub to manually send findings to your ticketing system.

Third, you can also use custom actions to send a set of insight results to CloudWatch Events. For example, if you see a particular insight result of interest that you want to share with a colleague, you can send that insight result to the colleague via a chat or ticketing system using custom actions.

> **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.

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.

# 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 an event is received, such as a finding from Security Hub.

**To create a CloudWatch Events rule for a Security Hub finding**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{
  "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 configure Security Hub and CloudWatch Events to send Security Hub findings that are associated with custom actions to CloudWatch Events, complete the following procedures.

**To create a custom action in Security Hub**

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
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** because 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**

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
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.

```
{
  "source": [
    "aws.securityhub"
  ],
  "detail-type": [
    "Security Hub Findings - Custom Action"
  ],
  "resources": [
    "arn:aws:securityhub:us-west-2:123456789012:action/custom/test-action1"
  ]
}
```

7. Replace the ARN listed in the `resources` section with the **Custom Action ARN** for the custom action you created. This value is displayed on the **Custom actions** page.
8. Choose **Save** to close the window.
9. Choose **Add target** and then select the target to invoke when this rule is matched.
10. Choose **Configure details**.
11. 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**.
12. Choose **Create rule**.

After this rule is created in CloudWatch Events, when you perform a custom action on findings in your account, events are generated in CloudWatch Events.

The format of the example pattern for associating custom actions with findings generated by insights is:

```
{
  "source": [
    "aws.securityhub"
  ],
  "detail-type": [
    "Security Hub Insight Results"
  ],
  "resources": [
    "arn:aws:securityhub:us-west-2:123456789012:action/custom/test-action1"
  ]
}
```

# CloudWatch Events Formats for Security Hub

Security Hub aggregates findings from enabled AWS services (Amazon GuardDuty, Amazon Inspector, and Amazon Macie) and from supported AWS partner products. Security Hub also consolidates findings

into insights that identify security areas that require attention or intervention. Security Hub also conducts automated and continuous compliance checks using CloudWatch Events best practices and supported industry standards (such as CIS Foundations Benchmarks).

The CloudWatch event for Security Hub findings is in the following format.

```
{
   "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":[
      "arn:aws:securityhub:us-west-2::product/aws/macie/arn:aws:macie:us-
west-2:111122223333:integtest/trigger/6294d71b927c41cbab915159a8f326a3/alert/f2893b211841"
   ],
   "detail":{
      "findings": [AMAZON_FINDING_JSON]
   }
}
```

The CloudWatch event for Security Hub aggregated findings with a custom action is in the following format.

```
{
  "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": [
    "arn:aws:securityhub:us-west-1:111122223333:action/custom/custom-action-name"
  ],
  "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 (p. 35).

The CloudWatch Events event for Security Hub insights results has the following format.

```
{
  "version": "0",
  "id": "1a1111a1-b22b-3c33-444d-5555e5ee5555",
  "detail-type": "Security Hub Insight Results",
  "source": "aws.securityhub",
  "account": "111122223333",
  "time": "2017-12-22T18:43:48Z",
  "region": "us-west-1",
  "resources": [
      "arn:aws:securityhub:us-west-1:111122223333::product/aws/macie:us-
west-1:222233334444:test/trigger/1ec9cf700ef6be062b19584e0b7d84ec/alert/f2893b211841"
  ],
  "detail": {
```

```
    "actionName":"name of the action",
    "actionDescription":"description of the action",
    "insightArn":"ARN of the insight",
    "insightName":"Name of the insight",
    "resultType":"ResourceAwsIamAccessKeyUserName",
    "number of results":"number of results, max of 100",
    "insightResults": [
        {"result 1": 5},
        {"result 2": 6}
    ]
  }
}
```

# 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.

**To send findings to CloudWatch Events**

1. On the Security Hub console, choose **Findings**.
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 **Successfull sent findings to Amazon CloudWatch Events** is displayed.

**To send insight results to CloudWatch Events**

1. On the Security Hub console, choose **Insights**.
2. On the **Insights** page , choose the insight that incldues 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 **Successfull sent findings to Amazon CloudWatch Events** is displayed.

# Disabling AWS Security Hub

You can use the AWS Security Hub console or the `DisableSecurityHub` operation of the Security Hub API to disable Security Hub. If you disable Security Hub, your existing findings and insights and any Security Hub configuration settings are deleted after 90 days and can't be recovered. Any enabled standards are disabled, and your master and member account relationships are removed. 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. 27).

**To disable Security Hub (console)**

1. Open the AWS Security Hub console at https://console.aws.amazon.com/securityhub/.
2. In the navigation pane, under **Settings**, choose **General**.
3. Choose **Disable AWS Security Hub**, then choose **Disable AWS Security Hub** again.

   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.

| update-history-change | update-history-description | update-history-date |
| --- | --- | --- |
| Updates to Terminology and Concepts | Updated some descriptions and added new terms to Terminology and Concepts. | September 21, 2019 |
| AWS Security Hub general availability release (p. 120) | Content updates to reflect improvements made to Security Hub during the preview period. | June 25, 2019 |
| Added remediation steps for CIS AWS Foundations checks | Added remediation steps to Standards Supported in AWS Security Hub. | April 15, 2019 |
| Preview release of AWS Security Hub (p. 120) | Published the preview release version of the *AWS Security Hub User Guide*. | November 18, 2018 |