Incident Manager

User Guide

Incident Manager: User Guide

Table of Contents

What Is AWS Systems Manager Incident Manager?

AWS Systems Manager Incident Manager is an incident management console designed to help users mitigate and recover from incidents affecting their AWS-hosted applications. An incident is any unplanned interruption or reduction in quality of services.

Incident Manager increases incident resolution by notifying responders of impact, highlighting relevant troubleshooting data, and providing collaboration tools to get services back up and running. To achieve the primary goal of reducing the time-to-resolution of critical incidents, Incident Manager automates response plans and enables responder team escalation.

Using AWS tools such as Amazon CloudWatch Alarms and CloudWatch Metrics, AWS CloudTrail, AWS Systems Manager, AWS Chatbot, and more, Incident Manager facilitates rapid incident response to get applications working again.

Features include:

- **Response plans** – Create and automate response plans initiated by CloudWatch alarms or Amazon EventBridge events.
- **Runbook automation** – Define runbooks through Systems Manager Automation to automate critical response and provide detailed steps to first responders.
- **Engagement and escalation** – Automatically connect the correct people for each unique incident. Engage through different contact methods and escalate through responders ensuring visibility and active participation during incidents.
- **Active collaboration** – Incident responders actively respond to incidents through integration with the AWS Chatbot client.
- **Incident tracking** – Review incident details for up-to-date information during an incident. Create and remediate follow-up items while following runbooks.

Introducing Incident Manager from AWS Systems Manager | Amazon Web Services

Benefits of using Incident Manager

Align quickly

Incident Manager provides the ability to automatically collect and track the metrics related to an incident, through the use of Amazon CloudWatch metrics.

You can add metrics manually to an incident, in real time, by using a *chat channel* or the Incident Manager *incident details*. Investigate metrics further by using the built-in CloudWatch graphs.

Use the Incident Manager incident timeline to display points of interest in chronological order. Responders can also use the timeline to add custom events, describing what they did or what happened. Automated points of interest include:

- Metrics going into alarm

- Added Metrics
- Engaged Responders
- Runbook steps completed

Collaborate effectively

Incident Manager brings incident responders together through the use of contacts, escalation plans, and chat channels. Define *contacts* directly in Incident Manager with their preferred contact channels. Using your defined contacts, create *escalation plans* to engage the necessary responders at the right time during an incident.

Bring together responders in connected chat channels where they can directly interact with the incident using AWS Chatbot clients. Incident Manager displays the real-time actions of incident responders in the chat channel, providing context to others. Communication during an incident is the key to faster resolution.

Automate and improve

Incident Manager enables your responders to focus on the key tasks required to resolve an incident through the use of *runbooks*. Runbooks, a series of actions taken to resolve an incident, combine the power of automated tasks and the detail of manual steps leaving responders more available to analyze and respond to impact.

Using Incident Manager *post incident analysis*, your team can develop more robust response plans and affect change across your applications to prevent future incidents and down time. Post incident analysis also provides for iterative learning and improvement of runbooks, response plans, and metrics.

Related services

Incident Manager integrates with your current AWS environment to provide rapid resolution of incidents.

- **AWS Chatbot** – DevOps teams can use Amazon Chime and Slack chat rooms to monitor and respond to incidents. To learn how Incident Manager works with AWS Chatbot, see Chat channels (p. 19) in the *Communications* section of this guide.
- **AWS CloudFormation** – Automate the creation of response plans using AWS CloudFormation. For more information, see the AWS CloudFormation User Guide.
- **Amazon CloudWatch** – Configure CloudWatch to monitor your application resources. Use CloudWatch alarms to initiate incidents in the Incident Manager console. To learn about monitoring best practices, see Response plans (p. 24). Review detailed metrics during an incident using the incident details page. For more information about metrics in the incident details page, see Metrics (p. 31) in the incident details section of this guide.
- **AWS Systems Manager** – Use Systems Manager capabilities to view and control your application infrastructure. For more information, see the Systems Manager User Guide.
  - OpsCenter – Create *OpsItems* directly from a post incident analysis to follow up on related work. To learn more about continuous improvement using Incident Manager post incident analysis, see the Post-incident analysis (p. 34) section of this guide.
  - Automation – To learn about creating runbooks using Systems Manager automation, see the Runbooks and automation (p. 22) section of this guide.

Accessing Incident Manager

You can access Incident Manager in any of the following ways:

- **AWS Systems Manager Incident Manager console** – Incident Manager console
- **AWS CLI** – For more information, see Getting set up with the AWS Command Line Interface in the *AWS Command Line Interface User Guide*.
- **Incident Manager API** – For more information, see the AWS Systems Manager Incident Manager API Reference
- **AWS SDKs** – For more information, see Tools for Amazon Web Services

Incident Manager regions and quotas

Incident Manager isn't supported in all Systems Manager regions.

To view information about Incident Manager regions and quotas, see Incident Manager endpoints and quotas in the Amazon Web Services General Reference guide.

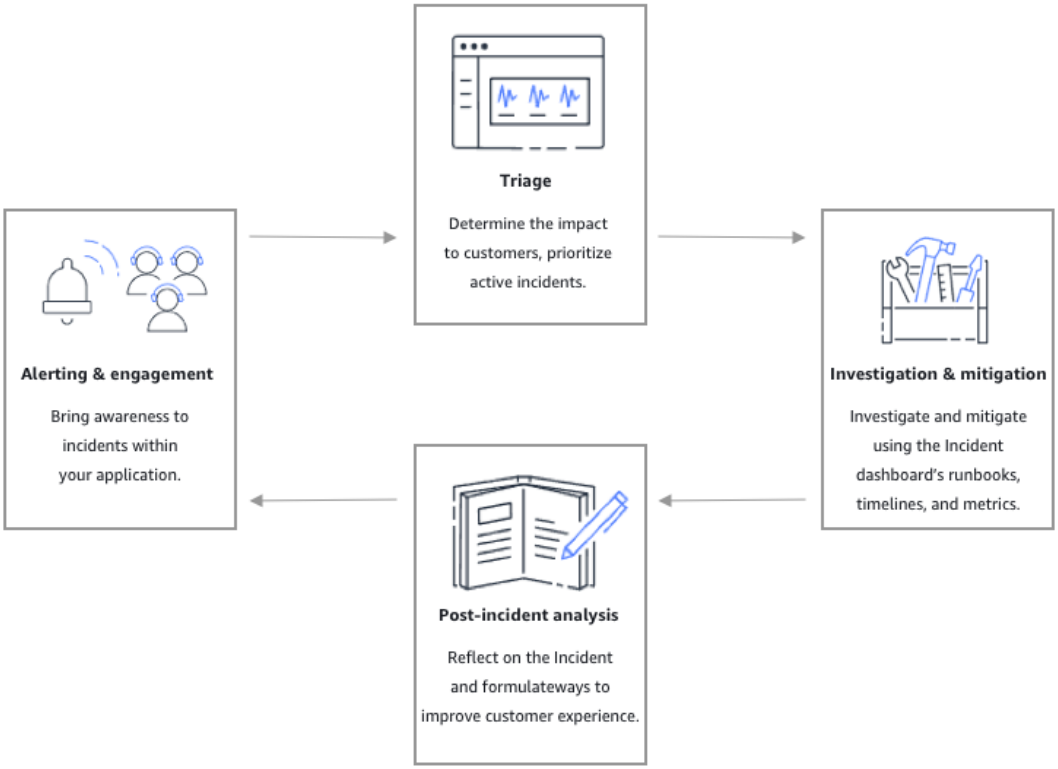
Pricing for Incident Manager

There is a charge to use Incident Manager. For more information, see AWS Systems Manager pricing.

Other AWS services, AWS content and third-party content made available in connection with this Service may be subject to separate charges and governed by additional terms.

Incident lifecycle

AWS Systems Manager Incident Manager is an incident lifecycle management tool. The primary goal of Incident Manager is to facilitate the return of your AWS-hosted applications to normal as quickly as possible. Incident Manager provides tools and best practices for every phase of the incident lifecycle:



Alerting and engagement

The alerting and engagement phase of the incident lifecycle focuses on bringing awareness to incidents within your applications. This phase begins before an incident is ever detected and requires a deep understanding of your applications. You can use Amazon CloudWatch metrics to monitor data about the performance of your applications. For more information, see Using Metrics in the CloudWatch user guide. After you've set up monitoring for your applications, you can begin alerting on metrics that stray outside the historical norm by using CloudWatch alarms. To learn more about monitoring best practices, see Monitoring (p. 16) in the incident preparedness section of this user guide.

Now that you are monitoring for incidents in your applications, you can define an incident *response plan* to use during an incident. To learn more about creating response plans, see Response plans (p. 24). Amazon EventBridge events or CloudWatch Alarms can automatically create an incident using with response plans as the template. To learn more about incident creation, see Incident creation (p. 27).

Response plans launch related *escalation plans* and *engagement plans* to bring first responders into the incident. For more information about setting up escalation plans, see Create an escalation plan (p. 19). Simultaneously, AWS Chatbot notifies responders using a *chat channel* directing them to the incident detail page. Using the chat channel and *incident details*, the team can communicate and triage an incident. For more information about setting up chat channels in Incident Manager, see Set up an AWS Chatbot client (p. 19).

Triage

Triage is when first responders attempt to determine the impact to customers. Responders prioritize incidents by using the following impact rating:

- 1 – Critical impact, this typically relates to full application failure that impacts many to all customers
- 2 – High impact, partial application failure with impact to many customers
- 3 – Medium impact, the application is providing reduced service to customers
- 4 – Low impact, customer might aren't impacted by the problem yet
- 5 – No impact, customers aren't currently impacted but urgent action is needed to avoid impact

Investigation and mitigation

The *incident* details view provides your team with runbooks, timelines, and metrics. To see how you can work with an incident, see the Incident details (p. 30).

Runbooks commonly provide investigation steps and can automatically pull data or attempt commonly used solutions. Runbooks also provide clear, repeatable steps that your team has found to be useful in mitigating incidents. The runbook tab focuses on the current runbook step and shows past and future steps.

Incident Manager integrates with Systems Manager Automation to build runbooks. Use runbooks to do any of the following:

- Manage instances and AWS resources
- Automatically run scripts
- Manage AWS CloudFormation resources

For more information about the supported action types, see Systems Manager Automation actions reference.

The *timeline tab* shows what actions have been taken. The timeline records each with a timestamp and automatically created details. To add custom events to the timeline, see the Timeline (p. 31) section in the *Incident details* page of this user guide.

The *metrics tab* shows automatically populated metrics and manually added metrics. This view provides valuable information into the activities of your application during an incident.

Using *chat channels*, through AWS Chatbot, you can directly interact with your incident. Using AWS Chatbot you can use any Incident Manager API action in the configured chat channel. Update the title and description while you resolve the incident directly from the chat channel. For more information about the commands available, see the AWS Systems Manager Incident Manager API Reference.

Post-incident analysis

Use *post incident analysis* to reflect on the incident. Post incident analysis provides the structure in which your team can formulate ways to improve response and customer experience. Improvements can include:

- Changes to the applications involved in an incident. Your team can use this time to improve the system and make it more fault tolerant.
- Changes to an incident response plan. Take the time to incorporate learned lessons.
- Changes to runbooks. Your team can dive deep into steps needed for resolution and the steps that you can automate.
- Changes to alerting. After an incident, your team might have noticed critical points in the metrics you can use to alert the team sooner about an incident.

Incident Manager facilitates these potential improvements by using a set of post incident analysis questions and action items alongside the incident timeline. To learn more about improvement through analysis, see Post-incident analysis (p. 34).

Getting prepared with Incident Manager

This section walks through **Get prepared** in the Incident Manager console. You're required to complete **Get prepared** in the console before you can begin to use it for incident management. The wizard walks you through setting up your replication set, at least one contact, at least one escalation plan, and your first response plan. The following is useful background information to help you understand Incident Manager and the incident lifecycle:

- What Is AWS Systems Manager Incident Manager? (p. 1)
- Incident lifecycle (p. 3)

Prerequisites

If you're using AWS for the first time or you're setting up your AWS Identity and Access Management account, see the AWS Identity and Access Management prerequisites (p. 8).

We recommend you complete the Systems Manager quick setup before beginning the Incident Manager **Get prepared** wizard. Use Systems Manager Quick Setup to configure frequently used AWS services and features with recommended best practices. Incident Manager uses Systems Manager features to manage incidents in your system and benefits from having Systems Manager configured first.

Before you begin your account must have the IAM permission `iam:CreateServiceLinkedRole`. Incident Manager uses this permission to create the `AWSServiceRoleforIncidentManager` in your account. To learn more about this service linked role, see Using service-linked roles for Incident Manager (p. 58).

We recommend setting up Incident Manager in the account that you use to manage your operations.

Get prepared wizard

The first time you use Incident Manager you can access the **Get prepared** wizard from the Incident Manager service homepage. To access the **Get prepared** wizard after first setup, choose **Prepare** on the **Incidents** list page.

1. Open the Incident Manager console.
2. On the Incident Manager service homepage, choose **Get prepared**.

General settings

1. Choose **General settings**. Use the **General settings** to set up your replication set. The replication set ensures that your response plans, contacts, and escalation plans are usable from any of the Regions that you develop or maintain applications in.

2. Read the on-boarding acknowledgment. If you agree to the Incident Manager terms and conditions, choose **I have read and agree to the AWS Systems Manager Incident Manager terms and conditions** and choose **Next**.
3. Set up the replication set using either an AWS owned key or your own AWS KMS key. All Incident Manager resources are encrypted To learn more about how your data is encrypted, see Data Protection in Incident Manager (p. 41).
   - If you want to use the AWS owned key, choose **Use AWS owned key** and choose **Create**.
   - If you want to use your own AWS KMS key, choose **Choose a different AWS KMS key (advanced)**.
     a. Your current Region appears as the first Region in your replication set. Search for an AWS key that you already have in our account or choose **Create an AWS KMS key**.
     b. To add more Regions to your replication set, choose **Add Region**.
4. To create your replication set and begin creating contacts, choose **Create**.

   Note
   Creating the replication set creates the `AWSServiceRoleforIncidentManager` service-linked role in your account. To learn more about this role, see Using service-linked roles for Incident Manager (p. 58)

Contacts (optional)

1. Choose **Create contact**. Incident Manager engages contacts during an incident. For more information about contacts, see Contacts (p. 16).
2. Provide the contact name.
3. Provide a unique and identifiable alias.
4. Create **Contact channels**.
   a. Choose the **Type**. Incident Manager supports **Email**, **SMS**, or **Voice**.
   b. Provide a unique and identifiable name.
   c. Provide the channel details, such as an email address for **Email**.
   d. To create another contact channel, choose **Add a new contact channel**.
5. Create the contacts **Engagement plan**. We recommend defining two or more devices in the engagement plan with at least one device that's engaged at the beginning of the engagement.
   a. Choose a **Contact channel name**.
   b. Define the number of minutes to wait until engaging the contact channel.
   c. To add more contact channels to the engagement plan, choose **Add engagement**.
6. To create the contact and send activation codes to the defined contact channels, choose **Next**.
7. (Optional) Enter the activation code sent to each device.
8. Repeat step four until you have added all of your contacts to Incident Manager.
9. Choose **Finish**.

Escalation plans (optional)

1. Choose **Create escalation plan**. An escalation plan escalates through your contacts during an incident, ensuring that Incident Manager engages the correct responders during an incident. For more information about escalation plans, see Escalation plans (p. 18).
2. Provide a unique and identifiable **Escalation plan name**.
3. Specify the number of minutes the first stage should last before starting the next stage.

4. Add contacts to the first stage by choosing them from the **Contact** search bar.
5. If you want the contact to be able to halt the progression of escalation plan stages, select **Acknowledgment stops plan progression**.
6. To add more contacts to a stage, choose **Add contact**.
7. To create a new stage in the escalation plan, choose **Add stage**.
8. After you've finished adding stages and contacts, choose **Create escalation plan**.

Response plan

1. Choose **Create response plan**. Use the response plan to put together contacts and escalation plans you created. During this **Getting started** wizard, skip the **Chat channel** and **Runbooks** sections of the response plan. To learn more about creating response plans with contacts, escalation plans, chat channels, and runbooks, see Incident preparation (p. 15).
2. Provide a unique, identifiable **Name**.
3. Provide an **Incident title prefix**.
4. Choose the expected **Impact** of the incident.
5. Provide a brief **Summary** of the incident to inform responders of what's happening. Incident Manager automatically populates relevant information into the summary during an incident.
6. (Optional) Provide a dedupe string. The dedupe string removes duplicate incidents created in the same account.
7. Choose contacts and escalation plans from the **Engagements** dropdown.
8. Choose **Create response plan**. After you choose **Create response plan**, the **Response plans** list console page opens.

After you've created a response plan, you can associate Amazon CloudWatch alarms or Amazon EventBridge events with the response plan to automatically create an incident based on an alarm or event. To learn more about incident creation, see Incident creation (p. 27).

AWS Identity and Access Management prerequisites

Get an AWS account and your root user credentials

To access AWS, you must sign up for an AWS account.

To sign up for an AWS account

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.

AWS sends you a confirmation email after the sign-up process is complete. At any time, you can view your current account activity and manage your account by going to https://aws.amazon.com/ and choosing **My Account**.

Creating an IAM user

If your account already includes an IAM user with full AWS administrative permissions, you can skip this section.

When you first create an Amazon Web Services (AWS) account, you begin with a single sign-in identity. That identity has complete access to all AWS services and resources in the account. This identity is called the AWS account *root user*. When you sign in, enter the email address and password that you used to create the account.

> **Important**
> 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. To view the tasks that require you to sign in as the root user, see Tasks that require root user credentials.

To create an administrator user for yourself and add the user to an administrators group (console)

1. Sign in to the IAM console as the account owner by choosing **Root user** and entering your AWS account email address. On the next page, enter your password.

   > **Note**
   > We strongly recommend that you adhere to the best practice of using the **`Administrator`** IAM user that follows and securely lock away the root user credentials. Sign in as the root user only to perform a few account and service management tasks.

2. In the navigation pane, choose **Users** and then choose **Add user**.
3. For **User name**, enter **`Administrator`**.
4. Select the check box next to **AWS Management Console access**. Then select **Custom password**, and then enter your new password in the text box.
5. (Optional) By default, AWS requires the new user to create a new password when first signing in. You can clear the check box next to **User must create a new password at next sign-in** to allow the new user to reset their password after they sign in.
6. Choose **Next: Permissions**.
7. Under **Set permissions**, choose **Add user to group**.
8. Choose **Create group**.
9. In the **Create group** dialog box, for **Group name** enter **`Administrators`**.
10. Choose **Filter policies**, and then select **AWS managed - job function** to filter the table contents.
11. In the policy list, select the check box for **AdministratorAccess**. Then choose **Create group**.

    > **Note**
    > You must activate IAM user and role access to Billing before you can use the `AdministratorAccess` permissions to access the AWS Billing and Cost Management console. To do this, follow the instructions in step 1 of the tutorial about delegating access to the billing console.

12. Back in the list of groups, select the check box for your new group. Choose **Refresh** if necessary to see the group in the list.
13. Choose **Next: Tags**.
14. (Optional) Add metadata to the user by attaching tags as key-value pairs. For more information about using tags in IAM, see Tagging IAM entities in the *IAM User Guide*.
15. Choose **Next: Review** to see the list of group memberships to be added to the new user. When you are ready to proceed, choose **Create user**.

You can use this same process to create more groups and users and to give your users access to your AWS account resources. To learn about using policies that restrict user permissions to specific AWS resources, see Access management and Example policies.

Signing in as an IAM user

Sign in to the IAM console by choosing **IAM user** and entering your AWS account ID or account alias. On the next page, enter your IAM user name and your password.

> **Note**
> For your convenience, the AWS sign-in page uses a browser cookie to remember your IAM user name and account information. If you previously signed in as a different user, choose the sign-in link beneath the button to return to the main sign-in page. From there, you can enter your AWS account ID or account alias to be redirected to the IAM user sign-in page for your account.

Creating IAM user access keys

Access keys consist of an access key ID and secret access key, which are used to sign programmatic requests that you make to AWS. If you don't have access keys, you can create them from the AWS Management Console. As a best practice, do not use the AWS account root user access keys for any task where it's not required. Instead, create a new administrator IAM user with access keys for yourself.

The only time that you can view or download the secret access key is when you create the keys. You cannot recover them later. However, you can create new access keys at any time. You must also have permissions to perform the required IAM actions. For more information, see Permissions required to access IAM resources in the *IAM User Guide*.

To create access keys for an IAM user

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Users**.
3. Choose the name of the user whose access keys you want to create, and then choose the **Security credentials** tab.
4. In the **Access keys** section, choose **Create access key**.
5. To view the new access key pair, choose **Show**. You will not have access to the secret access key again after this dialog box closes. Your credentials will look something like this:
   - Access key ID: AKIAIOSFODNN7EXAMPLE
   - Secret access key: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
6. To download the key pair, choose **Download .csv file**. Store the keys in a secure location. You will not have access to the secret access key again after this dialog box closes.

   Keep the keys confidential in order to protect your AWS account and never email them. Do not share them outside your organization, even if an inquiry appears to come from AWS or Amazon.com. No one who legitimately represents Amazon will ever ask you for your secret key.
7. After you download the `.csv` file, choose **Close**. When you create an access key, the key pair is active by default, and you can use the pair right away.

Related topics

- What is IAM? in the *IAM User Guide*
- AWS security credentials in *AWS General Reference*

Setting up cross-account functionality

AWS Systems Manager Incident Manager uses AWS Resource Access Manager (AWS RAM) to share Incident Manager resources across management and application accounts. This section describes cross-account best practices, how to set up cross-account functionality for Incident Manager, and some limitations of cross-account functionality in Incident Manager.

A management account is the account that you perform operations management from. The management account owns the response plans, contacts, escalation plans, runbooks, and other AWS Systems Manager resources.

An application account is the account that owns the resources that make up your applications. These resources can be Amazon EC2 instances, Amazon DynamoDB tables, or any of the other resources that you use to build applications in the AWS Cloud. Application accounts also own the Amazon CloudWatch alarms and Amazon EventBridge events that create incidents in Incident Manager.

AWS RAM uses resource shares to share resources between accounts. You can share the response plan and contact resources between accounts in AWS RAM. By sharing these resources, application accounts and management accounts can interact with engagements and incidents. Sharing a response plan shares all past and future incidents created using the response plan. Sharing a contact shares all past and future engagements of the contact or response plan.

Best practices

Follow these best practices when sharing your Incident Manager resources across accounts.

- Regularly update the resource share with response plans and contacts.
- Regularly review resource share principals.
- Set up Incident Manager, runbooks, and chat channels in your management account.

Set up and configuration

The following steps describe how to set up and configure Incident Manager resources and use them for cross-account functionality. You may have configured some services and resources for cross-account functionality in the past. Use these steps as a checklist of requirements before starting your first incident using cross-account resources.

1. (Optional) Create organizations and organizational units using AWS Organizations. Follow the steps in the Organizations Tutorial: Creating and configuring an organization.
2. (Optional) Use the Systems Manager quick setup to set up the correct AWS Identity and Access Management roles for you to use when configuring your cross-account runbooks. For more information, see Quick Setup in the Amazon EC2 Systems Manager user guide.
3. Create runbooks in Systems Manager automation documents using the recommended Systems Manager multiple account and Region setup. Follow the steps listed in that guide and then see the following steps for further configuration for Incident Manager runbooks. The following use cases require installing AWS CloudFormation templates for the necessary roles that can create and view runbooks during an incident. Use these policy templates during the setup described in the Systems Manager documentation.
    - *Running a runbook in the management account.* The management account must download and install this CloudFormation template: `AWS-SystemsManager-AutomationReadOnlyRole`. When installing `AWS-SystemsManager-AutomationReadOnlyRole`, specify the account IDs of all application accounts. This role will let your application accounts read the status of the runbook from the incident details page. The application account must install this CloudFormation template: `AWS-SystemsManager-`

`AutomationAdministrationReadOnlyRole`. The incident details page uses this role to get the automation status from the management account.

- *Running a runbook in an application account.* The management account must download and install the following CloudFormation template: `AWS-SystemsManager-AutomationAdministrationReadOnlyRole`. This role allows the management account to read the status of the runbook in the application account. The application account must download and install the following CloudFormation template: `AWS-SystemsManager-AutomationReadOnlyRole`. When installing `AWS-SystemsManager-AutomationReadOnlyRole`, specify the account ID of the management account and other application accounts. The management and other application accounts assume this role to read the status of the runbook.

4. To set up and create contacts, escalation plans, chat channels, and response plans, follow the steps detailed in Incident preparation (p. 15).
5. Add your contacts and response plan resources to your existing resource share or a new resource share in AWS RAM. For more information, see Getting started with AWS RAM. Adding response plans to AWS RAM enables application accounts to access incidents and incident dashboards created using the response plans. Application accounts also gain the ability to associate CloudWatch alarms and EventBridge events to a response plan. Adding the contacts and escalation plans to AWS RAM enables application accounts to view engagements and engage contacts from the incident dashboard.
6. Add cross-account cross-Region functionality to your CloudWatch console. For steps and information, see Cross-Account Cross-Region CloudWatch Console in the CloudWatch user guide. Adding this functionality ensures the application accounts and management account you've created can view and edit metrics from the incident and analysis dashboards.
7. Create a cross-account Amazon EventBridge event bus. For steps and information, see Running automations in multiple AWS Regions and accounts. You can then use this event bus to create event rules that detect incidents in application accounts and create incidents in the management account.

Limitations

The following are known limitations of the Incident Manager cross-account functionality:

- Timeline events aren't populated for automation documents run in application accounts. Updates of automation documents run in application accounts are visible in the runbook tab of the incident.

- SNS topics can't be used cross-account. SNS topics must be created in the same Region and account as the response plan it's used it. We recommend using the management account to create all SNS topics and response plans.

Using tagging with Incident Manager

Incident Manager supports tagging response plans in a single Region in your management account and tagging contacts and escalation plans in any region in your management account. Incident Manager sets up the first Region in your replication set as your tagging Region. Tags must be added to a response from the defined tagging Region. Use tagging to allocate costs to the correct teams based on resource usage in Incident Manager. Incident Manager supports tagging on the following resources:

- Response plans
- Contacts

To tag a resource while you're creating the resource, on the create resource page, search for the tag **Key** or **Value** and choose **Add new tag**.

To add or remove tags from a resource after it's created, in the **Tags** section of the edit resource page, search for the tags **Key** or **Value**. After searching for the tag, choose **Add new tag**. To remove a tag, choose **Remove**. After you have added or removed tags, choose **Update**.

Using the Incident Manager replication set

The Incident Manager replication set replicates your data to many Regions to increase cross Region redundancy, allow Incident Manager to access resources in different Regions and reduce latency for your users. The replication set it also used to encrypt your data with either an AWS owned key or your own customer owned key. All Incident Manager resources are encrypted by default. To learn more about how your resources are encrypted, see Data Protection in Incident Manager (p. 41). To get started with Incident Manager, first create your replication set using the **Get prepared** wizard. To learn more about getting prepared in Incident Manager, see the Get prepared wizard (p. 6).

Editing your replication set

Using the Incident Manager **Settings** page you can edit your replication set. You can add Regions, delete Regions, and enable or disable replication set deletion protection. You can't edit the key used to encrypt you data. To change the key, delete the replication set.

Add a Region

1. Navigate to the Incident Manager console and choose **Settings** from the left navigation bar.
2. Choose **Add Region**.
3. Select the **Region**.
4. Choose **Add**.

Delete a Region

1. Navigate to the Incident Manager console and choose **Settings** from the left navigation bar.
2. Select the Region you want to delete.
3. Choose **Delete**.
4. Enter **delete** into the text box and choose **Delete**.

Deleting your replication set

Deleting the last Region in your replication set deletes the entire replication set. Before you can delete the last Region, disable the deletion protection by toggling **Deletion protection** on the **Settings** page. After you've deleted your replication set you can create a new replication set by using the **Get prepared** wizard.

To delete Regions from your replication set, wait 24 hours after creating them. Attempting to delete a Region from your replication set sooner than 24 hours after creation causes the deletion to fail.

Deleting your replication set deletes all Incident Manager data.

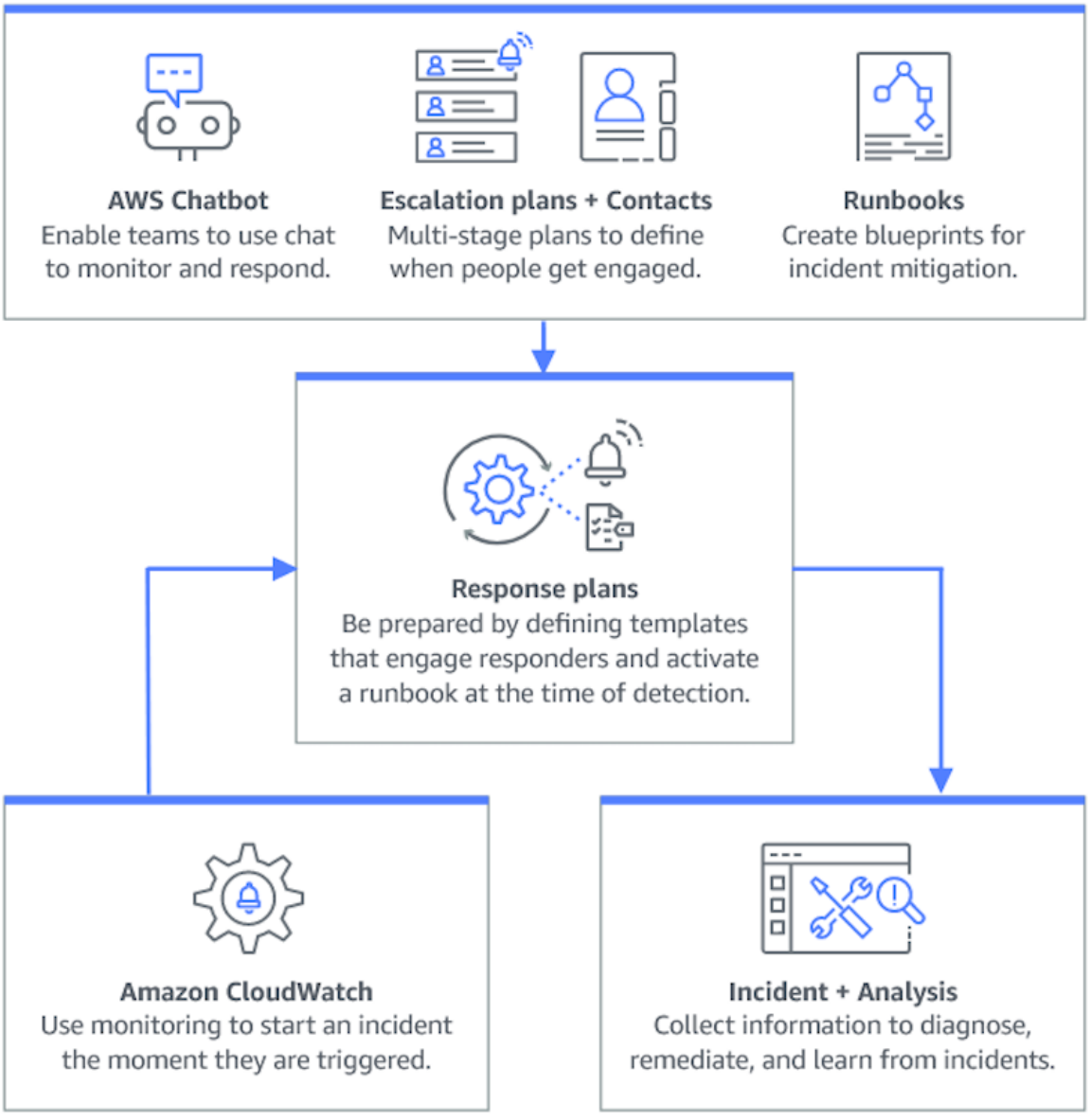
Delete the replication set

1. Navigate to the Incident Manager console and choose **Settings** from the left navigation bar.
2. Select the last Region in your replication set.
3. Choose **Delete**.

4. Enter **delete** into the text box and choose **Delete**.

Incident preparation

Planning for an incident begins long before the incident lifecycle. To prepare for an incident, consider each of the following topics before you create response plans. Use monitoring, contacts, escalation plans, chat channels, and runbooks to build response plans that automate response.



Topics

Monitoring

Monitoring the health of your AWS hosted applications is key to ensuring application up time and performance. When determining monitoring solutions, consider the following:

- **Criticality of feature** – If the system were to fail, how critical would the impact to downstream users be.
- **Commonality of failure** – How commonly does a system fail; systems that require frequent intervention should be closely monitored.
- **Increased latency** – How much the time to complete a task has increased or decreased.
- **Client-side versus server-side metrics** – If there is a discrepancy between related metrics on the client and server.
- **Dependency failures** – Failures that your team can and should prepare for.

After creating response plans, you can use your monitoring solutions to automatically track incidents the moment they happen in your environment. For more information about incident tracking and creation, see the Incident tracking (p. 30).

For more information about architecting secure, high-performing, resilient, and efficient infrastructure applications and workloads, see the AWS Well-Architected whitepaper.

Contacts

AWS Systems Manager Incident Manager contacts are responders to incidents. A contact can have multiple channels that Incident Manager can engage during an incident. You can define a contact's engagement plan to describe how and when Incident Manager engages the contact.

Topics

Contact channels

Contact channels are the various methods Incident Manager uses to engage a contact.

Incident Manager supports the following contact channels:

- Email
- SMS
- Voice

Contact channel activation

To protect your privacy and security, Incident Manager sends a device activation code to you when you create contacts. To engage your devices during an incident, you must first activate them. To do so, enter the device activation code on the create contact page.

Certain features of Incident Manager include functionality that send notifications to a contact channel. By using these features, you instruct us, as part of the functioning of these features, to send notifications (SMS/voice messages), of service disruptions or other events, to the contact channels entered into the applicable workflows. You confirm, by using these features, that you're authorized to input the contact channels provided to Incident Manager.

Opting out

You can cancel these notifications at any time by removing a mobile device as a contact channel. Individual notification recipients may also cancel notifications at any time by removing the device from their contact.

To remove a contact channel from a contact

1. Navigate to the Incident Manager console and choose **Contacts** from the left navigation.
2. Select the contact with the contact channel that you are removing and choose **Edit**.
3. Choose **Remove** next to the contact channel that you would like to remove.
4. Choose **Update**.

Contact channel deactivation

To deactivate a device, reply **UNSUBSCRIBE**. Replying **UNSUBSCRIBE** stops Incident Manager from engaging your device.

Contact channel reactivation

1. Reply **START** to the message from Incident Manager.
2. Navigate to the Incident Manager console and choose **Contacts** from the left navigation.
3. Select the contact with the contact channel that you are removing and choose **Edit**.
4. Choose **Activate devices**.
5. Enter the **Activation code** sent to the device by Incident Manager.
6. Choose **Activate**.

Engagement plans

Engagement plans define when Incident Manager engages the contact channels. You can engage contact channels multiple times at different stages from the start of an engagement. You can use engagement plans in an escalation plan or response plan. To learn more about escalation plans, see Escalation plans (p. 18).

Define a contact

To define a contact, use the following steps.

1. Open the Incident Manager console and choose **Contacts** from the left navigation.
2. Choose **Create Contact**.
3. Type the full name of the contact and provide a unique and identifiable alias.

4. Define a **Contact channel**. We recommend having two or more different types of contact channels.
   a. Choose the type: email, SMS, or voice.
   b. Enter an identifiable name for the contact channel.
   c. Provide the contact channel details, such as email: arosalez@example.com
5. To define more than one contact channel, choose **Add contact channel**. Repeat step 4 for each new contact channel added.
6. Define an engagement plan.

   Important
   To engage a contact you must define an engagement plan.

   a. Choose a **Contact channel name**.
   b. Define how many minutes from the start of the engagement to wait until Incident Manager engages this contact channel.
   c. To add another contact channel, choose **Add engagement**.
7. After defining your engagement plan, choose **Create**. Incident Manager sends an activation code to each of the defined contact channels.
8. (Optional) To activate the contact channels, enter the activation code that Incident Manager sent to each defined contact channel.
9. (Optional) To send a new activation code, choose **Send new code**.
10. Choose **Finish**.

After you define a contact and activate its contact channels, you can add contacts to escalation plans to form a chain of escalation. To learn more about escalation plans, see Escalation plans (p. 18). You can add contacts to a response plan for direct engagement. To learn more about creating response plans, see Response plans (p. 24).

Escalation plans

AWS Systems Manager Incident Manager provides escalation paths through your defined contacts. You can pull multiple contacts into an incident at the same time. If these contacts don't respond, Incident Manager escalates to the next set of contacts. You can also choose if a plan will stop escalating once a user acknowledges the engagement. You can add escalation plans to a response plan so escalation automatically starts at the beginning of an incident. You can also add escalation plans to an active incident.

Topics

Stages

Escalation plans use stages where each stage lasts a defined number of minutes. Each stage has the following information:

- **Duration** - The amount of time the plan waits until beginning the next stage. The first stage of the escalation plan begins once the engagement starts.
- **Contacts** - The escalation plan engages each contact using its defined engagement plan. You can set up each contact to stop the progression of the escalation plan before it goes to the next stage. Each

stage can have multiple contacts. To learn more about setting up contacts, see the Contacts (p. 16) section of this guide.

Create an escalation plan

1. Open the Incident Manager console and choose **Escalation plans** from the left navigation.
2. Choose **Create escalation plan**.
3. Provide a unique name for the escalation plan.
4. Define the number of minutes until the next stage begins.
5. Use the search bar to find and add a contact.
6. (Optional) To have a contact stop the escalation plan when they acknowledge the engagement, select **Contact acknowledgment stops plan progression**.
7. To add another contact to this stage, choose **Add contact**.
8. To add a new stage, choose **Add stage**.
9. Repeat steps 4 through 8 until you have added all contacts and stages you want.
10. Choose **Create escalation plan**.

Chat channels

A key feature of AWS Systems Manager Incident Manager is the ability to directly communicate through chat channels during an incident. During an incident, Incident Manager pushes incident updates and notifications directly to the chat channel to keep all responders informed. Responders can update and interact with the incident directly from the chat channel by using chat commands. For more information about AWS Chatbot, see AWS Chatbot Administrator Guide.

Topics

Set up an AWS Chatbot client

Incident Manager uses AWS Chatbot clients to connect responders in Amazon Chime or Slack. Add AWS Chatbot client to a response plan to notify the chat room that an incident started.

Configure an AWS Chatbot client.

1. Open the AWS Chatbot console, and in the left navigation bar, choose **Configured clients**.
2. Choose **Configure new client**.
3. Choose **Amazon Chime** or **Slack**.

Slack

1. Choose your workspace from the dropdown list on the top right.
   - If you're not already signed in to a workspace, choose **Sign in to a workspace**.

2. To grant AWS Chatbot permission to access your Slack workspace, choose **Allow**.
3. Choose **Configure new channel**.
4. Switch to Slack and add the AWS Chatbot app.
   a. In the Slack navigation bar, choose **More** and then choose **Apps**.
   b. Search for and choose **AWS Chatbot**. **aws** now appears in your **Apps** list in the navigation bar.
   c. Invite AWS Chatbot to your channel: `/invite @AWS`
5. Switch to the **Configure Slack channel** page and enter an identifiable configuration name.
6. *Optional*– If you would like logging on this channel, Select **Logging**.
7. For the channel type, choose **public** or **private**. Channels might take some time to populate because AWS Chatbot fetches all channels available in the workspace.
   - For public channels, use the search bar to choose your public channel.
   - For private channels, navigate to your channel within Slack and right-click the channel name. Choose **Copy link**, then enter the link in the **Private Channel ID** field on the AWS Chatbot configuration page.
8. Choose **Create an IAM role using a template**, and enter a role name.
9. For **policy templates**, choose **AWS Systems Manager Incident Manager permissions**.
10. In the **Notifications** section, choose the Region for your first SNS topic.
11. Choose the SNS topics you would like to notify during an incident. To learn more about SNS topics, see Amazon SNS.

    Note
    Incident Manager requires SNS topics to send notifications to your chat channels.

Amazon Chime

Note
Chat commands aren't supported on Amazon Chime.

1. Enter an identifiable **Configuration name**.
2. Open the Amazon Chime desktop client, then open the chat room you want.
3. Choose the gear icon in the upper-right corner, and then choose **Manage webhooks and bots**.
4. In the **Manage incoming webhooks and bots** dialog box, choose **Add webhook**, type a name for the webhook, and then choose **Create**.
5. Verify the webhook you created is listed, and choose **Copy URL** to copy the webhook URL to your clipboard.
6. On the **Configure Amazon Chime webhook** page, paste this copied webhook into the **Webhook URL** field.
7. Provide a brief description to identify the chat room and purpose.
8. You can optionally turn on logging for this chat room.
9. Choose **Create an IAM role using a template.**
10. Enter a **Role name**.
11. For **Policy templates**, choose **AWS Systems Manager Incident Manager permissions**.
12. In the **Notifications** section, choose the Region for your first SNS topic.
13. Choose any number of SNS topics you would like to notify during an incident. To learn more about SNS topics, see Amazon SNS.

    Note
    Incident Manager requires SNS topics to send notifications to your chat channels.

You can now add this AWS Chatbot client to an Incident Manager response plan. To learn more about setting up response plans, see Response plans (p. 24).

Using SNS topics with Incident Manager incurs the costs of SNS. For more information, see Amazon SNS pricing.

Configuring SNS permissions

Before you can use the AWS Chatbot client during an incident, update the access policy of the related Amazon SNS topics.

1. Navigate to the Amazon SNS console and choose **Topics** from the navigation panel.
2. Select the Amazon SNS topic related to the AWS Chatbot client you set up in the previous section.
3. Choose **Edit**.
4. Expand the **Access policy** section and add the following statement to the policy's **Statement** array.

```
{
 "Sid": "IncidentManagerSNSPublishingPermissions",
 "Effect": "Allow",
 "Principal": {
   "Service": "ssm-incidents.amazonaws.com"
 },
 "Action": "SNS:Publish",
 "Resource": "arn:aws:sns:us-east-1:111122223333:example_SNS_topic",
 "Condition": {
   "StringEqualsIfExists": {
     "AWS:SourceAccount": "111122223333"
    }
  }
}
```

Important
The AWS service ssm-incidents.amazonaws.com must have permissions to publish to the chat channel's SNS topic. Without permissions to publish to the SNS topic, Incident Manager won't be able to publish notifications to your chat channel.

5. Replace the `Resource` value *`"arn:aws:sns:us-east-1:111122223333:example_SNS_topic"`* with your Amazon SNS topic ARN.
6. Replace the `AWS:SourceAccount` value *`"111122223333"`* with your AWS account ID.
7. Choose **Save changes**.

Use the previous steps to update each SNS topic related to the configure AWS Chatbot client.

Interacting through chat

Incident Manager enables responders to interact with incidents directly from the chat channel. Chat commands are only available in Slack chat channel. These are some common commands:

- CreateTimelineEvent
- DeleteTimelineEvent
- GetIncidentRecord
- GetTimelineEvent
- ListIncidentRecords
- ListTimelineEvents

- UpdateIncidentRecord
- UpdateTimelineEvent

To use any of the preceding commands from an active incident's chat channel, use the following format. Replace *<CLI Command>* with any of the preceding commands and its appropriate fields.

```
@aws ssm-incidents <CLI Command>
```

```
@aws ssm-contacts <CLI Command>
```

Best practices

Best practices to keep in mind when configuring your chat channels using AWS Chatbot.

- AWS Chatbot enabled Slack channels inherit the permissions of the IAM role used to configure AWS Chatbot. This enables responders in an AWS Chatbot enabled Slack channel to call any allow-listed action; such as Incident Manager APIs and retrieving metrics graphs.
- Maintain principal of least permission, practice security standards, and regularly review membership of your AWS Chatbot enabled chat channels.

Runbooks and automation

A runbook drives incident mitigation and response. AWS Systems Manager Incident Manager brings your runbooks to a central place, ensuring responders focus on mitigation instead of tracking down the next steps. Setup and configure runbooks using AWS Systems Manager runbooks and connect them to an incident by defining them in a response plan. For more information about Automation runbooks, see AWS Systems Manager Automation in the Systems Manager user guide. Using automation steps in runbooks incurs costs in Systems Manager. For more information about Systems Manager billing, Systems Manager pricing. For more information about adding a runbook to a response plan, see Response plans (p. 24).

Define a runbook

When creating a runbook, you can follow the steps provided here or you can follow the more detailed guide provided in the Working with runbooks section in the Systems Manager user guide. If you're creating a multi-account, multi-region runbook, see Running automations in multiple AWS Regions and accounts in the Systems Manager user guide.

Define a runbook

1. Open the Systems Manager console at https://console.aws.amazon.com/systems-manager/.
2. In the navigation pane, choose **Documents**.
3. Choose **Create automation**.
4. Provide a unique and identifiable runbook name.
5. Provide a description of the runbook.
6. Provide an IAM role for the automation document to assume. This allows the runbook to run commands automatically. For more information, see Configuring a service role access for Automation workflows.
7. (Optional) Add the input parameters that the runbook starts with.

8. (Optional) Add a **Target** type.
9. (Optional) Add tags.
10. Fill in the steps that the runbook takes. Each step requires:
    - A name.
    - A description of the purpose of the step
    - The action to run during the step. Runbooks use the **Pause** action type to describe a manual step.
    - (Optional) Provide command properties.
11. After adding all required runbook steps, choose **Create Automation**.

To enable cross-account functionality, share the runbook in your management account with all application accounts that use the runbook during an incident.

Share a runbook

1. Open the Systems Manager console at https://console.aws.amazon.com/systems-manager/.
2. In the navigation pane, choose **Documents**.
3. In the documents list, choose the document you want to share and then choose **View details**. On the **Permissions** tab, verify that you're the document owner. Only a document owner can share a document.
4. Choose **Edit**.
5. To share the command publicly, choose **Public** and then choose **Save**. To share the command privately, choose **Private**, enter the AWS account ID, choose **Add permission**, and then choose **Save**.

Incident Manager runbook template

Incident Manager provides the following runbook templates to get your team started with authoring runbooks in Systems Manager automation. You can use these templates as is or edit them to include more details specific to your application and resources.

Find Incident Manager runbook templates

1. Open the Systems Manager console at https://console.aws.amazon.com/systems-manager/.
2. In the navigation pane, choose **Documents**.
3. In the documents list use the search to find *AWSIncidents-*. This displays all Incident Manager runbooks.

Using a template

1. Open the Systems Manager console at https://console.aws.amazon.com/systems-manager/.
2. In the navigation pane, choose **Documents**.
3. Choose the template you're updating from the documents list.
4. Choose the **Content** tab and copy the content of the document.
5. In the navigation pane, choose **Documents**.
6. Choose **Create automation**.
7. Provide a unique and identifiable name.
8. Choose the **Editor** tab.
9. Choose **Edit**.

10. Enter the copied details in the **Document editor** area.
11. Choose **Create automation**.

AWSIncidents-CriticalIncidentRunbookTemplate

The `AWSIncidents-CriticalIncidentRunbookTemplate` is a template that provides the Incident Manager incident lifecycle in manual steps. These steps are generic enough to use in most applications but detailed enough to get responders started with incident resolution.

Response plans

Use response plans to plan for incidents and define how to respond to incidents. Response plans provide a template for when an incident occurs. This template includes information about who to engage, the expected severity of the event, automatic runbooks to initiate, and metrics to monitor. To create a response plan, use the following steps.

Best practices

Taking the time to plan for incidents ahead of time saves operational time for teams later down the road. Teams should consider the following best practices when designing a response plan.

- **Streamlined engagement** – Identify the most appropriate team for an incident. Engaging wide distribution lists or the wrong teams causes confusion and wastes responder time during incidents.
- **Reliable escalation** – Using escalation plans rather than contacts ensures that responders are effectively and reliably engaged. Even with the best intentions, responders are sometimes unreachable. Having a backup responder configured in an escalation plan covers these scenarios.
- **Runbooks** – Developing runbooks that provide repeatable and understandable steps helps reduce the stress responders experience during incidents.
- **Collaboration** – Use chat channels to streamline communication during incidents. Chat channels help responders stay up to date with information and also share information with other responders.

Response plan creation

Using the response plan best practices and the Incident Manager console, create dynamic response plans to automate incident response.

Response plan details

1. Open the Incident Manager console, and in the left navigation, choose **Response plans**.
2. Choose **Create response plan**.
3. Enter a unique and identifiable response plan **Name**.
4. (Optional) Enter a **Display name**. Use the display name to provide a more user-friendly name to the response plan.

Incident defaults

1. Enter an incident title. The incident title helps to identify an incident on the incidents home page.
2. To indicate the potential scope of the incident, choose an **Impact**.

3. (Optional) Provide a brief description of the incident.
4. (Optional) Provide a dedupe string. Incident Manager uses the dedupe string to prevent the same root cause from creating multiple incidents in the same account.

(Optional) Chat channel

1. Choose a chat channel for the incident responders to interact in during an incident. For more information about chat channels, see Chat channels (p. 19).

   Important
   Incident Manager must have permissions to publish to the chat channel's SNS topic. Without permissions to publish to the SNS topic, you can't add it to the response plan. Incident Manager verifies permissions by publishing a test notification to the SNS topic.
2. (Optional) Choose additional SNS topics to publish to during the incident. Adding SNS topics in multiple Regions increases redundancy in case a Region is down at the time of the incident.

(Optional) Engagements

- For **Engagement**, choose any number of contacts and escalations plans. For information about contact and escalation plan creation, see Contacts (p. 16) and Escalation plans (p. 18).

(Optional) Runbook

1. To select a **Runbook**:
   - Choose **Select an existing runbook**. Select the **Owner**, **Runbook**, and **Version**. For information about runbook creation, see Runbooks and automation (p. 22).
   - Choose **Clone runbook from template**. Enter a descriptive runbook name.
2. Either choose an existing role or use the following steps to create a new role. The role must allow the `ssm:StartAutomationExecution` action for your specific runbook. For the runbook to work across accounts it must also allow the `sts:AssumeRole` action for the `AWS-SystemsManager-AutomationExecutionRole` role that you created during Setting up cross-account functionality (p. 11).

   a. Open the IAM console at https://console.aws.amazon.com/iam/.
   b. Choose **Roles** from the left navigation and choose **Create role**.
   c. Choose **Incident Manager** and choose the **Incident Manager** use case.
   d. Choose **Next: Permissions**.
   e. Choose **Create policy** and then choose the **JSON** tab.
   f. Copy and paste the following JSON blob describing the policy into the JSON editor. Replace the account number (*111122223333*) and runbook name (*DocumentName*) in the runbook's ARN in the following policy example.

```
{
   "Version": "2012-10-17",
   "Statement": [
     {
       "Effect": "Allow",
       "Resource": "arn:aws:ssm:*:111122223333:automation-
definition/DocumentName:*",
       "Action": "ssm:StartAutomationExecution"
     },
     {
       "Effect": "Allow",
       "Resource": "arn:aws:iam::*:role/AWS-SystemsManager-
AutomationExecutionRole",
```

```
      "Action": "sts:AssumeRole"
    }
  ]
}
```

g. Choose **Next: Tags** and (optional) add tags to your policy.
h. Choose **Next: Review**.
i. Provide a **Name** and (optional) provide a **Description** for the policy.
j. Choose **Create policy**.
k. Navigate back to the role you were creating and search for the policy you created. Select the policy.
l. (Optional) Add tags to your role.
m. Provide a **Role name** and (optional) update the **Role description**.
n. Choose **Create role**.

3. Navigate back to the response plan you are creating and refresh the **Role name** dropdown.
4. Select the role you created.
5. Choose the **Execution target**.

Add tags and create the response plan

1. (Optional) Add tags to your response plan.
2. Choose **Create response plan**.

Incident creation

AWS Systems Manager Incident Manager tracks incidents. Using Amazon CloudWatch or Amazon EventBridge, Incident Manager can automatically start incidents. You can also create incidents manually on the incident list page. This section describes automatic and manual incident creation. When you create an incident, Incident Manager creates a parent OpsItem in OpsCenter to track related work and future incident analyses. Calls to OpsCenter incur costs in Systems Manager. For more information about OpsCenter pricing, see Systems Manager pricing.

Automatically create incidents with CloudWatch alarms

CloudWatch uses your CloudWatch metrics to alert you about changes in your environment and to automatically perform the start incident action. CloudWatch works with Systems Manager and Incident Manager to create an incident from a response plan template when an alarm goes into alarm state. This requires the following prerequisites:

- Incident Manager configured and replication set created. This step creates the Incident Manager service linked role in your account, providing the necessary permissions.
- A configured Incident Manager response plan. To learn how to configure Incident Manager response plans, see Response plans (p. 24) in the *Incident preparation* section of this guide.
- Configured CloudWatch metrics monitoring your application. For monitoring best practices, see Monitoring (p. 16) in the *Incident preparation* section of this guide.

To create an alarm with a Start incident action

1. Decide what type of alarm you're creating and follow the steps found in the following sections of the CloudWatch user guide.
   - Create an Alarm Based on a Static Threshold
   - Creating an Alarm Based on Anomaly Detection
   - Creating an Alarm Based on a Metric Math Expression
   - Creating a Composite Alarm
   - Creating a CPU Usage Alarm
   - Creating a Load Balancer Latency Alarm
   - Creating a Storage Throughput Alarm
2. When choosing the action for the alarm to perform, select **Add Systems Manager action**.
3. Choose **Create incident** and select the **Response plan** for this incident.
4. Complete the remaining steps in your selected alarm type guide.

Tip
You can also add the create incident action to any existing alarm.

Automatically create incidents with EventBridge events

EventBridge rules watch for event patterns. If the event matches the defined pattern, Incident Manager creates an incident using the chosen response plan.

Creating incidents using SaaS partners events

You can configure EventBridge to receive events from software as a service (SaaS) partner applications and services, allowing for third-party integration. After configuring EventBridge to receive events from third-party partners, you can create rules that match on partner events to create incidents. To see a list of third-party integrations, see Receiving events from a SaaS partner.

Configure EventBridge to receive events from a SaaS integration.

1. Open the Amazon EventBridge console at https://console.aws.amazon.com/events/.
2. In the navigation pane, choose **Partner event sources**.
3. Use the search bar to find the partner that you want and choose **Set up** for that partner.
4. Choose **Copy** to copy your account ID to the clipboard.

   Note
   To integrate with Salesforce use the steps described in the Amazon AppFlow user guide.
5. Go to the partner's website and follow the instructions to create a partner event source. Use your account ID for this. The event source that you create is available only on your account.
6. Go back to the EventBridge console and choose **Partner event sources** in the navigation pane.
7. Select the button next to the partner event source, and choose **Associate with event bus**.

Create a rule that triggers on events from a SaaS partner

1. Open the Amazon EventBridge console at https://console.aws.amazon.com/events/.
2. In the navigation pane, choose Rules.
3. Choose **Create rule**.
4. Enter a name and description for the rule.
5. For **Define pattern**, choose **Event pattern**.
6. Choose **Pre-defined pattern by service**.
7. For **Service provider**, choose **Service partners**.
8. For **Service name**, choose the name of the partner.
9. For **Event type**, choose **All Events** or choose the type of event to use for this rule. If you choose **All Events**, all events emitted by this partner event source will match the rule.

   If you want to customize the event pattern, choose **Edit**, make your changes, and then choose **Save**.
10. For **Service event bus**, select the event bus that corresponds to this partner.
11. For **Select targets**, choose **Incident Manager response plan**.
12. For **Response plan**, choose a response plan.

    Note
    When selecting a response plan, all response plans that you own and have been shared with your account appear in the **Response plan** dropdown.
13. EventBridge can create the IAM role needed for your rule to run:

    - To create an IAM role automatically, choose **Create a new role for this specific resource**.

- To use an IAM role that you created before, choose **Use existing role**.

14. (Optional) Enter one or more tags for the rule.
15. Choose **Create**.

Creating incidents using AWS service events

EventBridge also receives events from the AWS services listed in Events from Supported AWS Services. Similar to how you configure rules for SaaS partners, you can configure them for AWS services.

Create a rule that triggers on events from an AWS service

1. Open the Amazon EventBridge console at https://console.aws.amazon.com/events/.
2. In the navigation pane, choose Rules.
3. Choose **Create rule**.
4. Enter a name and description for the rule.
5. For **Define pattern**, choose **Event pattern**.
6. Choose **Pre-defined pattern by service**.
7. For **Service provider**, choose **AWS**.
8. For **Service name**, choose the service that monitors for an incident.
9. For **Event type**, choose **All Events** or choose the type of event to use for this rule. If you choose **All Events**, all events emitted by this partner event source will match the rule.

   If you want to customize the event pattern, choose **Edit**, make your changes, and then choose **Save**.
10. For **Service event bus**, select the **AWS default event bus**.
11. For **Select targets**, choose **Incident Manager response plan**.
12. For **Response plan**, choose a response plan.

    Note
    When selecting a response plan, all response plans that you own and have been shared with your account appear in the **Response plan** dropdown.
13. EventBridge can create the IAM role needed for your rule to run:

    - To create an IAM role automatically, choose **Create a new role for this specific resource**.
    - To use an IAM role that you created before, choose **Use existing role**.
14. (Optional) Enter one or more tags for the rule.
15. Choose **Create**.

Manually create incidents

Responders can manually track an incident using the Incident Manager console by using a predefined response plan. Use the following steps to create an incident.

1. Open the Incident Manager console.
2. Choose **Start incident**.
3. For **Response plan**, choose a response plan from the list.
4. (Optional) To override the title provided by the defined response plan, enter an **Incident title**.
5. (Optional) To override the impact provided by the defined response plan, enter the **Impact** of the incident.

Incident tracking

AWS Systems Manager Incident Manager tracks your incidents from the moment they're detected to resolution and through post incident analysis. You can find all incidents on the **Incident list** page, with links directly to the **Incident details**.

Topics

Incident list

The **Incident list** page contains three sections. **Open incidents**, **Resolved incidents**, and **Analyses**. You can manually track new incidents and create analyses from this page. To learn more about manually tracking an incident, see Manually create incidents (p. 29) in the *Incident creation* section of this guide. To learn about post incident analysis, see the Post-incident analysis (p. 34) section of this guide.

The **Incident details** displays **Open incidents** in tiles with the title, impact, duration, and chat channel. After you resolve an incident, it moves to the **Resolved incidents** list. **Analyses** are in the second tab under **Open incidents**.

Incident details

The Incident Manager incident details page is an incident responder's point of reference for all things involved in an incident. The incident details page automatically populates incidents as they're created.

The top banner of every incident details page includes the **Incident title**, **Impact**, **Chat channel**, and **Duration**. You can edit the incident title, impact, and chat channel by choosing **Edit** in the top right of the banner.

The incident details page has seven tabs, making it easier for responders to locate and view information during an incident. The tabs display a counter in the tab name, indicating the number of updates to the tab. For more information about the contents each tab and how it works, continue reading.

Tabs

Overview

The **Overview** tab is the landing page for responders. It contains the summary, a list of recent timeline events, and the current runbook step.

Responders use the **Summary** to catch up on what actions have been taken, the results of changes, possible next steps, and information about the impact of the incident. To update the summary, choose **Edit** in the top-right corner of the **Summary** section. If multiple responders are editing the summary field simultaneously, the responder who submits their edits last overwrites all other input.

Recent timeline events is the next section of the Overview tab. Incident Manager populates the timeline with the five most recent events. Use this section to understand the status of the incident and what has recently occurred. To view a complete timeline, continue to the **Timeline** tab.

The overview page also displays the **Current runbook step**. This step may be an automatic step running in your AWS environment or it may be a set of manual instructions for responders. To view the complete runbook, including prior and upcoming steps, continue to the **Runbook** tab.

Metrics

The **Metrics** tab contains vital information about your AWS hosted applications and systems. Incident Manager uses Amazon CloudWatch to populate the metrics and alarm graphs found on this page. To learn more about incident management best practices for defining alarms and metrics, see Monitoring (p. 16) in the *Incident planning* section of this user guide.

To add metrics

- Choose **Add** in the upper right corner of this tab.
  - Add a metric from an existing CloudWatch dashboard by selecting **From existing CloudWatch dashboard**.
    a. Choose a **Dashboard**. This adds all metrics and alarms that are part of the dashboard.
    b. (Optional) You can **Select metrics** from the dashboard.
  - Add a single metric by selecting **From CloudWatch** and pasting a metric source. To copy a metric source:
    a. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
    b. In the navigation pane, choose **Metrics**.
    c. On the **All metrics** tab, enter a search term in the search field, such as a metric name or resource name, and press **Enter**.

       For example, if you search for the `CPUUtilization` metric, you see the namespaces and dimensions with this metric.
    d. Choose one of the results for your search to view the metrics.
    e. Choose the **Source** tab and copy the source.

Metric alarm graphs can only be added to the incident details through the related response plan or by selecting **From existing CloudWatch dashboard** when adding a metric.

To remove metrics, choose **Remove** and then choose metrics from the provided **Metrics** dropdown.

Timeline

Use the **Timeline** tab to track events that occur during an incident. Incident Manager automatically populates timeline events that identify significant occurrences during the incident. Responders can add custom events based on occurrences detected manually. During the post incident analysis, the timeline tab provides valuable insights into how to better prepare and respond to incidents in the future. For more information about post incident analysis, see Post-incident analysis (p. 34).

To add a custom timeline event, choose **Add**. Select a date using the calendar and enter a time; all times are in your local time. Provide a brief description of the event that appears in the timeline.

To edit an existing custom event, select the event on the timeline and choose **Edit**. You can change the time, date, and description of *custom* events. You can only edit custom events.

Runbook

The runbook tab of the incident details displays the list of steps that the runbook automatically takes or responders manually perform. The steps expand as they become the current step, displaying information required to complete the step or details about what the step does. Automatic runbook steps resolve after the automation is complete. Manual steps require the responders to choose **Next step** at the bottom of the step. After a step has completed, the step output appears as a dropdown.

To navigate to the runbook definition in Systems Manager, choose the runbook's title under **Runbook**. To navigate to the running instance of the runbook in Systems Manager, choose the execution details under **Runbook details**. These pages display the template used to start the runbook and the specific details of the currently running instance of the automation document.

Incident Manager starts runbooks automatically using Systems Manager. To keep track of runbooks started in Systems Manager, we recommend copying the runbook's ARN (Amazon Resource Name) and adding it as a related item in the Related items (p. 32) tab of the incident details.

Engagements

The **Engagements** tab of the incident details drives engagement of responders and teams. From this tab you can see who has been engaged, who has responded, and responders who are going to be engaged as part of an escalation plan. Responders can engage other contacts directly from this tab. To learn more about creating contacts and escalation plans, see the Contacts (p. 16) and Escalation plans (p. 18) sections of this guide.

You can configure response plans with contacts and escalation plans to automatically start engagement at the beginning of an incident. To learn more about configuring response plans, see the Response plans (p. 24) section of this guide.

You can find information about each contact in the table in the contacts tab. The following are descriptions of each column in the table:

- **Name** – Links to the contact's details page that displays the contact's contact methods and engagement plan.
- **Escalation plan** – Links to the escalation plan used to engage the contact.
- **Engaged** – Displays when the contact was engaged or when they will be engaged.
- **Acknowledged** – Displays if the contact has acknowledged the engagement.

To acknowledge an engagement the responder can do one of the following:

- Phone call – Press 1 when prompted.
- SMS – Reply to the message with the provided code or enter the provided code on the **Engagements** tab of the incident.
- Email – Enter the provided code on the **Engagements** tab of the incident.

Related items

The **Related Items** tab is a place for responders to gather useful resources used during the mitigation of the incident. These resources can be an ARN or a link directly to an external resource. The table displays a descriptive title and ARN or link details.

Adding a resource

1. Choose **Add** in the upper right corner of the **Related items** tab.
2. Enter a descriptive **Title**. This title helps responders understand what the ARN or link is.
3. In **Detail**, provide an ARN or a link to an external resource.
4. Choose **Add**.

Properties

The properties tab provides details about the incident. You can view the following details:

- Status – Describes the current status of the incident. The incident can be **Open** or **Resolved**
- Start time – The time when the incident was created in Incident Manager.
- Resolved time – The time that the incident was resolved in Incident Manager
- Amazon Resource Name (ARN) – The ARN of the incident. Use this when referencing the incident via chat or CLI commands.
- Response Plan – Identifies the response plan for the selected incident. Selecting the response plan opens the response plan's details page.
- Parent OpsItem – Identifies the OpsItem created as the parent of the incident. A parent OpsItem can have multiple related incidents and follow up action items. Selecting the parent OpsItem opens the parent OpsItems details page in OpsCenter.
- Analysis – Identifies the analysis created from this incident. Create an analysis from a resolved incident to improve your incident response process. Selecting the analysis opens the analysis details page.
- Owner – The account that the incident was created in.

Post-incident analysis

Post-incident analysis guides you through identifying improvements to your incident response, including time to detection and mitigation. An analysis can also help you understand the root cause of the incidents. Incident Manager creates recommended *action items* to improve your incident response.

Benefits of a post-incident analysis

- Improve incident response
- Understand the root cause of the problem
- Address root causes with deliverable action items
- Analyze the impact of incidents
- Capture and share learnings within an organization

What not to use an analysis for

An analysis is blameless and doesn't call out people by name.

"Regardless of what we discover, we understand and truly believe that everyone did the best job they could, given what they knew at the time, their skills and abilities, the resources available, and the situation at hand." - Norm Kerth, Project Retrospectives: A Handbook for Team Review

Analysis details

The analysis details page guides you through gathering information, assessing improvements, and creating action items. The analysis details page is similar to the incident details with some key differences such as historical metrics, editable timeline, and questions to improve future incidents.

Overview

The overview is a summary of the incident. This summary includes background, what occurred, why it happened, how it was mitigated, duration, and key action items to prevent the incident from happening again. The overview is high level. You'll explore more details in the **Questions** tab of the analysis.

Metrics

Use the metrics tab to visualize key metrics in your application over the duration of the incident. You can add metric graphs here that have one or more metrics depicted in the same graph. Metrics used during an incident are automatically populated on this tab. We recommend you adding a description, title, and annotations of key timepoints during the incident.

Some key time points you can consider when analyzing a metric graph:

- Deployment change
- Configuration change
- Incident start time
- Alarm time
- Time of engagement
- Mitigation start time
- Incident resolved time

Limitations

- CloudWatch alarms and metric expressions aren't imported from an incident.
- Metrics that are in a region that Incident Manager doesn't support aren't imported from the incident.
- Metrics in application accounts require configuration of the `CloudWatch-CrossAccountSharingRole` prior to creating the analysis. For more information about the role, see Cross-Account Cross-Region CloudWatch console in the CloudWatch user guide.

Timeline

Describe key time points on the timeline as you dive deeper into understanding the incident. The incidents timeline is automatically populated on this tab. You can delete timepoints that aren't relevant to the analysis. You can also add and edit time points to more accurately describe the incident and its impact.

Use the timeline tab to answer questions you find on the **Questions** tab about the incident response.

Questions

Use Incident Manager questions to improve the time to resolution of incidents in your application and reduce the occurrence of incidents. As you answer questions, update the **Metrics** and **Timeline** tabs for accuracy. The questions focus on these key aspects of incident response:

- Detection – Could you improve time to detection? Are there updates to metrics and alarms that would detect the incident sooner?
- Diagnosis – Can you improve the time to diagnosis? Are there updates to your response plans or escalation plans that would engage the correct responders sooner?
- Mitigation – Can you improve the time to mitigation? Are there runbook steps that you could add or improve?
- Prevention – Can you prevent future incidents from occurring? To discover the root causes of an incident, Amazon uses the 5-Whys approach in problem investigation.

Actions

Incident Manager creates recommended action items for you to review as you complete the questions. You can choose to accept and complete these actions from this tab or you can dismiss these actions. You can review dismissed action items by choosing **Dismissed action items**. Action items are a type of OpsItem that are linked to the analysis and incident in OpsCenter.

Checklist

Before closing an analysis, use the checklist to review actions that a responder should take. As responders complete actions in the checklist, the icon next to the action changes from an ellipse to a check-mark, indicating that the action is complete. If you haven't completed checklist items, Incident Manager displays a message to confirm the responder wants to close the analysis without completing it.

Analysis templates

An analysis template provides a set of questions that dive deep into the root cause of incidents. You can use your answers to these questions to improve application performance and incident response.

AWS standard template

Incident Manager provides a standard template of questions based on AWS incident response and problem analysis best practices. The standard AWS template provided by Incident Manager is `AWSIncidents-PostIncidentAnalysisTemplate`.

Create an analysis template

We encourage you to use the default template and add additional questions or sections that are appropriate for your use cases. Create analysis templates based on the default template in your management account and duplicate them to each Region where you have enabled Incident Manager.

Create an analysis template

1. Download `AWSIncidents-PostIncidentAnalysisTemplate` by calling the `GetDocument` action using the parameter "Name" `AWSIncidents-PostIncidentAnalysisTemplate`. For more information about the `GetDocument` syntax, see Systems Manager API Reference.
2. The content in the response contains the JSON building blocks for the analysis. Use the question building blocks to insert additional questions in the analysis. We recommend that you add questions or sections in the `Incident questions` section.
3. To create the new template, use the `CreateDocument` operation with the updated JSON from the previous step. You must include `DocumentFormat: "JSON"`, `DocumentType: "ProblemAnalysisTemplate"`, and `Name: "`*`Analysis_Template_Name`*.

Create an analysis

1. To create an analysis, choose **Create analysis** from the incident details page of a closed incident.
2. Choose the analysis template and provide a descriptive name of the analysis.
3. Choose **Create**.

Incident Manager tutorials

These AWS Systems Manager Incident Manager tutorials help you build a more robust incident management system. These tutorials cover common activities that occur during an incident or support incident response.

Topics

On-call rotations

An on-call rotation uses a schedule to rotate through a group of on-call contacts, ensuring that there is always an contact on-call. On-call contacts are tasked with monitoring and immediately responding when an incident happens. A new contact is assigned as the on-call contact according to the schedule you set. Rotating the on-call contact prevents alert fatigue. This tutorial walks you through how to configure an on-call rotation using Amazon S3, AWS CloudFormation, and AWS Systems Manager Incident Manager.

This on-call rotation script automatically replaces the first contact in an escalation plan with the next on-call contact from a list of provided contacts. The frequency and timing that the contacts are rotated is customizable.

> **Note**
> This tutorial uses Amazon S3, CloudFormation, AWS Lambda, and Amazon CloudWatch Events. You will incur costs from these services by doing this tutorial.

Set up an on-call rotation

Prerequisites

- A configured escalation plan with one or more contacts. For more information about configuring an escalation plan, see the Escalation plans (p. 18) section of this user guide.
- Two or more configured contacts. These contacts are the group of on-call contacts used during this set up. For more information about configuring contacts, see the Contacts (p. 16) section of this user guide.

Download the on-call rotation script

The `OncallRotationScript` zip file contains both the CloudFormation YAML template and the python boto3 layer required for the AWS Lambda script.

1. Download the `oncallRotationScript.zip` file.
2. Unzip the `oncallRotationScript.zip` file.

Create an S3 bucket and upload the boto3 layer

1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
2. Choose **Create bucket**.

3. Give the bucket a unique name.
4. Select a region. This must be the same region that you create your AWS CloudFormation stack in later in this tutorial.
5. Choose **Block *all* public access**.
6. Disable **Bucket Versioning**.
7. (Optional) Add tags.
8. Disable **Server-side encryption**.
9. Choose **Create bucket**.
10. Choose the bucket that you just created.
11. Choose **Upload**.
12. Choose **Add files**, select the `boto3_layer.zip` file, and choose **Upload**.

Note
Note the bucket name and the object key. You will use these while configuring your CloudFormation stack.

Configure and deploy the CloudFormation stack

1. Open the CloudFormation console at https://console.aws.amazon.com/cloudformation.
2. Choose **Stacks** from the left navigation menu.
3. Choose **Create stack**, and then **With new resources (standard)**.
4. Choose **Template Is ready**.
5. Choose **Upload a template file**.
6. Choose **Choose file** and select the `oncall_updater.yml` file.
7. Choose **Next**.
8. Enter a **Stack Name**.
9. Set the following parameters in the **Parameters** section:

    a. **BotoLayerS3Bucket** – Enter the name of the bucket that you created in the previous section for the boto3 layer.

    b. **BotoLayerS3Key** – Provide the object key of the boto3 layer file that you uploaded to your Amazon S3 bucket. The object key is the name of the boto3 layer file, boto3_layer.zip. If you changed the name of the file, enter that name instead.

    c. **ContactRotation** – Enter a comma separated list of Amazon Resource Names (ARNs) for each contact in the on-call rotation. You can find the ARN for each contact on the contact's details page. The following string is an example of a comma separated list of contact ARNs:

    ```
    arn:aws:ssm-contacts:us-east-1:123456789101:contact/arosalez,arn:aws:ssm-contacts:us-east-1:123456789101:contact/csalazar,arn:aws:ssm-contacts:us-east-1:123456789101:contact/dramirez
    ```

    d. **CronExpression** – Provide a cron expression that defines when the on-call contact will be rotated. A cron expression is defined in GMT. The following example will rotate the on-call contact every Thursday at 17:00 GMT:

    ```
    00 17 ? * THU *
    ```

    e. **EscalationPlanArn** – Provide the ARN of the escalation plan you will use for the on-call rotation. The following example is the ARN of an escalation plan named test-plan:

    ```
    arn:aws:ssm-contacts:us-east-1:123456789101:contact/test-plan
    ```

10. Choose **Next**.
11. (Optional) Provide tags.
12. Choose **Next**.
13. Review the details of the stack and choose **Create stack**.

You've now completed set up of the on-call rotation script. This script automatically changes to the next contact in your **ContactRotation** list on the schedule you provided with the cron expression.

Update an on-call rotation

In case that your on-call changes you can update the parameters of the CloudFormation stack to represent the new on-call rotation. Use the following steps to update any of the on-call rotation parameters.

1. Open the CloudFormation console at https://console.aws.amazon.com/cloudformation.
2. Select the stack that you created when you set up the on-call rotation.
3. Choose **Update**.
4. Choose **Use current template**, and then **Next**.
5. Update the **Parameters** fields to update your on-call rotation.
   - Change the order of the **ContactRotation**.
   - Add or remove contacts from the **ContactRotation**.
   - Update the schedule using the **CronExpression**.
   - Change the escalation plan used by updating the **EscalationPlanArn**.
6. Choose **Next** on the **Parameters** page.
7. Choose **Next** on the **Configure stack options** page.
8. Select **I acknowledge that AWS CloudFormation might create IAM resources.**, and then **Update stack**.

Delete on-call rotation resources

If you no longer wish to use the on-call rotation, you can delete the CloudFormation stack to clean up all of the resources created using CloudFormation. Deleting the stack won't delete the contacts or escalation plan related to the on-call rotation.

1. Open the CloudFormation console at https://console.aws.amazon.com/cloudformation.
2. Select the stack that you created while setting up the on-call rotation.
3. Choose **Delete**.
4. Choose **Delete stack**.

After choosing to delete the stack, the IAM role, Lambda function, and CloudWatch Events rule are deleted according to their deletion policy.

Manage security incidents

You can use AWS Security Hub, Amazon EventBridge, and Incident Manager together to identify and manage security incidents in your AWS-hosted-applications. This tutorial walks you through configuring an EventBridge rule that creates an incident based on Security Hub automatically sent findings.

Note
This tutorial uses EventBridge Security Hub. You may incur costs from using these services.

Prerequisites

- Set up Security Hub. For more information, see Setting up AWS Security Hub.
- Create or update findings in Security Hub. For more information, see Findings in AWS Security Hub.
- Configure a response plan that Incident Manager will use as the template when creating your security incident. For more information, see Incident preparation (p. 15).

For this tutorial, we use a predefined pattern to create the EventBridge rule. To create the rule using a custom pattern, see Using a custom pattern to create the rule in the AWS Security Hub user guide.

Create an EventBridge rule

1. Open the EventBridge console at https://console.aws.amazon.com/events/.
2. In the navigation pane, under **Events**, choose **Rules**.
3. Choose **Create rule**.
4. Enter a **Name** and **Description** for the rule.
5. For **Define pattern**, choose **Event pattern**.
6. For **Event matching pattern**, choose **Pre-defined pattern by service**.
7. For **Service provider**, choose **AWS**.
8. For **Service name**, choose **Security Hub**.
9. For **Event type**, choose **Security Hub Findings - Imported**.
10. By default, EventBridge configures the event pattern without any filter values. For each attribute, the **Any *`attribute name`*** option is selected. Update these filters to create incidents based on the security findings that most impact your environment.
11. Under **Select targets**, choose **Incident Manager response plan**.
12. For **Response plan**, choose a response plan to use as a template for created incidents.
13. EventBridge can create the IAM role needed for your rule to run:
    - To create an IAM role automatically, choose **Create a new role for the specific resource**
    - To use an IAM role that already exists in your account, choose **Use existing role**.
14. (Optional) Enter one or more tags for the rule.
15. Choose **Create**.

Now that you've created this EventBridge rule, security findings that match the attribute values you defined will create incidents in Incident Manager. You can triage, manage, monitor, and create post-incident analysis from these incidents.

Security in AWS Systems Manager Incident Manager

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

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

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs. To learn about the compliance programs that apply to AWS Systems Manager Incident Manager, 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 Incident Manager. The following topics show you how to configure Incident Manager to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Incident Manager resources.

Topics

Data Protection in Incident Manager

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

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

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We recommend TLS 1.2 or later.
- Set up API and user activity logging with AWS CloudTrail.
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing personal data that is stored in Amazon S3.
- If you require FIPS 140-2 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-2.

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

By default, Incident Manager encrypts data in transit using SSL/TLS (Secure Socket Layers/Transport Layer Security).

Data encryption

Incident Manager uses AWS Key Management Service (AWS KMS) keys to encrypt your Incident Manager resources. For more information about AWS KMS, see the AWS KMS Developer Guide. AWS KMS combines secure, highly available hardware and software to provide a key management system scaled for the cloud. Incident Manager encrypts your data using your specified key and encrypts metadata using an AWS owned key. To use Incident Manager, you must set up your replication set, which includes setting up encryption. Incident Manager requires data encryption for use.

You can use an AWS owned key to encrypt your replication set or you can use your own customer managed key to encrypt the Regions in your replication set. If you use customer managed keys, you use the AWS KMS console or AWS KMS APIs to centrally create the customer managed keys and define the key policies that control how Incident Manager can use the customer managed keys. When you use a customer managed key for encryption with Incident Manager, the AWS KMS customer managed key must be in the same Region as the resources. To learn more about setting up data encryption in Incident Manager, see Get prepared wizard (p. 6).

There are additional charges for using AWS KMS customer managed keys. For more information, see AWS KMS concepts - KMS keys in the AWS Key Management Service Developer Guide and AWS KMS pricing.

To allow Incident Manager to use your customer managed key to encrypt your data, you must add the following policy statements to the key policy of your customer managed key. To learn more about setting up and changing the key policy in your account, see Using key policies in AWS KMS. The policy provides the following permissions:

- Allows Incident Manager to perform read-only operations to find the CMK for Incident Manager in your account.
- Allows Incident Manager to use the CMK to create grants and describe the key, but only when it's acting on behalf of principals in the account who have permission to use Incident Manager. If the principals specified in the policy statement don't have permission to use the KMS keys and to use Incident Manager, the call fails, even when it comes from the Incident Manager service.

```
{
"Sid": "Allow CreateGrant through AWS Systems Manager Incident Manager",
"Effect": "Allow",
```

```
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:user/ssm-lead"
      },
      "Action": [
        "kms:CreateGrant",
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:ViaService": [
            "ssm-incidents.amazonaws.com",
            "ssm-contacts.amazonaws.com"
          ]
        }
      }
    }
```

Replace the `Principal` value with the IAM principal that created your replication set.

Incident Manager uses an encryption context in all requests to KMS for cryptographic operations. You can use this encryption context to identify CloudTrail log events where Incident Manager uses your KMS keys. Incident Manager uses the following encryption context:

- `contactArn=`*`ARN of the contact or escalation plan`*

Identity and Access Management for AWS Systems Manager Incident Manager

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 Incident Manager resources. IAM is an AWS service that you can use with no additional charge.

Topics

Audience

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

Service user – If you use the Incident Manager service to do your job, then your administrator provides you with the credentials and permissions that you need. As you use more Incident Manager 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 Incident Manager, see Troubleshooting AWS Systems Manager Incident Manager identity and access (p. 56).

Service administrator – If you're in charge of Incident Manager resources at your company, you probably have full access to Incident Manager. It's your job to determine which Incident Manager 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 Incident Manager, see How AWS Systems Manager Incident Manager works with IAM (p. 47).

IAM administrator – If you're an IAM administrator, you might want to learn details about how you can write policies to manage access to Incident Manager. To view example Incident Manager identity-based policies that you can use in IAM, see Identity-based policy examples for AWS Systems Manager Incident Manager (p. 52).

Authenticating with identities

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

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

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

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

AWS account root user

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

IAM users and groups

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

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

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

IAM roles

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

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

- **Temporary IAM user permissions** – An IAM user can assume an IAM role to temporarily take on different permissions for a specific task.
- **Federated user access** – Instead of creating an IAM user, you can use existing identities from AWS Directory Service, your enterprise user directory, or a web identity provider. These are known as *federated users*. AWS assigns a role to a federated user when access is requested through an identity provider. For more information about federated users, see Federated users and roles in the *IAM User Guide*.
- **Cross-account access** – You can use an IAM role to allow someone (a trusted principal) in a different account to access resources in your account. Roles are the primary way to grant cross-account access. However, with some AWS services, you can attach a policy directly to a resource (instead of using a role as a proxy). To learn the difference between roles and resource-based policies for cross-account access, see How IAM roles differ from resource-based policies in the *IAM User Guide*.
- **Cross-service access** – Some AWS services use features in other AWS services. For example, when you make a call in a service, it's common for that service to run applications in Amazon EC2 or store objects in Amazon S3. A service might do this using the calling principal's permissions, using a service role, or using a service-linked role.
  - **Principal permissions** – When you use an IAM user or role to perform actions in AWS, you are considered a principal. Policies grant permissions to a principal. When you use some services, you might perform an action that then triggers another action in a different service. In this case, you must have permissions to perform both actions. To see whether an action requires additional dependent actions in a policy, see Actions, resources, and condition keys for AWS Systems Manager Incident Manager in the *Service Authorization Reference*.
  - **Service role** – A service role is an IAM role that a service assumes to perform actions on your behalf. An IAM administrator can create, modify, and delete a service role from within IAM. For more information, see Creating a role to delegate permissions to an AWS service in the *IAM User Guide*.
  - **Service-linked role** – A service-linked role is a type of service role that is linked to an AWS service. The service can assume the role to perform an action on your behalf. Service-linked roles appear in your IAM account and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.
- **Applications running on Amazon EC2** – You can use an IAM role to manage temporary credentials for applications that are running on an EC2 instance and making AWS CLI or AWS API requests. This is preferable to storing access keys within the EC2 instance. To assign an AWS role to an EC2 instance and make it available to all of its applications, you create an instance profile that is attached to the instance. An instance profile contains the role and enables programs that are running on the EC2 instance to get temporary credentials. For more information, see Using an IAM role to grant permissions to applications running on Amazon EC2 instances in the *IAM User Guide*.

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

Managing access using policies

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

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

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

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

Identity-based policies

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

Identity-based policies can be further categorized as *inline policies* or *managed policies*. Inline policies are embedded directly into a single user, group, or role. Managed policies are standalone policies that you can attach to multiple users, groups, and roles in your AWS account. Managed policies include AWS managed policies and customer managed policies. To learn how to choose between a managed policy or an inline policy, see Choosing between managed policies and inline policies in the *IAM User Guide*.

Resource-based policies

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

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

Access control lists (ACLs)

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

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

Other policy types

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

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

Multiple policy types

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

How AWS Systems Manager Incident Manager works with IAM

Before you use IAM to manage access to Incident Manager, learn what IAM features are available to use with Incident Manager.

IAM features you can use with AWS Systems Manager Incident Manager

IAM feature	Incident Manager support
Identity-based policies (p. 48)	Yes
Resource-based policies (p. 48)	Yes
Policy actions (p. 49)	Yes
Policy resources (p. 50)	Yes
Policy condition keys (p. 50)	No
ACLs (p. 51)	No
ABAC (tags in policies) (p. 51)	No

IAM feature	Incident Manager support
Temporary credentials (p. 51)	Yes
Principal permissions (p. 52)	Yes
Service roles (p. 52)	Yes
Service-linked roles (p. 52)	Yes

To get a high-level view of how Incident Manager and other AWS services work with most IAM features, see AWS services that work with IAM in the *IAM User Guide*.

Incident Manager doesn't support policies that deny access to resources shared using AWS RAM.

Identity-based policies for Incident Manager

Supports identity-based policies	Yes

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

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. You can't specify the principal in an identity-based policy because it applies to the user or role to which it is attached. To learn about all of the elements that you can use in a JSON policy, see IAM JSON policy elements reference in the *IAM User Guide*.

Identity-based policy examples for Incident Manager

To view examples of Incident Manager identity-based policies, see Identity-based policy examples for AWS Systems Manager Incident Manager (p. 52).

Resource-based policies within Incident Manager

Supports resource-based policies	Yes

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

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, an IAM administrator in the trusted account must also grant the principal entity (user or role) permission to access the resource. They 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*.

The Incident Manager service supports only two types of resource-based policies called using either the AWS RAM console or the PutResourcePolicy action, which is attached to a response plan or contact. This policy defines which principals can perform actions on the response plans, contacts, escalation plans, and incidents. Incident Manager uses resource based policies to share resources across accounts.

Incident Manager doesn't support policies that deny access to resources shared using AWS RAM.

To learn how to attach a resource-based policy to a response plan or contact, see Setting up cross-account functionality (p. 11).

Resource-based policy examples within Incident Manager

To view examples of Incident Manager resource-based policies, see Resource-based policy examples for AWS Systems Manager Incident Manager (p. 55),

Policy actions for Incident Manager

Supports policy actions	Yes

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

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

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

To see a list of Incident Manager actions, see Actions defined by AWS Systems Manager Incident Manager in the *Service Authorization Reference*.

Policy actions in Incident Manager use the following prefixes before the action:

```
ssm-incidents
ssm-contacts
```

To specify multiple actions in a single statement, separate them with commas.

```
"Action": [
      "ssm-incidents:GetResponsePlan",
      "ssm-contacts:GetContact"
         ]
```

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

```
"Action": "ssm-incidents:Get*"
```

To view examples of Incident Manager identity-based policies, see Identity-based policy examples for AWS Systems Manager Incident Manager (p. 52).

Incident Manager uses actions in two different namespaces, ssm-incidents and ssm-contacts. When creating policies for Incident Manager make sure to use the namespace correct for the action. SSM-Incidents is used for response plan and incident related action. SSM-Contacts is used for actions related to contacts and contact engagement. For example:

- `ssm-contacts:GetContact`
- `ssm-incidents:GetResponsePlan`

Policy resources for Incident Manager

Supports policy resources	Yes

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

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

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

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

To see a list of Incident Manager resource types and their ARNs, see Resources defined by AWS Systems Manager Incident Manager in the *Service Authorization Reference*. To learn with which actions you can specify the ARN of each resource, see Actions defined by AWS Systems Manager Incident Manager.

To view examples of Incident Manager identity-based policies, see Identity-based policy examples for AWS Systems Manager Incident Manager (p. 52).

Incident Manager resources are used to create incidents, collaborate in chat channels, resolve incidents, and engage responders. If a user has access to a response plan they have access to all incidents created from it. If a user has access to a contact or escalation plan they can engage the contact or contacts in the escalation plan.

Policy condition keys for Incident Manager

Supports policy condition keys	No

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

The `Condition` element (or `Condition` *block*) lets you specify conditions in which a statement is in effect. The `Condition` element is optional. You can create 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*.

AWS supports global condition keys and service-specific condition keys. To see all AWS global condition keys, see AWS global condition context keys in the *IAM User Guide*.

Access control lists (ACLs) in Incident Manager

Supports ACLs	No

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

Attribute-based access control (ABAC) with Incident Manager

Supports ABAC (tags in policies)	No

Attribute-based access control (ABAC) is an authorization strategy that defines permissions based on attributes. In AWS, these attributes are called *tags*. You can attach tags to IAM entities (users or roles) and to many AWS resources. Tagging entities and resources is the first step of ABAC. Then you design ABAC policies to allow operations when the principal's tag matches the tag on the resource that they are trying to access.

ABAC is helpful in environments that are growing rapidly and helps with situations where policy management becomes cumbersome.

To control access based on tags, you provide tag information in the condition element of a policy using the `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, or `aws:TagKeys` condition keys.

For more information about ABAC, see What is ABAC? in the *IAM User Guide*. To view a tutorial with steps for setting up ABAC, see Use attribute-based access control (ABAC) in the *IAM User Guide*.

Using temporary credentials with Incident Manager

Supports temporary credentials	Yes

Some AWS services don't work when you sign in using temporary credentials. For additional information, including which AWS services work with temporary credentials, see AWS services that work with IAM in the *IAM User Guide*.

You are using temporary credentials if you sign in to the AWS Management Console using any method except a user name and password. For example, when you access AWS using your company's single sign-on (SSO) link, that process automatically creates temporary credentials. You also automatically create temporary credentials when you sign in to the console as a user and then switch roles. For more information about switching roles, see Switching to a role (console) in the *IAM User Guide*.

You can manually create temporary credentials using the AWS CLI or AWS API. You can then use those temporary credentials to access AWS. AWS recommends that you dynamically generate temporary credentials instead of using long-term access keys. For more information, see Temporary security credentials in IAM.

Cross-service principal permissions for Incident Manager

Supports principal permissions	Yes

When you use an IAM user or role to perform actions in AWS, you are considered a principal. Policies grant permissions to a principal. When you use some services, you might perform an action that then triggers another action in a different service. In this case, you must have permissions to perform both actions. To see whether an action requires additional dependent actions in a policy, see Actions, resources, and condition keys for AWS Systems Manager Incident Manager in the *Service Authorization Reference*.

Service roles for Incident Manager

Supports service roles	Yes

A service role is an IAM role that a service assumes to perform actions on your behalf. An IAM administrator can create, modify, and delete a service role from within IAM. For more information, see Creating a role to delegate permissions to an AWS service in the *IAM User Guide*.

> **Warning**
> Changing the permissions for a service role might break Incident Manager functionality. Edit service roles only when Incident Manager provides guidance to do so.

Choosing an IAM role in Incident Manager

When you create a response plan resource in Incident Manager, you must choose a role to allow Incident Manager to run a Systems Manager automation document on your behalf. If you have previously created a service role or service-linked role, then Incident Manager provides you with a list of roles to choose from. It's important to choose a role that allows access to run your automation document instances. For more information, see Runbooks and automation (p. 22). When you create a AWS Chatbot chat channel to be used during an incident you can select a service role that allows you to use commands directly from chat. To learn more about creating chat channels for incident collaboration, see Chat channels (p. 19). To learn more about IAM policies in AWS Chatbot, see Managing permissions for running commands using AWS Chatbot in the *AWS Chatbot Administrator guide*.

Service-linked roles for Incident Manager

Supports service-linked roles	Yes

A service-linked role is a type of service role that is linked to an AWS service. The service can assume the role to perform an action on your behalf. Service-linked roles appear in your IAM account and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.

For information about creating or managing Incident Manager service-linked roles, see Using service-linked roles for Incident Manager (p. 58).

Identity-based policy examples for AWS Systems Manager Incident Manager

By default, IAM users and roles don't have permission to create or modify Incident Manager 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 actions on the resources that they need. The administrator must then attach those policies to the IAM users or groups that require those permissions.

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

Topics

Policy best practices

Identity-based policies are very powerful. They determine whether someone can create, access, or delete Incident Manager 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 Incident Manager 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 Incident Manager console

To access the AWS Systems Manager Incident Manager console, you must have a minimum set of permissions. These permissions must allow you to list and view details about the Incident Manager 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.

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.

To ensure that users and roles can resolve incident using the Incident Manager console, also attach the Incident Manager `IncidentManagerResolverAccess` AWS managed policy to the entities. For more information, see Adding permissions to a user in the *IAM User Guide*.

```
IncidentManagerResolverAccess
```

Allow users to view their own permissions

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

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Accessing a response plan

In this example, you want to grant an IAM user in your Amazon Web Services account access to one of your Incident Manager response plans, `exampleplan`. You also want to allow the user to add, update, and delete the response plan.

The policy grants the `ssm-incidents:ListResponsePlans`, `ssm-incidents:GetResponsePlan`, `ssm-incidents:UpdateResponsePlan` and `ssm-incident:ListResponsePlan` permissions to the user.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"ListResponsePlans",
         "Effect":"Allow",
         "Action":[
            "ssm-incidents:ListResponsePlans"
         ],
         "Resource":"arn:aws:ssm-incidents:::*"
      },
      {
```

```
         "Sid":"ViewSpecificResponsePlanInfo",
         "Effect":"Allow",
         "Action":[
            "ssm-incidents:GetResponsePlan"
         ],
         "Resource":"arn:aws:ssm-incidents:*:111122223333:response-plan/exampleplan"
      },
      {
         "Sid":"ManageResponsePlan",
         "Effect":"Allow",
         "Action":[
            "ssm-incidents:UpdateResponsePlan"
         ],
         "Resource":"arn:aws:ssm-incidents:*:111122223333:response-plan/exampleplan/*"
      }
   ]
}
```

Resource-based policy examples for AWS Systems Manager Incident Manager

Incident Manager doesn't support resource-based policies that deny access to resources shared using AWS RAM.

To learn how to create a response plan or contact, see Response plans (p. 24) and Contacts (p. 16).

Restricting Incident Manager response plan access by organization

The following example grants permissions to users in the organization with the organization ID: `o-abc123def45` to respond to incidents created using the response plan `myplan`.

The `Condition` block uses the `StringEquals` conditions and the `aws:PrincipalOrgID` condition key, which is an AWS Organizations specific condition key. For more information about these condition keys, see Specifying conditions in a policy.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "OrganizationAccess",
      "Effect": "Allow",
      "Principal": “*”,
      "Condition": {
         "StringEquals": {"aws:PrincipalOrgID" : ["o-abc123def45"] }
      },
      "Action": [
        "ssm-incidents:GetResponsePlan",
        "ssm-incidents:StartIncident",
        "ssm-incidents:UpdateIncidentRecord",
        "ssm-incidents:GetIncidentRecord",
        "ssm-incidents:CreateTimelineEvent",
        "ssm-incidents:UpdateTimelineEvent",
        "ssm-incidents:GetTimelineEvent",
        "ssm-incidents:ListTimelineEvents",
        "ssm-incidents:UpdateRelatedItems",
        "ssm-incidents:ListRelatedItems"
```

```
      ],
      "Resource": [
        "arn:aws:ssm-incidents:*:111122223333:response-plan/myplan",
        "arn:aws:ssm-incidents:*:111122223333:incident-record/myplan/*"
      ]
    }
  ]
}
```

Providing Incident Manager contact access to a principal

The following example grants permission to the principal with the ARN `arn:aws:iam::999988887777:root` to create engagements to the contact `mycontact`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PrincipalAccess",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::999988887777:root"
            },
            "Action": [
                "ssm-contacts:GetContact",
                "ssm-contacts:StartEngagement",
                "ssm-contacts:DescribeEngagement",
                "ssm-contacts:ListPagesByContact"
            ],
            "Resource": [
                "arn:aws:ssm-contacts:*:111122223333:contact/mycontact"
                "arn:aws:ssm-contacts:*:111122223333:engagement/mycontact/*"
            ]
        }
    ]
}
```

Troubleshooting AWS Systems Manager Incident Manager identity and access

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

Topics

I am not authorized to perform an action in Incident Manager

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 fictional `example-plan` resource but does not have the fictional `ssm-incident:GetResponsePlan` permissions.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform: ssm-incident:GetResponsePlan on resource: example-plan
```

In this case, Mateo asks his administrator to update his policies to allow him to access the `example-plan` resource using the `ssm-incidents:GetResponsePlan` 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 Incident Manager.

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 Incident Manager. 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 Incident Manager

To allow others to access Incident Manager, 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 Incident Manager.

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

I want to allow people outside of my Amazon Web Services account to access my Incident Manager 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 Incident Manager supports these features, see How AWS Systems Manager Incident Manager works with IAM (p. 47).
- To learn how to provide access to your resources across AWS accounts that you own, see Providing access to an IAM user in another AWS account that you own in the *IAM User Guide*.
- To learn how to provide access to your resources to third-party AWS accounts, see Providing access to AWS accounts owned by third parties in the *IAM User Guide*.
- To learn how to provide access through identity federation, see Providing access to externally authenticated users (identity federation) in the *IAM User Guide*.
- To learn the difference between using roles and resource-based policies for cross-account access, see How IAM roles differ from resource-based policies in the *IAM User Guide*.

Using service-linked roles for Incident Manager

AWS Systems Manager Incident Manager 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 Incident Manager. Service-linked roles are predefined by Incident Manager and include all the permissions that the service requires to call other AWS services on your behalf.

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

You can delete a service-linked role only after first deleting their related resources. This protects your Incident Manager resources because you can't inadvertently remove permission to access the resources.

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

Service-linked role permissions for Incident Manager

Incident Manager uses the service-linked role named **AWSServiceRoleforIncidentManager** – allows Incident Manager to manage Incident Manager incident records and related resources on your behalf.

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

- `ssm-incidents.amazonaws.com`

The role permissions policy allows Incident Manager to complete the following actions on the specified resources:

- Action: `ssm-incidents:ListIncidentRecords` on all resources related to the action.
- Action: `ssm-incidents:CreateTimelineEvent` on all resources related to the action.
- Action: `ssm:CreateOpsItem` on all resources related to the action.
- Action: `ssm:AssociateOpsItemRelatedItem` on `all resources related to the action.`
- Action: `ssm-contacts:StartEngagement` on all resources related to the action.

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 more information, see Service-Linked Role Permissions in the *IAM User Guide*.

Creating a service-linked role for Incident Manager

You don't need to manually create a service-linked role. When you create a replication set in the AWS Management Console, the AWS CLI, or the AWS API, Incident Manager creates the service-linked role for you.

If you delete this service-linked role, and then need to create it again, you can use the same process to recreate the role in your account. When you create a replication set, Incident Manager creates the service-linked role for you again.

Editing a service-linked role for Incident Manager

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

Deleting a service-linked role for Incident Manager

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. However, you must clean up the resources for your service-linked role before you can manually delete it.

To delete the service-linked role you must first delete the replication set. Deleting the replication set deletes all data created and stored in Incident Manager, including response plans, contacts, and escalation plans. You will also lose all previously created incidents. Any alarms and EventBridge rules pointing to deleted response plans will no longer create an incident on alarm or rule match. To delete the replication set you must delete every region in the set.

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

To delete the regions in the replication set used by the AWSServiceRoleforIncidentManager

1. Open the Incident Manager console and choose **Settings** from the left navigation.
2. Select a region in the **Replication set**.
3. Choose **Delete**.
4. To confirm deletion of the region, enter the Region name and choose **Delete**.
5. Repeat these steps until you have deleted all Regions in your replication set. When deleting the final Region, the console informs you that it deletes the replication set with it.

To manually delete the service-linked role using IAM

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

Supported Regions for Incident Manager service-linked roles

Incident Manager supports using service-linked roles in all of the Regions where the service is available. For more information, see AWS Regions and Endpoints.

AWS managed policies for AWS Systems Manager Incident Manager

To add permissions to users, groups, and roles, it is easier to use AWS managed policies than to write policies yourself. It takes time and expertise to create IAM customer managed policies that provide your team with only the permissions they need. To get started quickly, you can use our AWS managed policies. These policies cover common use cases and are available in your AWS account. For more information about AWS managed policies, see AWS managed policies in the *IAM User Guide*.

AWS services maintain and update AWS managed policies. You can't change the permissions in AWS managed policies. Services occasionally add additional permissions to an AWS managed policy to support new features. This type of update affects all identities (users, groups, and roles) where the policy is attached. Services are most likely to update an AWS managed policy when a new feature is launched or when new operations become available. Services do not remove permissions from an AWS managed policy, so policy updates won't break your existing permissions.

Additionally, AWS supports managed policies for job functions that span multiple services. For example, the **ViewOnlyAccess** AWS managed policy provides read-only access to many AWS services and resources. When a service launches a new feature, AWS adds read-only permissions for new operations and resources. For a list and descriptions of job function policies, see AWS managed policies for job functions in the *IAM User Guide*.

AWS managed policy: `AWSIncidentManagerServiceRolePolicy`

You can't attach `AWSIncidentManagerServiceRolePolicy` to your IAM entities. This policy is attached to a service-linked role that allows Incident Manager to perform actions on your behalf. For more information, see Using service-linked roles for Incident Manager (p. 58).

This policy grants Incident Manager permissions to list incidents, create timeline events, create OpsItems, associate related items to OpsItems, and start engagements related to an incident.

Permissions details

This policy includes the following permissions.

- `ssm-incidents` – Allows principals to list incidents and create timeline events. This is required so responders can collaborate during an incident on the incident dashboard.

- `ssm` – Allows the principals to create OpsItems and associate related items. This is required to create a parent OpsItem when an incident starts.
- `ssm-contacts` – Allows principals to start engagements. This is required for Incident Manager to engage contacts during an incident.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "UpdateIncidentRecordPermissions",
            "Effect": "Allow",
            "Action": [
                "ssm-incidents:ListIncidentRecords",
                "ssm-incidents:CreateTimelineEvent"
            ],
            "Resource": "*"
        },
        {
            "Sid": "RelatedOpsItemPermissions",
            "Effect": "Allow",
            "Action": [
                "ssm:CreateOpsItem",
                "ssm:AssociateOpsItemRelatedItem"
            ],
            "Resource": "*"
        },
        {
            "Sid": "IncidentEngagementPermissions",
            "Effect": "Allow",
            "Action": "ssm-contacts:StartEngagement",
            "Resource": "*"
        }
    ]
}
```

AWS managed policy: `AWSIncidentManagerResolverAccess`

You can attach `AWSIncidentManagerResolverAccess` to your IAM entities to allow them to start, view, and update incidents. This also allows them to create customer timeline events and related items in the incident dashboard. You can also attach this policy to the AWS Chatbot service role or directly to your customer managed role associated with any chat channel used for incident collaboration. To learn more about IAM policies in AWS Chatbot, see Managing permissions for running commands using AWS Chatbot.

Permissions details

This policy includes the following permissions.

- `ssm-incidents` – Allows you to start incidents, list response plans, list incidents, update incidents, list timeline events, create custom timeline events, update custom timeline events, delete custom timeline events, list related items, create related items, and update related items.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "StartIncidentPermissions",
            "Effect": "Allow",
            "Action": [
                "ssm-incidents:StartIncident"
            ],
            "Resource": "*"
        },
        {
            "Sid": "ResponsePlanReadOnlyPermissions",
            "Effect": "Allow",
            "Action": [
                "ssm-incidents:ListResponsePlans",
                "ssm-incidents:GetResponsePlan"
            ],
            "Resource": "*"
        },
        {
            "Sid": "IncidentRecordResolverPermissions",
            "Effect": "Allow",
            "Action": [
                "ssm-incidents:ListIncidentRecords",
                "ssm-incidents:GetIncidentRecord",
                "ssm-incidents:UpdateIncidentRecord",
                "ssm-incidents:ListTimelineEvents",
                "ssm-incidents:CreateTimelineEvent",
                "ssm-incidents:GetTimelineEvent",
                "ssm-incidents:UpdateTimelineEvent",
                "ssm-incidents:DeleteTimelineEvent",
                "ssm-incidents:ListRelatedItems",
                "ssm-incidents:UpdateRelatedItems"
            ],
            "Resource": "*"
        }
    ]
}
```

Incident Manager updates to AWS managed policies

View details about updates to AWS managed policies for Incident Manager since this service began tracking these changes. For automatic alerts about changes to this page, subscribe to the RSS feed on the Incident Manager Document history page.

Change	Description	Date
AWSIncidentManagerResolverAccess (p. 61) – New policy	Incident Manager added a new policy to allow you to start incidents, list response plans, list incidents, update incidents, list timeline events, create custom timeline events, update custom timeline events, delete custom timeline events, list related items, create related items, and update related items.	April 26, 2021

Change	Description	Date
AWSIncidentManagerServiceRolePolicy (p. 60) – New policy	Incident Manager added a new policy to grant Incident Manager permissions to list incidents, create timeline events, create OpsItems, associate related items to OpsItems, and start engagements related to an incident.	April 26, 2021
Incident Manager started tracking changes	Incident Manager started tracking changes for its AWS managed policies.	April 26, 2021

Working with shared contacts and response plans

With contact sharing, as a contact owner, you can share contact information, escalation plans, and engagements with other AWS accounts or within an AWS organization. You can create and manage contacts and escalation plans centrally, and ensure that others can engage the correct contacts during an incident.

With response plan sharing, as a response plan owner, you can share a response plan and the related incidents with other AWS accounts or within an AWS organization. You can create and manage response plans centrally so that responders in consumer accounts can interact with incidents as they happen.

A contact or response plan owner can share contacts and response plans with:

- Specific AWS accounts inside or outside of its organization in AWS Organizations
- An organizational unit inside its organization in AWS Organizations
- Its entire organization in AWS Organizations

Contents

Prerequisites for sharing contacts and response plans

To share a contact or response plan with your organization or organizational unit in AWS Organizations:

- You must own the resource in your AWS account. You can't share a contact or response plan that has been shared with you.
- You must enable sharing with AWS Organizations. For more information, see Enable Sharing with AWS Organizations in the *AWS RAM User Guide*.

Related services

Contact and response plan sharing integrates with AWS Resource Access Manager (AWS RAM). With AWS RAM, you can share your AWS resources with any AWS account or through AWS Organizations. You share resources that you own by creating a *resource share*. A resource share specifies the resources to share, and the consumers with whom to share them. Consumers can be individual AWS accounts, organizational units, or an entire organization in AWS Organizations.

For more information about AWS RAM, see the *AWS RAM User Guide*.

Sharing a contact or response plan

After you share a response plan, the consumers have access to all past, current, and future incidents created using that response plan.

After you share a contact, the consumers have access to the contact information, engagement plan, escalation plans, and engagements that occur during an incident. Consumers can also engage a contact or escalation plan during an incident.

If you're part of an organization in AWS Organizations and sharing within your organization is enabled, consumers in your organization are automatically granted access to the shared contact or response plan. Otherwise, consumers receive an invitation to join the resource share and are granted access to the shared contact or response plan after accepting the invitation.

You can share a contact or response plan that you own by using the AWS RAM console or the AWS CLI.

To share a contact or response plan that you own by using the AWS RAM console

See Creating a Resource Share in the *AWS RAM User Guide*.

To share a contact or response plan that you own by using the AWS CLI

Use the create-resource-share command.

Stop sharing a shared contact or response plan

When a resource owner stops sharing a contact or response plan with a consumer, the contacts, response plans, escalation plans, engagements, and incidents no longer appear in the consumer's console.

> **Note**
> The consumer continues to see the contacts, response plans, escalation plans, engagements, or incidents without updates, if they're viewing them in the console, until they refresh the page or navigate away from the page.

To stop sharing a shared contact or response plan that you own, you must remove it from the resource share. You can do this by using the AWS RAM console or the AWS CLI.

To stop sharing a shared contact or response plan that you own by using the AWS RAM console

See Updating a Resource Share in the *AWS RAM User Guide*.

To stop sharing a shared contact or response plan that you own by using the AWS CLI

Use the disassociate-resource-share command.

Identifying a shared contact or response plan

Owners and consumers can identify shared contacts and response plans by using the Incident Manager console and AWS CLI.

To identify a shared contact or response plan by using the Incident Manager console

Note
Contacts, response plans, escalation plans, engagements, and incidents are generally not identifiable as a shared resource in the Incident Manager console. In places where the Amazon Resource Name (ARN) is visible, the ARN contains the owner's account ID.

To identify a shared contact or response plan by using the AWS CLI

Use the ListResponsePlans or ListContacts commands. The command returns the contacts and response plans that you own and contacts and response plans that are shared with you. The ARN shows the AWS account ID of the contact or response plan owner.

Shared contact and response plan permissions

Permissions for owners

Owners can update, view, share, stop sharing, and use contacts and response plans. Contacts and response plans include related engagements and incidents.

Permissions for consumers

Consumers can use and view only response plans and contacts. Contacts and response plans include related engagements and incidents.

Billing and metering

The owner of the resource is billed for the resource. Consumers aren't billed for resources shared with them. There aren't extra costs associated with sharing a resource.

Instance limits

Sharing a resource doesn't affect the limits of the resource in the owner's or consumer's account. Only the owner's account is used to calculate the limits of the resource.

Logging AWS Systems Manager Incident Manager API calls using AWS CloudTrail

AWS Systems Manager Incident Manager integrates with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Incident Manager. CloudTrail captures all API calls for Incident Manager as events. The calls captured include calls from the Incident Manager console and code calls to the Incident Manager API operations. If you create a trail, you can turn on continual delivery of CloudTrail events to an Amazon S3 bucket, including events for Incident Manager. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to Incident Manager, the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, see the AWS CloudTrail User Guide.

Incident Manager information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in Incident Manager, that activity is recorded in a CloudTrail event along with other AWS service events in **Event**

history. You can view, search, and download recent events in your AWS account. For more information, see Viewing events with CloudTrail Event history.

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

- Overview for creating a trail
- CloudTrail supported services and integrations
- Configuring Amazon SNS notifications for CloudTrail
- Receiving CloudTrail log files from multiple regions and Receiving CloudTrail log files from multiple accounts

CloudTrail logs all Incident Manager actions and Incident Manager documents all actions in the AWS Systems Manager Incident Manager API Reference. For example, calls to the `CreateResponsePlan`, `ActivateDevice`, and `StartIncident` actions generate entries in the CloudTrail log files.

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.

Understanding Incident Manager 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 `StartIncident` action.

```
    {
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "1234567890abcdef0",
        "arn": "arn:aws:iam::246873129580111122223333:user/nikki_wolf",
        "accountId": "abcdef01234567890",
        "accessKeyId": "021345abcdef6789",
        "userName": "nikki_wolf"
    },
    "eventTime": "2021-04-22T23:20:10Z",
    "eventSource": "gamma-ssm-incidents.amazonaws.com",
    "eventName": "StartIncident",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/2.0.58 Python/3.7.4 Darwin/19.6.0 exe/x86_64 command/
ssmincidents.start-incident",
    "requestParameters": {
```

```
        "responsePlanArn": "arn:aws:ssm-incidents::555555555555:response-plan/security-
test-response-plan-non-dedupe-v1",
        "clientToken": "12345678-1111-2222-3333-abcdefghijkl"
    },
    "responseElements": {
        "incidentRecordArn": "arn:aws:ssm-incidents::444455556666:incident-record/security-
test-response-plan-non-dedupe-v1/abcdefgh-abcd-1234-1234-1234567890"
    },
    "requestID": "abcdefgh-1234-abcd-1234-1234567abcdef",
    "eventID": "12345678-1234-1234-abcd-abcdef1234567",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "12345678901234567"
}
```


The following example shows a CloudTrail log entry that demonstrates the `DeleteContactChannel` action.


```
    {
    "eventVersion":"1.08",
    "userIdentity":{
        "type":"IAMUser",
        "principalId":"1234567890abcdef0",
        "arn":"arn:aws:iam::246873129580111122223333:user/nikki_wolf",
        "accountId":"abcdef01234567890",
        "accessKeyId":"021345abcdef6789",
        "userName":"nikki_wolf"
    },
    "eventTime":"2021-04-08T02:27:21Z",
    "eventSource":"ssm-contacts.amazonaws.com",
    "eventName":"DeleteContactChannel",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"192.0.2.0",
    "userAgent":"Apache-HttpClient/UNAVAILABLE (Java/1.8.0_282)",
    "requestParameters":{
        "contactChannelId":"arn:aws:ssm-contacts:us-west-2:555555555555:device/bnuomysohc/
abcdefgh-abcd-1234-1234-1234567890"
    },
    "responseElements":null,
    "requestID":"abcdefgh-1234-abcd-1234-1234567abcdef",
    "eventID":"12345678-1234-1234-abcd-abcdef1234567",
    "readOnly":true,
    "eventType":"AwsApiCall",
    "managementEvent":true,
    "eventCategory":"Management",
    "recipientAccountId":"12345678901234567"
}
```


Compliance validation for AWS Systems Manager Incident Manager

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

To learn whether Incident Manager or other AWS services are 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 AWS services 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 baseline environments on AWS that are security and compliance focused.
- Architecting for HIPAA Security and Compliance Whitepaper – This whitepaper describes how companies can use AWS to create HIPAA-compliant applications.

  Note
  Not all services are compliant with HIPAA.
- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- Evaluating Resources with Rules in the *AWS Config Developer Guide* – The AWS Config service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- AWS Security Hub – This AWS service provides a comprehensive view of your security state within AWS that helps you check your compliance with security industry standards and best practices.
- AWS Audit Manager – This AWS service helps you continuously audit your AWS usage to simplify how you manage risk and compliance with regulations and industry standards.

Resilience in AWS Systems Manager Incident Manager

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

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

Incident Manager is a global-regional service and does not currently support Availability Zones.

In addition to the AWS global infrastructure, Incident Manager offers several features to help support your data resiliency and backup needs. During the Getting started wizard you're asked to set up a replication set. This regional replication set ensures that your data and resources are accessible from multiple Regions, making incident management across a cloud-network more manageable. This replication also ensures that your data is safe and accessible in the event that one of your Regions goes down. To learn more about the replication set, see Get prepared wizard (p. 6).

Infrastructure security in AWS Systems Manager Incident Manager

As a managed service, AWS Systems Manager Incident Manager 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 Incident Manager through the network. Clients must support Transport Layer Security (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's associated with an IAM principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.

Configuration and vulnerability analysis in Incident Manager

Configuration and IT controls are a shared responsibility between AWS and you, our customer. For more information, see the AWS shared responsibility model.

Security best practices in AWS Systems Manager Incident Manager

AWS Systems Manager Incident Manager provides many security features to consider as you develop and implement your own security policies. The following best practices are general guidelines and don't represent a complete security solution. Because these best practices might not be appropriate or sufficient for your environment, treat them as helpful considerations rather than prescriptions.

Topics

Incident Manager preventative security best practices

Implement least privilege access

When granting permissions, you decide who is getting what permissions to which Incident Manager resources. You enable specific actions that you want to allow on those resources. Therefore, grant only the permissions that are required to perform a task. Implementing least privilege access is fundamental in reducing security risk and the impact that could result from errors or malicious intent.

The following tools are available to implement least privilege access:

- IAM user policies and Permissions Boundaries for IAM Entities
- Service Control Policies

Creating and managing contacts

When activating contacts, Incident Manager reaches out to the device to confirm the activation. Ensure the device information is correct before activating the device. This reduces the possibility that Incident Manager contacts the wrong device or person during activation.

Regularly review your contacts and escalation plans to ensure that only contacts that need to be contacted during an incident are being contacted. Regularly review the contacts to remove outdated or

incorrect information. If a contact should no longer be informed when an incident occurs, remove them from the related escalation plans or remove them from Incident Manager.

Make chat channels private

You can make your incident chat channels private to implement least privilege access. Consider using a different chat channel with a scoped down user list for each response plan template. This ensures only the correct responders are pulled into a chat channel that may contain sensitive information.

AWS Chatbot enabled Slack channels inherit the permissions of the IAM role used to configure AWS Chatbot. This enables responders in an AWS Chatbot enabled Slack channel to call any allow-listed action, such as Incident Manager APIs and retrieving metrics graphs.

Keep AWS tools up to date

AWS regularly releases updated versions of tools and plugins that you can use in your AWS operations. Keeping these resources up to date ensures that users and instances in your account have access to the latest functionality and security features in these tools.

- AWS CLI – The AWS Command Line Interface (AWS CLI) is an open source tool that enables you to interact with AWS services using commands in your command-line shell. To update the AWS CLI, you run the same command used to install the AWS CLI. We recommend creating a scheduled task on your local machine to run the command appropriate to your operating system at least once every two weeks. For information about installation commands, see Installing the AWS Command Line Interface in the *AWS Command Line Interface User Guide*.
- AWS Tools for Windows PowerShell – The Tools for Windows PowerShell are a set of PowerShell modules that are built on the functionality exposed by the AWS SDK for .NET. The Tools for Windows PowerShell enable you to script operations on your AWS resources from the PowerShell command line. Periodically, as updated versions of the Tools for Windows PowerShell are released, you should update the version that you're running locally. For information, see Updating the AWS Tools for Windows PowerShell on Windows or Updating the AWS Tools for Windows PowerShell on Linux or macOS.

Related content

Security best practices for Systems Manager

Incident Manager detective security best practices

Identify and audit all your Incident Manager resources

Identification of your IT assets is a crucial aspect of governance and security. Identify your Systems Manager resources to assess their security posture and take action on potential areas of weakness. Create resource groups for your Incident Manager resources. For more information, see What Is AWS Resource Groups?

Use AWS CloudTrail

AWS CloudTrail provides a record of actions taken by a user, role, or an AWS service in Incident Manager. Using the information collected by AWS CloudTrail, you can determine the request that was made to Incident Manager, the IP address from which the request was made, who made the request, when it was made, and additional details. For more information, see Logging AWS Systems Manager Incident Manager API calls using AWS CloudTrail (p. 65).

Monitor AWS security advisories

Regularly check security advisories posted in Trusted Advisor for your AWS account. You can do this programmatically using describe-trusted-advisor-checks.

Further, actively monitor the primary email address registered to each of your AWS accounts. AWS will contact you, using this email address, about emerging security issues that might affect you.

AWS operational issues with broad impact are posted on the AWS Service Health Dashboard. Operational issues are also posted to individual accounts through the AWS Personal Health Dashboard. For more information, see the AWS Health documentation.

Related content

Amazon Web Services: Overview of Security Processes (whitepaper)

Getting Started: Follow Security Best Practices as You Configure Your AWS Resources (AWS Security Blog)

IAM Best Practices

Security Best Practices in AWS CloudTrail

AWS glossary

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

Document History for Incident Manager

update-history-change	update-history-description	update-history-date
Console engagement acknowledgement (p. 73)	You can now acknowledge engagements directly from the Incident Manager console.	August 5, 2021
Properties tab (p. 73)	Incident Manager introduced a properties tab to the incident details page, providing more information about the incidents, the parent OpsItem, and the related post-incident analysis.	August 3, 2021
Incident Manager Launch (p. 73)	Incident Manager is an incident management console designed to help users mitigate and recover from incidents affecting their AWS hosted applications.	May 10, 2021