User Guide

AWS Support

API Version 2025-12-23

AWS Support: User Guide

This documentation is a draft for private preview for regions in the AWS European Sovereign Cloud. Documentation content will continue to evolve. Published: December 30, 2025.

Table of Contents

Getting started with AWS Support

AWS Support offers a range of plans that provide access to tools and expertise that support the success and operational health of your AWS solutions. All support plans provide round-the-clock access to customer service, AWS documentation, technical papers, and support forums. For technical support and more resources to plan, deploy, and improve your AWS environment, you can choose a support plan for your AWS use case.

Notes

- To create a Support interaction to use AI-generated troubleshooting to resolve your issue, and optionally create a support case, see AI-enhanced troubleshooting in the Support Center Console.
- For more information about the different AWS Support plans, see Compare AWS Support plans and Change AWS Support Plans.
- Support plans offer different response times for your support cases. See Choosing an initial support case severity level and Understanding AWS Support response times.

Topics

AI-enhanced troubleshooting in the Support Center Console

AI-enhanced troubleshooting capabilities that help you resolve issues faster and more efficiently are available in supported AWS Regions. If you have an AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, then you can use the capabilities in the Support Center Console to troubleshoot technical issues as well as account and billing issues. If you

have a Basic Support plan, you can use the capabilities in the Support Center Console for troubleshooting general questions and for assistance with account and billing issues. Using AI-enhanced troubleshooting streamlines the support experience by using contextual awareness and automated diagnostics to provide targeted solutions for your AWS environment.

AI-enhanced troubleshooting in the Support Center Console is supported in the following AWS Regions:

- US East (N. Virginia) Region
- US East (Ohio) Region
- Europe (Ireland) Region

> **Note**
>
> If you operate in an AWS Region that doesn't support AI-enhanced capabilities in the Support Center Console then you will use the legacy method of case management. For more information, see Legacy experience: Creating support cases and case management.

When you access the Support Center Console, you can enter your issue description in natural language, import relevant Amazon Q conversations, receive generative AI troubleshooting guidance, and choose to create a support case with pre-populated fields, if needed.

You can provide contextual information about your environment and issue to receive personalized solutions throughout the troubleshooting process.

AI-enhanced troubleshooting in the AWS Support console provides the following key benefits:

- **Faster issue resolution:** Get immediate responses and relevant solutions as soon as you describe your problem.
- **Context preservation:** Import your previous Amazon Q conversations to maintain troubleshooting context.
- **Streamlined case creation:** Use natural language to describe issues instead of navigating multiple form fields.
- **Intelligent follow-up:** Receive relevant follow-up questions based on your specific AWS environment.

For a complete list of the capabilities available in your Support plan, see Compare AWS Support Plans.

Notes

- To change your support plan, see Change AWS Support Plans.
- To close your account, see Closing an account in the *AWS Billing User Guide*.
- To find common troubleshooting topics for AWS services, see Troubleshooting resources.
- If you're a customer of an AWS Partner that is part of the AWS Partner Network, and you use Resold Support, contact your AWS Partner directly for any billing related issues. AWS Support can't assist with non-technical issues for Resold Support, such as billing and account management. For more information, see the following topics:
  - How AWS Partners can determine AWS Support plans in an organization
  - AWS Partner-Led Support

Topics

Set up permissions to use AI-enhanced troubleshooting

To access AI-enhanced troubleshooting capabilities in Support Center, you need specific AWS Identity and Access Management permissions. This section describes the necessary IAM permissions and explains how to configure them so that you can fully use these capabilities.

AI-enhanced troubleshooting requires permissions beyond traditional support case management. The required permissions fall into three categories:

- **Support interaction permissions:** Enable the new interaction-based workflow in Support Center.
- **AI-powered classification permissions:** Allow access to AI-powered issue classification features.

- **Amazon Q integration permissions:** Enable conversation import from Amazon Q Developer.

These permissions supplement your existing AWS Support permissions and don't replace them.

You can set up permissions for AI-enhanced troubleshooting in two ways:

Option 1: Use the AWS managed policy (recommended). Attach the `AWSSupportAccess` managed policy to your users or roles. This policy includes all required permissions and is automatically updated when new Support features are released.

Option 2: Create a custom policy with minimum required permissions. This approach gives you more control but requires manual updates when new features are added.

Option 1: Use the AWS managed policy (recommended)

If you currently have the AWSSupportAccess managed policy attached, no additional permissions are required. However, to continue to use the functions included in the Support Center Console API, you must add the Support Center Console operations to your IAM policies before June 1, 2026, if you don't already have them. To do this, update the AWS Support managed policy to include the `support-console:*` actions. For more information, see Adding IAM policies for the Support Center Console API operations.

Option 2: Create a custom policy with minimum required permissions

You can explicitly allow-list specific actions instead of using wildcards. The following are the required permissions for support interactions, case creation, and case management:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "support:AddAttachmentsToSet",
        "support:AddCommunicationToCase",
        "support:CreateCase",
        "support:DescribeAttachment",
        "support:DescribeCaseAttributes",
```

```
        "support:DescribeCases",
        "support:DescribeCommunication",
        "support:DescribeCommunications",
        "support:DescribeCreateCaseOptions",
        "support:DescribeIssueTypes",
        "support:DescribeServices",
        "support:DescribeSeverityLevels",
        "support:DescribeSupportedLanguages",
        "support:DescribeSupportLevel",
        "support:GetInteraction",
        "support:InitiateCallForCase",
        "support:ListInteractionEntries",
        "support:ListInteractions",
        "support:InitiateChatForCase",
        "support:PutCaseAttributes",
        "support:ResolveCase",
        "support:ResolveInteraction",
        "support:SearchForCases",
        "support:StartInteraction",
        "support:UpdateInteraction",
        "support-console:GetAccountState",
        "support-console:GetAccountGovCloudEnabled",
        "support-console:GetCaseDraft",
        "support-console:CreateCaseDraft",
        "support-console:DeleteCaseDraft",
        "support-console:GetBanner",
        "support-console:DescribeDynamicHelp",
        "support-console:CreateContact",
      ],
      "Resource": "*"
    }
  ]
}
```

Note

Using a custom policy requires ongoing maintenance as AWS Support releases new features. For more information about the Support Center Console API operations, see Adding IAM policies for the Support Center Console API operations. For more information about each of the Support API operations, see Manage access to AWS Support Center.

Required permissions for Amazon Q integration

To use the Amazon Q conversation import feature in Support Center, IAM identities need permissions for the following Amazon Q Developer actions:

- `q:StartConversation`: Start a new conversation with Amazon Q.
- `q:SendMessage`: Send messages within a conversation.
- `q:GetConversation`: Retrieve conversation details. This action is required for console access.
- `q:ListConversations`: List available conversations. This action is required for console access and Support Center integration.

Amazon Q integration with Support Center Console specifically requires the `q:ListConversations` permission to display your recent conversations for import. For detailed guidance on configuring Amazon Q Developer permissions, see Amazon Q Developer permissions reference and Manage access to Amazon Q Developer with policies.

Applying required permissions for support interactions

To apply permissions to your IAM users, complete the following steps:

1. Sign in to the AWS Management Console and open the IAM console at https://eusc-de-east-1.console.amazonaws-eusc.eu/iam/.
2. In the navigation pane, choose **Policies**, then choose **Create policy**.
3. Choose the **JSON** tab and paste one of the policy documents mentioned in the previous sections.
4. Choose **Next: Tags**, then **Next: Review**.
5. Enter a policy name such as `SupportConsoleInteractionsAccess` and provide a description that explains the policy's purpose.
6. Choose **Create policy**.
7. Attach the policy to your IAM users, groups, or roles that need access to the Support Center.

If you have existing AWSSupportAccess managed policy attachments, then attach the supplementary custom policy alongside the managed policy.

Create a support interaction

A support interaction is how you begin your engagement with AWS Support. You first describe your issue using natural language and receive assistance tailored to you using AI-enhanced troubleshooting. Your initial interaction might include clarifying questions, contextual solutions, and automated problem resolution, without creating a support case. These interactions might resolve issues independently or serve as the foundation for a support case looping in a human engineer, if needed.

Support interactions differ from support cases in that support cases include engagement with a Cloud Support Engineer. You can choose to automatically generate a support case based on a prior support interaction. The support case maintains all context from the initial support interaction and includes additional AI-generated insights to assist the Cloud Support Engineer with resolving your issue. This powerful combination of AI-enhanced troubleshooting and assistance from AWS Cloud Support Engineers potentially lead to faster issue resolution and reduced down time.

Notes

- You can revert to the legacy method of case management by choosing **Use the old experience** in the banner at the top of the Support Center Console. For more information, see Legacy experience: Creating support cases and case management.
- You can sign in to the Support Center Console as an AWS Identity and Access Management (IAM) user. For more information, see Manage access to AWS Support Center.
- If you can't sign in to the Support Center Console and create a support case, you can use the Contact Us page instead. You can use this page to get help with billing and account issues.

To start a support interaction, complete the following steps:

1. Sign in to the AWS Support Center Console.

> **Tip**
>
> In the AWS Management Console, you can also choose the question mark icon () and then choose **Support Center**.

2. You have several options for starting your support interaction:

   - Enter details about the issues that you need assistance with. This is how you begin a new support interaction. Enter detailed information about your issue and any troubleshooting steps that you have already taken.
   - **Continue an existing support interaction:** Choose from a recent support interaction shown in the **Describe your issue or continue with** section. This section shows the two most recent support interactions. Access the **Viewing past support interactions** section to see additional past support interactions.
   - **Use a Amazon Q transcript:** Select the Amazon Q icon in the text field to see a list of recent Amazon Q conversations. The five most recent conversations from the AWS GovCloud (US-East) AWS Region are shown. Or, choose from a recent Amazon Q interaction shown in the **Describe your issue or continue with**. When you select a conversation, a summary of that conversation is generated and added to the text box. If you select an Amazon Q conversation, then you see a disclaimer regarding AWS Region and user accessibility.
3. Choose the **Send** icon in the bottom right of the text field.
4. AWS Support generative AI-powered troubleshooting analyzes your query along with your specific AWS environment. You might be prompted to provide additional information to assist with the analysis. If you see a prompt for additional information, enter the requested data, then choose **Submit**. If you don't know or don't have access to the requested information, then you can skip this step and receive a general guidance response instead. Keep in mind that a general guidance response isn't specific to your AWS environment.

   When the analysis is complete, you see a summary of the findings, along with remediation steps. To view the sources used in the analysis and remediation steps, choose **Sources**.
5. If you need further assistance, you can complete one of the following options:

   - **Continue with AI-assisted support:** To further refine the AI-assisted analysis and generate a new response, choose **Add more details for a better response**. Enter information in the **Additional details** field, and then choose **Submit**. Keep in mind that this option is for

additional context for the original issue. If you need to enter context for a new issue, select **Start new interaction** at the top or bottom of the screen.

- **Create a support case:** To create a support case with AWS Support, choose **Create a case**. This option starts the case creation workflow. Many of the case details are auto-populated for you based on your support interaction. You can change this information as needed. Your support interaction, including details of any resolution steps provided, are added to the support case. For details on how to create a support case, see Create a support case from a support interaction.

At any time throughout the support interaction, you can use the **Thumbs up** and **Thumbs down** icons to provide feedback on your experience.

Create a support case from a support interaction

When you select **Create case** during your support interaction, a support case is created for you with many of the case details, such as the **Subject**, **Description**, **Case type Service**, **Category**, and **Severity level** populated for you. You can change this information as needed. Make sure that you review this information for accuracy.

After you choose **Create case**, enter or verify the following information:

1. Verify the **Subject** for this support case. The **Subject** is a brief synopsis of what your support interaction is about.
2. Verify the **Description**. Your initial inquiry appears in the **Description** field. Modify this information as needed. Make your description as detailed as possible. Include relevant resource information, along with anything else that might help us understand your issue.
3. (Optional) Choose **Attach files** to add any relevant files to your case, such as error logs or screenshots. You can attach up to three files. Each file can be up to 5 MB.
4. For **Case type**, choose one of the following options:

   - **Account and billing**
   - **Technical**
   - **Service quotas**. You can only request certain types of service quota increases from the Support Center Console. For more information, see Request a service quota increase.

> **Note**
>
> If you have a Basic Support plan, then you can't create a **Technical** support case.

5. Verify the **Service**, **Category**, and **Severity**.
6. In the **Communication preference** section, indicate how you want AWS to communicate with you. You can choose one of the following options:

   a. **Email:** Receive a response to your email.

   b. **Phone:** Receive a phone call from a support agent. If you choose this option, enter the following information:

      - **Country or region**
      - **Phone number**
      - **(Optional) Extension**

   c. **Chat:** Start a live chat with a support agent. If you can't connect to a chat, see Troubleshooting.

   (Optional) If you have an AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, you can enter up to 10 additional email addresses in the **Additional contacts**. You can enter the email addresses of people to notify when the status of the case changes. If you're signed in as an IAM user, include your email address. If you're signed in with your root account email address and password, you don't need to include your email address.

   > **Note**
   >
   > If you have the Basic Support plan, the **Additional contacts** option isn't available. However, the **Operations contact** specified in the **Alternate Contacts** section of the **My Account** page receives copies of the case correspondence, but only for the specific case types of account and billing.

7. When you're ready to submit the support case, select **Submit**. You are directed to the **Case details** page where you can see your case details, the support interaction, and the case correspondences.

Select **Case details** to view the information about your case, such as attachments, or severity level. Select **Support interactions** to see the support interactions associated with this case.

View support interactions

Past interactions with AWS Support are saved for 10 years. You can view past interactions from your Support Dashboard by selecting the **List** icon. Then, choose the interaction that you want to view. The AWS Support interaction details appear. If you choose to create a support case from an interaction, then the interaction no longer appears in your past interactions list. The interaction now displays in the **Case details** page of the associated support case.

You can add additional details to the interaction to generate a new response. Or, you can choose to create a Support case from the interaction by choosing **Create a case** on the AWS Support interaction details screen.

Troubleshooting

If you have difficulty when you create or manage your support case, see the following troubleshooting information.

I want to reopen a live chat for my case

You can reply to your existing support case to open another chat window. For more information, see Updating an existing support case.

I can't connect to a live chat

If you chose the **Chat** option but you can't connect to the chat window, first perform the following checks:

- Ensure that you've configured your browser to allow pop-up windows in Support Center.

  > **Note**
  >
  > Review the settings for your browser. For more information, see the Chrome Help and Firefox Support websites.

- Ensure that you've configured your network so that you can use AWS Support:
  - Your firewall supports web socket connections.

If you still can't connect to the chat window, contact AWS Support using email or phone contact options.

Virtual meetings with AWS Support

Virtual meetings enable you to connect with AWS Support engineers through video calls with screen sharing capabilities. This feature helps you resolve complex technical issues that require visual demonstration or real-time collaboration.

When a support engineer determines that your case requires visual assistance, they can initiate a virtual meeting. You receive a meeting invitation on your case details page in the AWS Support Center. After you accept the invitation, you join a secure video call where you can share your screen and collaborate with the support engineer.

Virtual meetings integrate with Amazon Connect and use WebRTC technology to provide secure, browser-based video conferencing without requiring additional software installation.

Virtual meetings are available in the commercial AWS Regions only.

Topics

Join a virtual meeting

Virtual meetings are initiated by the AWS Support engineer assisting you with your support case. To join a virtual meeting, complete the following steps.

1. When a support engineer initiates a virtual meeting, you see a meeting invitation on the **Case details** page in your support case. To join the meeting, choose **Join virtual meeting** to accept the invitation.

   Meeting invitations expire after 10 minutes. If you don't join within this time, request a new meeting from your support engineer.
2. Grant browser permissions for camera and microphone access when prompted.

The virtual meeting opens in a new window and you're connected to the support engineer. You can mute and unmute your microphone, or disconnect from the meeting, useing the buttons at the bottom of the screen.

Security and privacy during your virtual meeting

Virtual meetings use the same authentication and authorization mechanisms as other AWS Support operations. The following security measures protect your meetings:

- **Case ownership validation:** You can only join meetings for cases that belong to your AWS account.
- **AWS Identity and Access Management (IAM) based access control:** You must have the appropriate IAM permissions to join virtual meetings.
- **Encrypted connections:** All meeting data is transmitted over encrypted WebRTC connections.
- **Audit logging:** All meeting activities are logged in AWS CloudTrail for compliance and auditing purposes.

> **Important**
>
> Virtual meetings are recorded for quality assurance and training purposes. Don't share sensitive information such as passwords or access keys during the meeting.

Required IAM permissions for virtual meetings

To join virtual meetings, your IAM user or role must have the following permission:

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "support:InitiateLiveContactForCase"
        ],
        "Resource": "*"
    }]
}
```

For more information about AWS Support permissions, see Manage access to AWS Support Center.

Troubleshooting virtual meetings

I can't see the meeting invitation.

Verify that your support engineer has initiated the meeting. Refresh the case details page. If the invitation still doesn't appear, contact your support engineer through the case correspondence.

The meeting invitation expired.

Meeting invitations expire after 10 minutes for security reasons. Request a new meeting invitation from your support engineer.

I'm experiencing connection issues.

Check your internet connection. Ensure that your firewall or network security settings allow WebRTC traffic. Try using a different browser or network connection.

I receive an authorization error.

Verify that your IAM user or role has permissions for the `support:InitiateLiveContactForCase` action.

Case management

> **Note**
>
> You can revert to the legacy method of case management by choosing **Use the old experience** in the banner at the top of the Support Center console. For more information, see Legacy experience: Creating support cases and case management.

In the AWS Management Console, you can create three types of customer cases in Support:

- **Account and billing** support cases are available to all AWS customers. You can get help with billing and account questions.
- **Service limit increase** requests are available to all AWS customers. For more information about the default service quotas, formerly referred to as limits, see AWS service quotas in the *AWS General Reference*.

- **Technical** support cases connect you to technical support for help with service-related technical issues and, in some cases, third-party applications. If you have Basic Support, you can't create a technical support case.

> **Notes**
>
> - To change your support plan, see Change AWS Support Plans.
> - To close your account, see Closing an Account in the *AWS Billing User Guide*.
> - To find common troubleshooting topics for AWS services, see Troubleshooting resources.
> - If you're a customer of an AWS Partner that is part of the AWS Partner Network, and you use Resold Support, contact your AWS Partner directly for any billing related issues. AWS Support can't assist with non-technical issues for Resold Support, such as billing and account management. For more information, see the following topics:
>   - How AWS Partners can determine AWS Support plans in an organization
>   - AWS Partner-Led Support

Describing your problem

Make your description as detailed as possible. Include relevant resource information, along with anything else that might help us understand your issue. For example, to troubleshoot performance, include timestamps and logs. For feature requests or general guidance questions, include a description of your environment and purpose.

When you provide as much detail as possible, you increase the chances that your case can be resolved quickly.

Choosing an initial support case severity level

You might want to create a support case at the highest severity that your support plan allows. However, it's a best practice to choose the highest severity only for cases that can't be worked around or that directly affect production applications. For information about building your services so that losing a single resource doesn't affect your applications, see the Building Fault-Tolerant Applications on AWS technical paper.

The following table lists the severity levels, response times, and example problems.

Notes

- If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, you can reassign your support case severity level to reflect changes to urgency and business impact. For example, you can change your support case from **System impaired** to **Production system impaired**. When you change the case severity, AWS Support receives notification and routes the case according to the new severity level. For more information, see Changing the severity level of your support case.
- If you don't have Basic Support plan, then you can't change the severity level for a support case after you create it. If your situation changes, work with the Support agent.
- For more information about the severity level, see the AWS Support API Reference.

Severity	Severity level code	First-response time	Description and support plan
General guidance	`low`	24 hours	You have a general development question, or you want to request a feature. (AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan)
System impaired	`normal`	12 hours	Non-critical functions of your application are behaving abnormally, or you have a time-sensitive development question. (AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan)
Production system impaired	`high`	4 hours	Important functions of your application are impaired or degraded. (AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan)
Production system down	`urgent`	1 hour	Your business is significantly impacted. Important functions of your application aren't available. (AWS Business Support+, AWS

Severity	Severity level code	First-response time	Description and support plan
			Enterprise Support, or AWS Unified Operations plan)
Business-critical system down	`critical`	• AWS Business Support+: Less than 30 minutes • AWS Enterprise Support: Less than 15 minutes • AWS Unified Operations: 5 minutes from an Incident Management Engineer	Your business is at risk. Critical functions of your application aren't available.

Understanding AWS Support response times

AWS Support makes every reasonable effort to respond to your initial request within the indicated timeframe. For information about the scope of support for each Support plan, see AWS Support features.

If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, you have round-the-clock access for technical support.

Note

If you choose Japanese as your preferred contact language for support cases, support in Japanese may be available as follows:

- If you need customer service for non-technical support cases, or if you have a Developer Support plan and need technical support, support in Japanese is available during business hours in Japan defined as 09:00 AM to 06:00 PM Japan Standard Time (GMT+9), excluding holidays and weekends.
- If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, technical support is available all day, every day in Japanese.

If you choose Chinese as your preferred contact language for support cases, support in Chinese might be available as follows:

- If you need customer service for non-technical support cases, support in Chinese is available 09:00 AM to 06:00 PM (GMT+8), excluding holidays and weekends.
- If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, technical support is available all day, every day in Chinese.

If you choose Korean as your preferred contact language for support cases, support in Korean may be available as follows:

- If you need customer service for non-technical support cases, support in Korean is available during business hours in Korea defined as 09:00 AM to 06:00 PM Korean Standard Time (GMT+9), excluding holidays and weekends.
- If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, technical support is available all day, every day in Korean.

Changing the severity level of your support case

If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, you can reassign your support case severity level to reflect changes to urgency and business impact. For example, you can change your support case from **System impaired** to **Production system impaired**. When you change the case severity, AWS Support receives notification and attends to the case according to the new severity level.

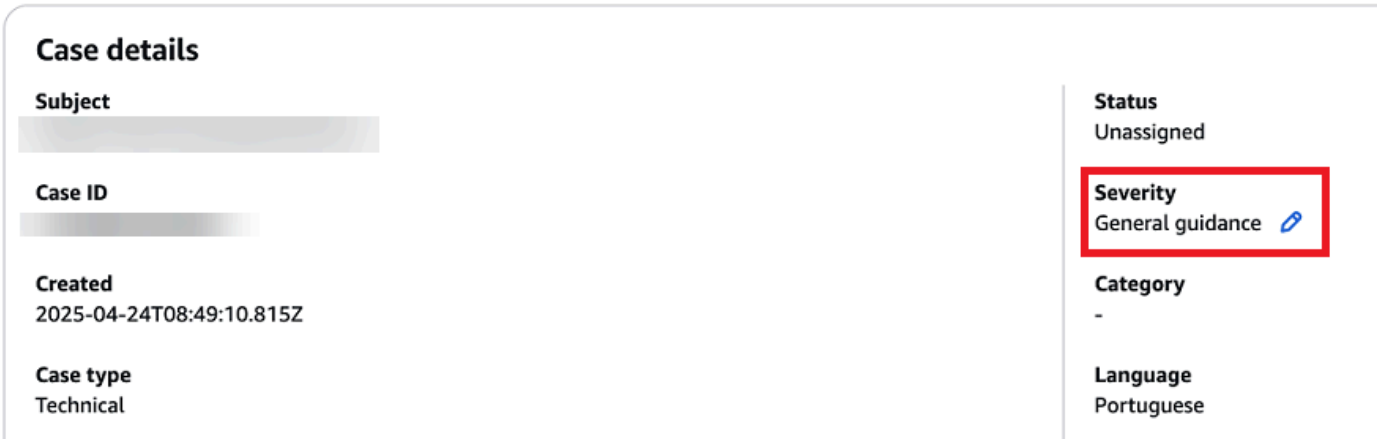
> **Note**
>
> Japanese (JP) account or billing, Service Quota Increase Request (SQIR), and Turkish (TR) account or billing cases created in these languages have a default severity and can't be changed.

To change the severity of a support case, complete the following steps:

1. Sign in to the AWS Support Center Console.

   > **Tip**
   >
   > In the AWS Management Console, you can also choose the question mark icon ( ) and then choose **Support Center**.

2. Select the case that you want to change the severity level for.
3. In **Case details**, choose the pencil icon next to the **Severity** field, as shown in the following example.

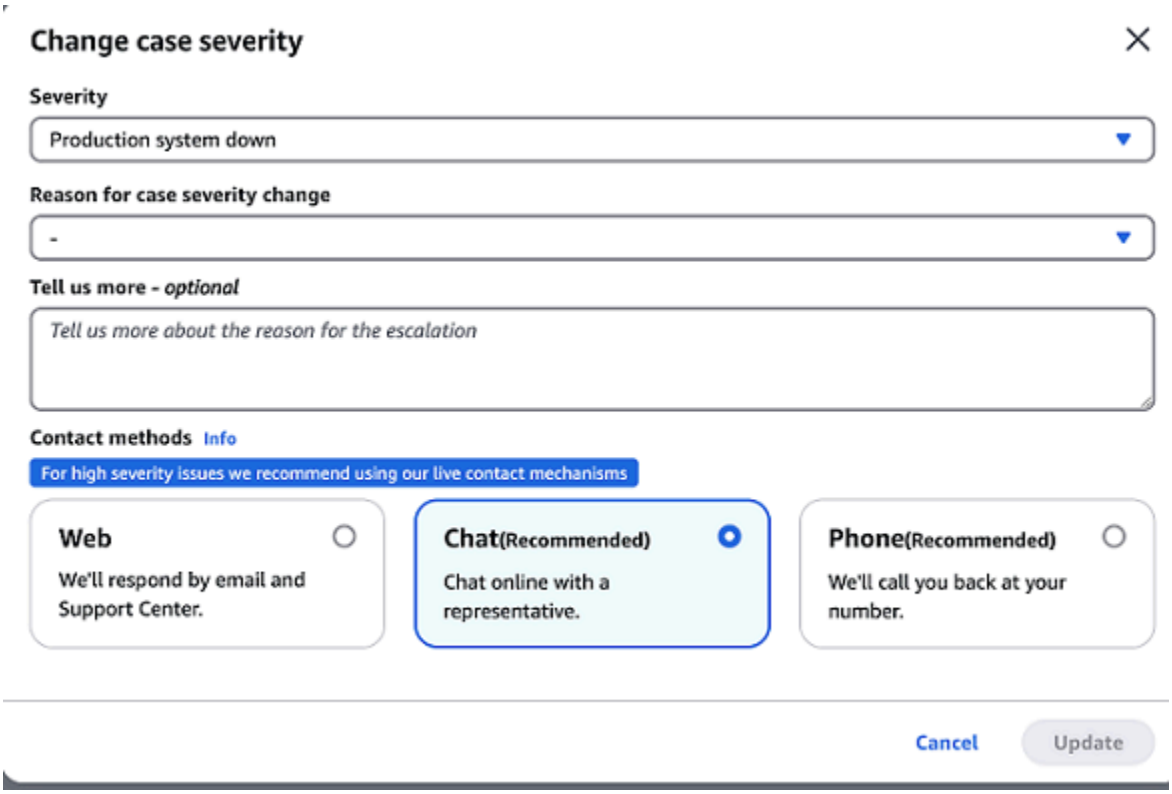


4. For **Severity**, choose the new severity level from the following options:

   - General guidance
   - System impaired
   - Production system impaired
   - Production system down
   - Business-critical system down

5. For **Reason for case severity change**, choose from the available options for why you're changing the case severity.
6. (Optional) For **Tell us more**, enter additional information about this change.
7. Do one of the following:

   - If you're lowering the support case severity, or if you're raising it from **General guidance** to **System impaired** or **Production system impaired**, choose **Update**.
   - If you're raising the severity to **Production system down** or **Business-critical system down**, use one of the options in the **Contact methods** section to engage with AWS Support, and then choose **Update**. The following example shows the options available in the **Contact methods** section.



Note

- If you upgrade your support case severity to **Production system down** or **Business-critical system down**, you must wait 60 minutes before you can change the severity again.
- If your support case is currently set to **Business-critical system down**, you're prompted to initiate live contact with AWS Support instead of assigning a higher severity.
- If you're raising your support case severity level after already raising it at least once, you might encounter a waiting period. For example, if you change the severity from **System impaired** to **Production system impaired** at 6:00 AM, then your support case falls under the 4-hour first-response time for the **Production system impaired** severity level. In this scenario, you can upgrade the severity level again at 10:00 AM, after the 4-hour window. For a list of first-response times for each severity level, see the table in Understanding AWS Support response times.

Request a service quota increase

You can request increases to your service quotas (formerly referred to as limits) to support your workload requirements.

Use the Service Quotas service to request increases directly for your services. For more information, see the following documentation:

- What is Service Quotas? in the *Service Quotas User Guide*
- Requesting a quota increase in the *Service Quotas User Guide*

At this time, Service Quotas doesn't support service quotas for all AWS services in all AWS Regions. If your AWS service or AWS Region isn't available in the Service Quotas console, complete the following steps to create a support case to request the quota increase:

1. Sign in to the AWS Support Center Console.

> **Tip**
>
> In the AWS Management Console, you can also choose the question mark icon () and then choose **Support Center**.

2. On the **Support interactions** page, enter details about this service limit increase. When prompted, choose **Create a case**. Many of the Support case fields will be pre-populated with the text that you entered during your interaction. You can edit these fields as needed. For additional details on creating a support interaction, see Create a support interaction.
3. For **Case type**, select **Service quotas**.
4. For **Service**, select **Service Limit increase**.
5. For **Category**, select the type of increase that you're requesting from the list. Only service limit increase requests available in Support Center are listed here. For other types of service limit requests, see Requesting a quota increase in the *Service Quotas User Guide*.
6. (Optional) From the **Preferred contact language** drop down, select the language that you want AWS Support to use when corresponding with you.
7. For **Region**, select the AWS Region where you're requesting the increase.

> **Note**
>
> AWS Region selection isn't available if you selected **General** as the **Category**.

8. (Optional) To request multiple limit increases, choose **Add another limit** and then choose another AWS Region.
9. Enter a **Description** for this service quota increase or multiple increases. You can attach files, if necessary.
10. Choose **Next step: Solve now or contact us**.
11. For **Contact options**, choose one of the following options:

    - **Web** – Receive a reply in Support Center.
    - **Chat** – Start a live chat with a support agent. If you can't connect to a chat, see Troubleshooting.

- **Phone** – Receive a phone call from a support agent. If you choose this option, enter the following information:
  - **Country/Region**
  - **Phone number**
  - **(Optional) Extension**

12. When you're ready to submit the support case, select **Submit**. You are directed to the **Case details** page where you can see your case details, the support interaction, and the case correspondences.

    Select **Case details** to view the information about your case, such as attachments, or severity level. Select **Support interactions** to see the support interactions associated with this case.

Legacy experience: Creating support cases and case management

> **Important**
>
> You can revert to the legacy method of case management by choosing **Use the old experience** in the banner at the top of the Support Center console.

In the AWS Management Console, you can create three types of customer cases in Support:

- **Account and billing** support cases are available to all AWS customers. You can get help with billing and account questions.
- **Service limit increase** requests are available to all AWS customers. For more information about the default service quotas, formerly referred to as limits, see AWS service quotas in the *AWS General Reference*.
- **Technical** support cases connect you to technical support for help with service-related technical issues and, in some cases, third-party applications. If you have Basic Support, you can't create a technical support case.

> **Notes**
>
> - To change your support plan, see Change AWS Support Plans.

- To close your account, see Closing an Account in the *AWS Billing User Guide*.
- To find common troubleshooting topics for AWS services, see Troubleshooting resources.
- If you're a customer of an AWS Partner that is part of the AWS Partner Network, and you use Resold Support, contact your AWS Partner directly for any billing related issues. AWS Support can't assist with non-technical issues for Resold Support, such as billing and account management. For more information, see the following topics:
  - How AWS Partners can determine AWS Support plans in an organization
  - AWS Partner-Led Support

Creating a support case

You can create a support case in the Support Center of the AWS Management Console.

Notes

- You can sign in to Support Center as an AWS Identity and Access Management (IAM) user. For more information, see Manage access to AWS Support Center.
- If you can't sign in to Support Center and create a support case, you can use the Contact Us page instead. You can use this page to get help with billing and account issues.

To create a support case

1. Sign in to the AWS Support Center Console.

   Tip

   In the AWS Management Console, you can also choose the question mark icon () and then choose **Support Center**.

2. Choose **Create case**.
3. Choose one of the following options:

- **Account and billing**
- **Technical**
- For service quota increases, choose **Looking for service quota increases?** and then follow the instructions for Request a service quota increase.

4. Choose the **Service**, **Category**, and **Severity**.

> **Tip**
>
> You can use the recommended solutions that appear for commonly asked questions.

5. Choose **Next step: Additional information**
6. On the **Additional information** page, for **Subject**, enter a title about your issue.
7. For **Description**, follow the prompts to describe your case, such as the following:

   - Error messages that you received
   - Troubleshooting steps that you followed
   - How you're accessing the service:
     - AWS Management Console
     - AWS Command Line Interface (AWS CLI)
     - API operations

8. (Optional) Choose **Attach files** to add any relevant files to your case, such as error logs or screenshots. You can attach up to three files. Each file can be up to 5 MB.
9. Choose **Next step: Solve now or contact us**.
10. On the **Contact us** page, choose your preferred language.
11. Choose your preferred contact method. You can choose one of the following options:

    a. **Web** – Receive a reply in Support Center.

    b. **Chat** – Start a live chat with a support agent. If you can't connect to a chat, see Troubleshooting.

    c. **Phone** – Receive a phone call from a support agent. If you choose this option, enter the following information:

       - **Country or region**
       - **Phone number**

- **(Optional) Extension**

> **Notes**
>
> - The contact options that appear depend on the type of case and your support plan.
> - You can choose **Discard draft** to clear your support case draft.

12. (Optional) If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, the **Additional contacts** option appears. You can enter the email addresses of people to notify when the status of the case changes. If you're signed in as an IAM user, include your email address. If you're signed in with your root account email address and password, you don't need to include your email address

> **Note**
>
> If you have the Basic Support plan, the **Additional contacts** option isn't available. However, the **Operations** contact specified in the **Alternate Contacts** section of the My Account page receives copies of the case correspondence, but only for the specific case types of account and billing, and technical.

13. Review your case details and then choose **Submit**. Your case ID number and summary appear.

Describing your problem

Make your description as detailed as possible. Include relevant resource information, along with anything else that might help us understand your issue. For example, to troubleshoot performance, include timestamps and logs. For feature requests or general guidance questions, include a description of your environment and purpose. In all cases, follow the **Description Guidance** that appears on your case submission form.

When you provide as much detail as possible, you increase the chances that your case can be resolved quickly.

Choosing an initial support case severity level

You might want to create a support case at the highest severity that your support plan allows. However, it's a best practice to choose the highest severity only for cases that can't be worked

around or that directly affect production applications. For information about building your services so that losing a single resource doesn't affect your applications, see the Building Fault-Tolerant Applications on AWS technical paper.

The following table lists the severity levels, response times, and example problems.

Notes

- If you have Enterprise Support or an Enterprise On-Ramp plan, you can reassign your support case severity level to reflect changes to urgency and business impact. For example, you can change your support case from **System impaired** to **Production system impaired**. When you change the case severity, AWS Support receives notification and routes the case according to the new severity level. For more information, see Changing the severity level of your support case.
- If you don't have Enterprise support or an Enterprise On-Ramp plan, then you can't change the severity level for a support case after you create it. If your situation changes, work with the Support agent for your support case.
- For more information about the severity level, see the AWS Support API Reference.

Severity	Severity level code	First-response time	Description and support plan
General guidance	`low`	24 hours	You have a general development question, or you want to request a feature. (*Developer, AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan)
System impaired	`normal`	12 hours	Non-critical functions of your application are behaving abnormally, or you have a time-sensitive development question. (*Developer, AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan)
Production system impaired	`high`	4 hours	Important functions of your application are impaired or degraded. (AWS Business Support

Severity	Severity level code	First-response time	Description and support plan
			+, AWS Enterprise Support, or AWS Unified Operations plan)
Production system down	`urgent`	1 hour	Your business is significantly impacted. Important functions of your application aren't available. (AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan)
Business-critical system down	`critical`	15 minutes	Your business is at risk. Critical functions of your application aren't available (Enterprise Support plan). Note that this is 30 minutes for the Enterprise On-Ramp Support plan.

Understanding AWS Support response times

AWS Support makes every reasonable effort to respond to your initial request within the indicated timeframe. For information about the scope of support for each Support plan, see AWS Support features.

If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, you have round-the-clock access for technical support. *For Developer Support, response targets for support cases are calculated in business hours. Business hours are generally defined as 08:00 to 18:00 in the customer country, excluding holidays and weekends. These times can vary in countries with multiple time zones. The customer country information appears in the **Contact Information** section of the My Account page in the AWS Management Console.

> **Note**
>
> If you choose Japanese as your preferred contact language for support cases, support in Japanese may be available as follows:
>
> - If you need customer service for non-technical support cases, or if you have a Developer Support plan and need technical support, support in Japanese is available during

business hours in Japan defined as 09:00 AM to 06:00 PM Japan Standard Time (GMT+9), excluding holidays and weekends.

- If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, technical support is available all day, every day in Japanese.

If you choose Chinese as your preferred contact language for support cases, support in Chinese may be available as follows:

- If you need customer service for non-technical support cases, support in Chinese is available 09:00 AM to 06:00 PM (GMT+8), excluding holidays and weekends.
- If you have a Developer Support plan, technical support in Chinese is available during business hours generally defined as 8:00 AM to 6:00 PM in your country as set in My Account, excluding holidays and weekends. These times may vary in countries with multiple time zones.
- If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, technical support is available all day, every day in Chinese.

If you choose Korean as your preferred contact language for support cases, support in Korean may be available as follows:

- If you need customer service for non-technical support cases, support in Korean is available during business hours in Korea defined as 09:00 AM to 06:00 PM Korean Standard Time (GMT+9), excluding holidays and weekends.
- If you have a Developer Support plan, technical support in Korean is available during business hours generally defined as 8:00 AM to 6:00 PM in your country as set in My Account, excluding holidays and weekends. These times may vary in countries with multiple time zones.
- If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, technical support is available all day, every day in Korean.

Changing the severity level of your support case

If you have Enterprise Support or an Enterprise On-Ramp plan, you can reassign your support case severity level to reflect changes to urgency and business impact. For example, you can change your support case from **System impaired** to **Production system impaired**. When you change the case

severity, AWS Support receives notification and attends to the case according to the new severity level.

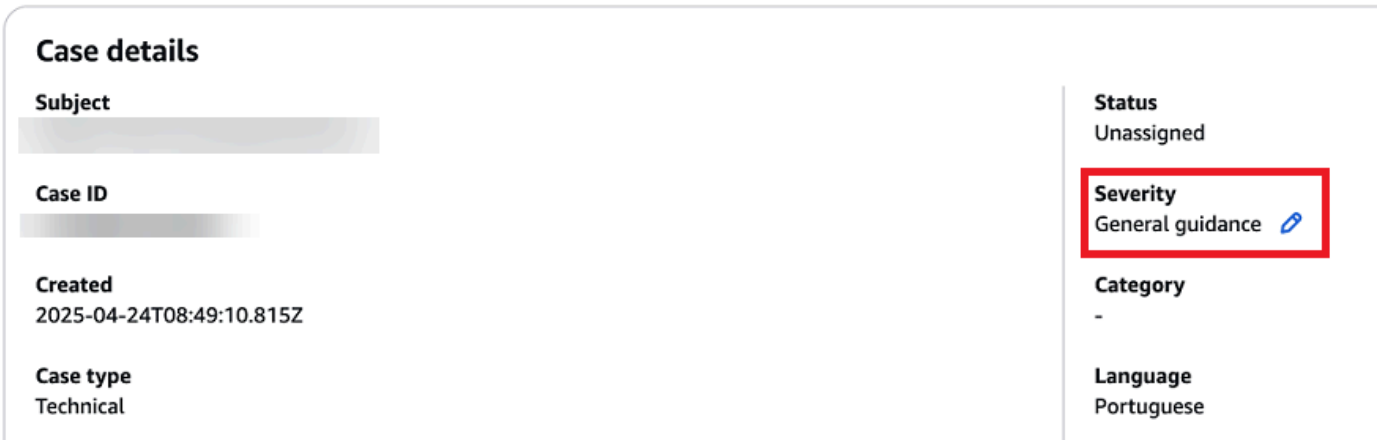
> **Note**
>
> Japanese (JP) account or billing, Service Quota Increase Request (SQIR), and Turkish (TR) account or billing cases created in these languages have a default severity and can't be changed.

To change the severity of a support case, complete the following steps:

1. Sign in to the AWS Support Center Console.

   > **Tip**
   >
   > In the AWS Management Console, you can also choose the question mark icon ( ) and then choose **Support Center**.

2. Select the case that you want to change the severity level for.
3. In **Case details**, choose the pencil icon next to the **Severity** field, as shown in the following example.

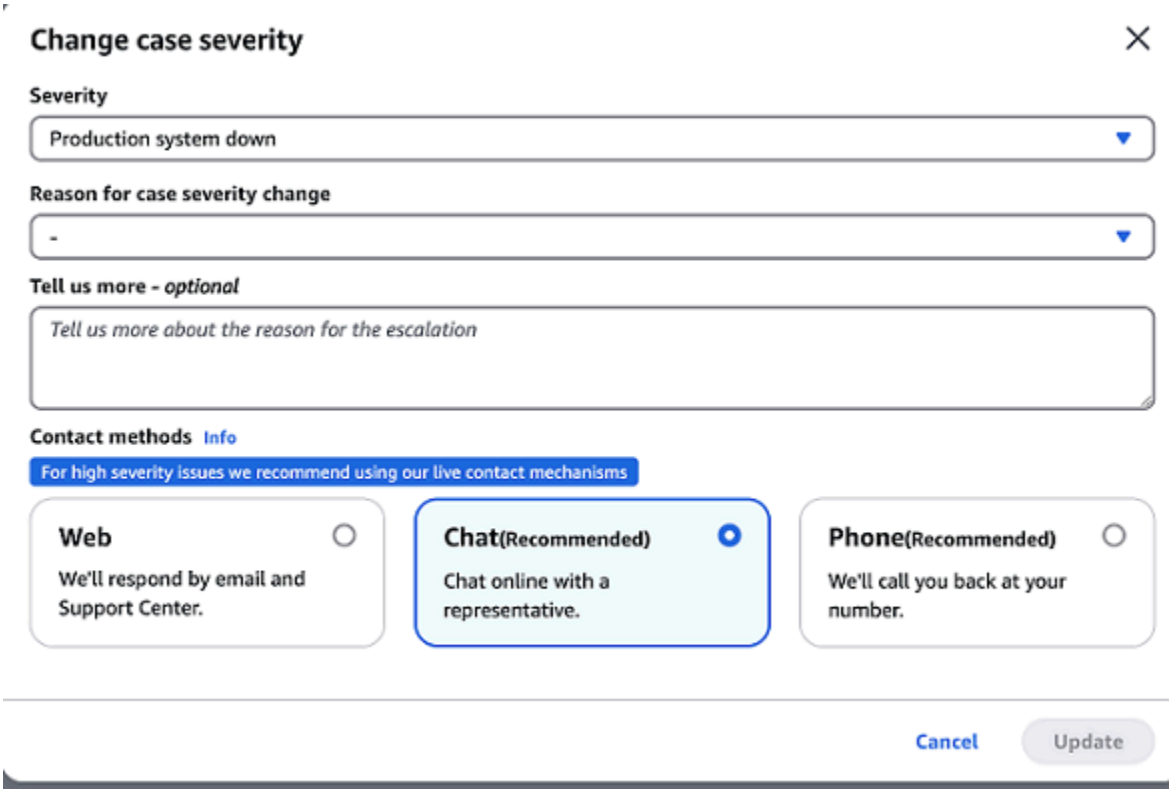


4. For **Severity**, choose the new severity level from the following options:
   - General guidance
   - System impaired

- Production system impaired
- Production system down
- Business-critical system down

5. For **Reason for case severity change**, choose from the available options for why you're changing the case severity.
6. (Optional) For **Tell us more**, enter additional information about this change.
7. Do one of the following:

   - If you're lowering the support case severity, or if you're raising it from **General guidance** to **System impaired** or **Production system impaired**, choose **Update**.
   - If you're raising the severity to **Production system down** or **Business-critical system down**, use one of the options in the **Contact methods** section to engage with AWS Support, and then choose **Update**. The following example shows the options available in the **Contact methods** section.

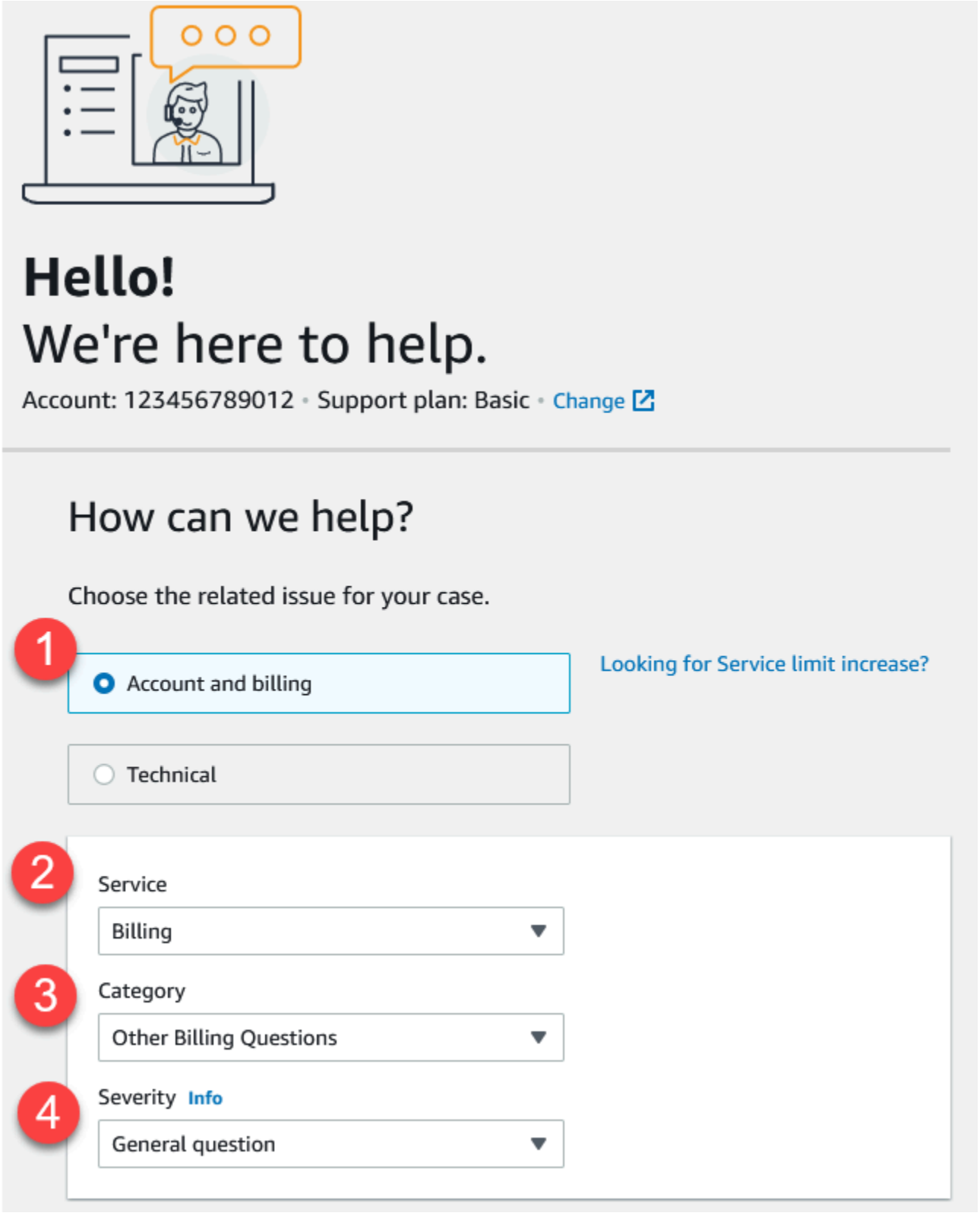


Note

- If you upgrade your support case severity to **Production system down** or **Business-critical system down**, you must wait 60 minutes before you can change the severity again.
- If your support case is currently set to **Business-critical system down**, you're prompted to initiate live contact with AWS Support instead of assigning a higher severity.
- If you're raising your support case severity level after already raising it at least once, you might encounter a waiting period. For example, if you change the severity from **System impaired** to **Production system impaired** at 6:00 AM, then your support case falls under the 4-hour first-response time for the **Production system impaired** severity level. In this scenario, you can upgrade the severity level again at 10:00 AM, after the 4-hour window. For a list of first-response times for each severity level, see the table in Understanding AWS Support response times.

Example: Create a support case for account and billing

The following example is a support case for a billing and account issue.



1. **Create case** – Choose the type of case to create. In this example, the case type is **Account and billing**.

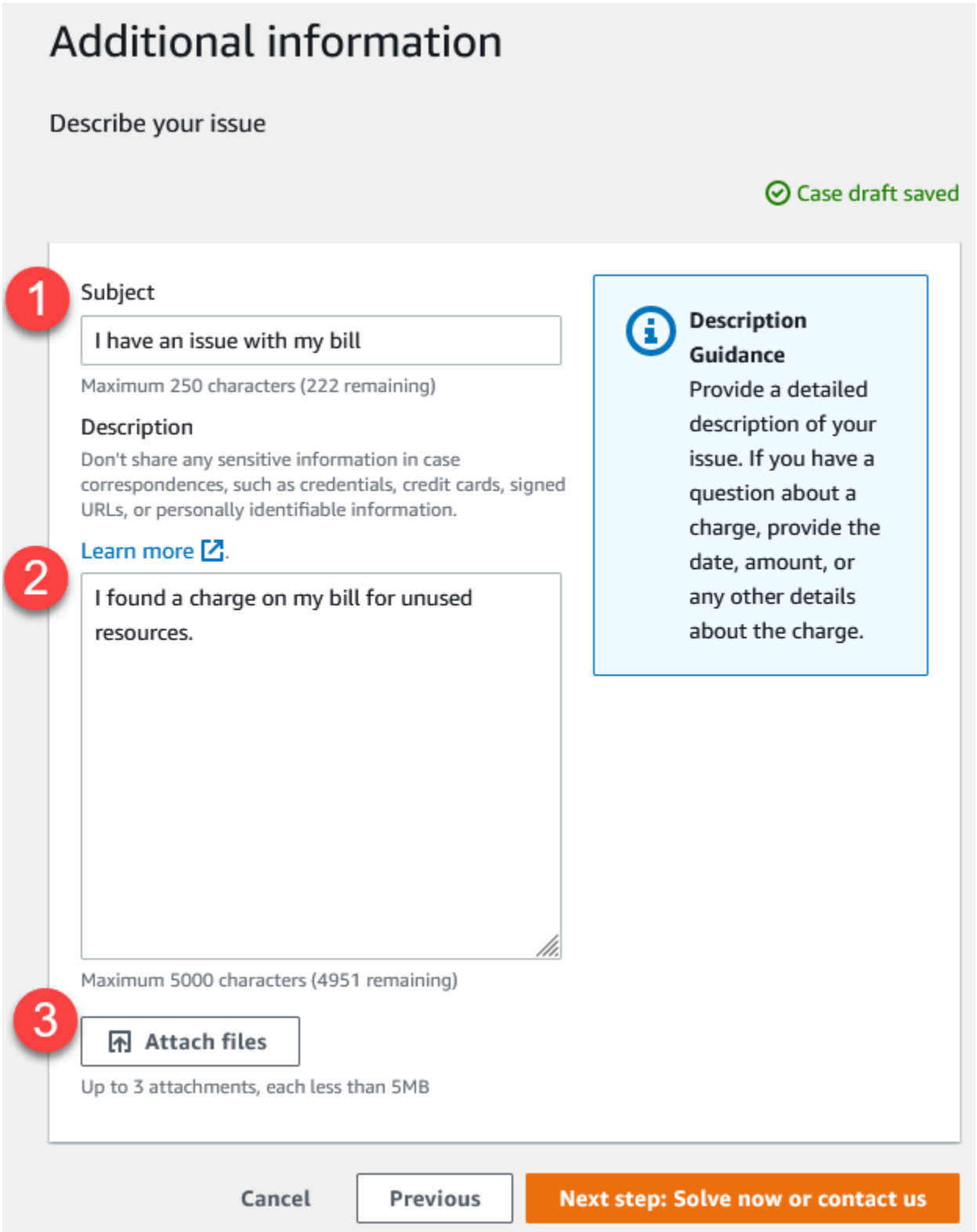
> **Note**
>
> If you have the Basic Support plan, you can't create a technical support case.

2. **Service** – If your question affects multiple services, choose the service that's most applicable.
3. **Category** – Choose the category that best fits your use case. When you choose a category, links to information that might resolve your problem appear below.
4. **Severity** – Customers with a paid support plan can choose the **General guidance** (1-day response time) or **System impaired** (12-hour response time) severity level. Customers with a Business Support plan can also choose **Production system impaired** (4-hour response) or **Production system down** (1-hour response). Customers with an Enterprise On-Ramp or Enterprise Support plan can choose **Business-critical system down** (15-minute response for Enterprise Support and 30-minute response for Enterprise On-Ramp).

   Response times are for first response from AWS Support. These response times don't apply to subsequent responses. For third-party issues, response times can be longer, depending on the availability of skilled personnel. For more information, see Choosing an initial support case severity level.

   > **Note**
   >
   > Based on your category choice, you might be prompted for more information.

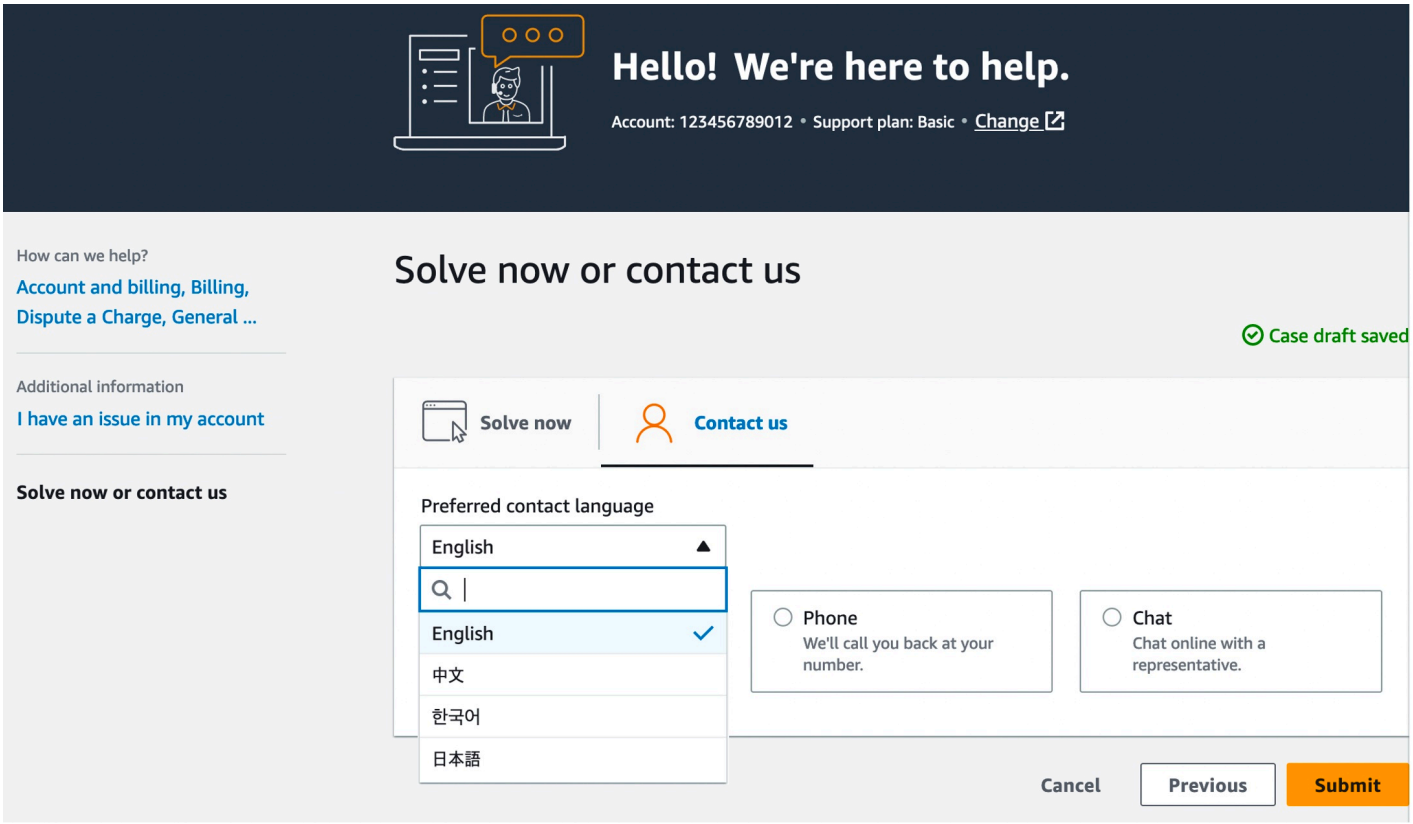
After you specify the case type and classification, you can specify the description and how you want to be contacted.



1. **Subject** – Enter a title that briefly describes your issue.

2. **Description** – Describe your support case. This is the most important information that you provide to Support. For some service and category combinations, a prompt appears with related information. Use these links to help resolve your issue. For more information, see Describing your problem.
3. **Attachments** – Attach screenshots and other files that can help support agents resolve your case faster. You can attach up to three files. Each file can be up to 5 MB.

After you add your case details, you can choose how you want to be contacted.



1. **Preferred contact language** – Choose your preferred language. Currently you can choose Chinese, English, Japanese, or Korean. The customized contact options in your preferred language will be shown by your support plan.
2. Choose a contact method. The contact options that appear depend on the type of case and your support plan.
   - If you choose **Web**, you can read and respond to the case progress in Support Center.
   - Choose **Chat** or **Phone**. If you choose **Phone**, you're prompted for a callback number.
3. Choose **Submit** when your information is complete and you're ready to create the case.

Note

If you choose Japanese as your preferred contact language for support cases, support in Japanese may be available as follows:

- If you need customer service for non-technical support cases, or if you have a Developer Support plan and need technical support, support in Japanese is available during business hours in Japan defined as 09:00 AM to 06:00 PM Japan Standard Time (GMT+9), excluding holidays and weekends.
- If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, technical support is available 24/7 in Japanese.

If you choose Chinese as your preferred contact language for support cases, support in Chinese may be available as follows:

- If you need customer service for non-technical support cases, support in Chinese is available 09:00 AM to 06:00 PM (GMT+8), excluding holidays and weekends.
- If you have a Developer Support plan, technical support in Chinese is available during business hours generally defined as 8:00 AM to 6:00 PM in your country as set in My Account, excluding holidays and weekends. These times may vary in countries with multiple time zones.
- If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, technical support is available 24/7 in Chinese.

If you choose Korean as your preferred contact language for support cases, support in Korean may be available as follows:

- If you need customer service for non-technical support cases, support in Korean is available during business hours in Korea defined as 09:00 AM to 06:00 PM Korean Standard Time (GMT+9), excluding holidays and weekends.
- If you have a Developer Support plan, technical support in Korean is available during business hours generally defined as 8:00 AM to 6:00 PM in your country as set in My Account, excluding holidays and weekends. These times may vary in countries with multiple time zones.

- If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, technical support is available 24/7 in Korean.

Legacy experience: Updating, resolving, and reopening your case

After you create your support case, you can monitor the status of your case in Support Center. A new case begins in the **Unassigned** state. When a support agent begins work on a case, the status changes to **Work in Progress**. The support agent might respond to your case to ask for more information (**Pending Customer Action**) or to let you know that the case is being investigated (**Pending Amazon Action**).

When your case is updated, you receive an email with the correspondence and a link to the case in Support Center. Use the link in the email message to navigate to the support case. You can't respond to case correspondences by email.

Notes

- You must sign in to the AWS account that submitted the support case. If you sign in as an AWS Identity and Access Management (IAM) user, you must have the required permissions to view support cases. For more information, see Manage access to AWS Support Center.
- If you don't respond to the case within a few days, AWS Support resolves the case automatically.
- Support cases that have been in the resolved state for more than 14 days can't be reopened. If you have a similar issue that is related to the resolved case, you can create a related case. For more information, see Creating a related case.

Topics

Updating an existing support case

You can update your case to provide more information for the support agent. For example, you can reply to correspondences, start another live chat, add additional email recipients, and so on.

> **Note**
>
> If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, you can reassign your support case severity level to reflect changes to urgency and business impact. If you don't have one of these support plans, then you can't update the severity of your case. For more information, see Choosing an initial support case severity level and Changing the severity level of your support case.

To update an existing support case

1. Sign in to the AWS Support Center Console.

   > **Tip**
   >
   > In the AWS Management Console, you can also choose the question mark icon () and then choose **Support Center**.

2. Under **Open support cases**, choose the **Subject** of the support case.
3. Choose **Reply**. In the **Correspondence** section, you can also make any of the following changes:
   - Provide information that the support agent requested
   - Upload file attachments
   - Change your preferred contact method
   - Add email addresses to receive case updates
4. Choose **Submit**.

> **Tip**
>
> If you closed a chat window and you want to start another live chat, add a **Reply** to your support case, choose **Chat**, and then choose **Submit**. A new pop-up chat window opens.

Resolving a support case

When you're satisfied with the response or your problem is solved, you can resolve the case in Support Center.

To resolve a support case

1. Sign in to the AWS Support Center Console.

   > **Tip**
   >
   > In the AWS Management Console, you can also choose the question mark icon () and then choose **Support Center**.

2. Under **Open support cases**, choose the **Subject** of the support case that you want to resolve.
3. (Optional) Choose **Reply** and in the **Correspondence** section, enter why you're resolving the case, and then choose **Submit**. For example, you can enter information about how you fixed the issue yourself in case you need this information for future reference.
4. Choose **Resolve case**.
5. In the dialog box, choose **Ok** to resolve the case.

> **Note**
>
> If AWS Support resolved your case for you, you can use the feedback link to provide more information about your experience with AWS Support.

Reopening a resolved case

If you're experiencing the same issue again, you can reopen the original case. Provide details about when the issue occurred again and what troubleshooting steps that you tried. Include any related case numbers so that the support agent can refer to previous correspondences.

Notes

- You can reopen your support case up to 14 days from when your issue was resolved. However, you can't reopen a case that has been inactive for more than 14 days. You can create a new case or a related case. For more information, see Creating a related case.
- If you reopen an existing case that has different information than your current issue, the support agent might ask you to create a new case.

To reopen a resolved case

1. Sign in to the AWS Support Center Console.

   Tip

   In the AWS Management Console, you can also choose the question mark icon () and then choose **Support Center**.

2. Choose **View all cases** and then choose the **Subject** or the **Case ID** of the support case that you want to reopen.
3. Choose **Reopen case**.
4. Under **Correspondence**, for **Reply**, enter the case details.
5. (Optional) Choose **Choose files** to attach files to your case. You can attach up to 3 files.
6. For **Contact methods**, choose one of the following options:
   - **Web** – Get notified by email and the Support Center.
   - **Chat** – Chat online with a support agent.
   - **Phone** – Receive a phone call from a support agent.
7. (Optional) For **Additional contacts**, enter email addresses for other people that you want to receive case correspondences.

8. Review your case details and choose **Submit**.

Creating a related case

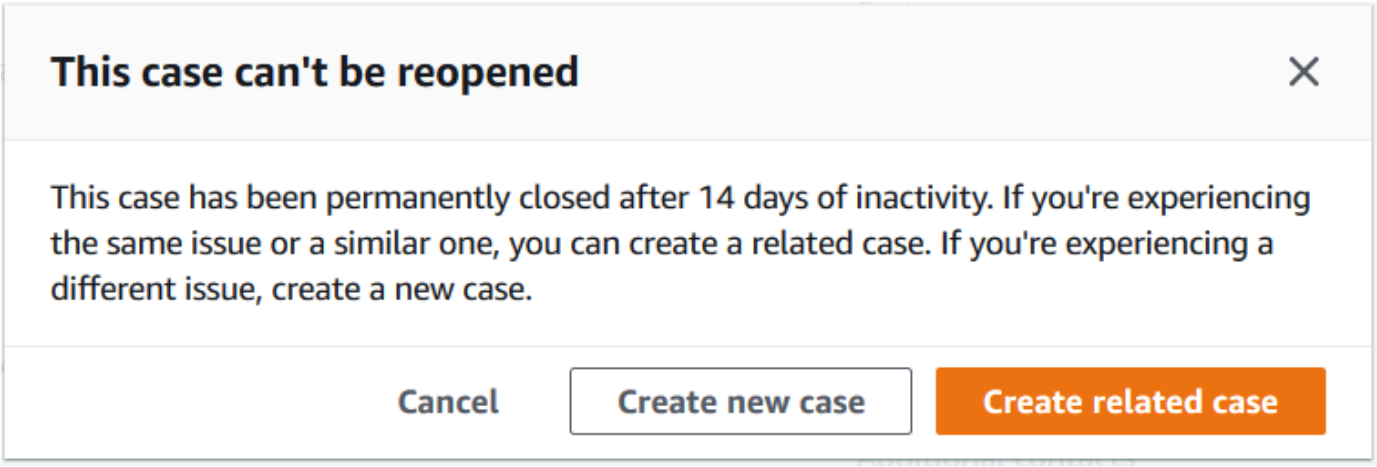
After 14 days of inactivity, you can't reopen a resolved case. If you have a similar issue that is related to the resolved case, you can create a related case. This related case will include a link to the previously resolved case, so that the support agent can review the previous case details and correspondences. If you're experiencing a different issue, we recommend that you create a new case.

To create a related case

1. Sign in to the AWS Support Center Console.

> **Tip**
>
> In the AWS Management Console, you can also choose the question mark icon ( ) and then choose **Support Center**.

2. Choose **View all cases** and then choose the **Subject** or the **Case ID** of the support case that you want to reopen.
3. Choose **Reopen case**.
4. In the dialog box, choose **Create related case**. The previous case information will be automatically added to your related case. If you have a different issue, choose **Create new case**.

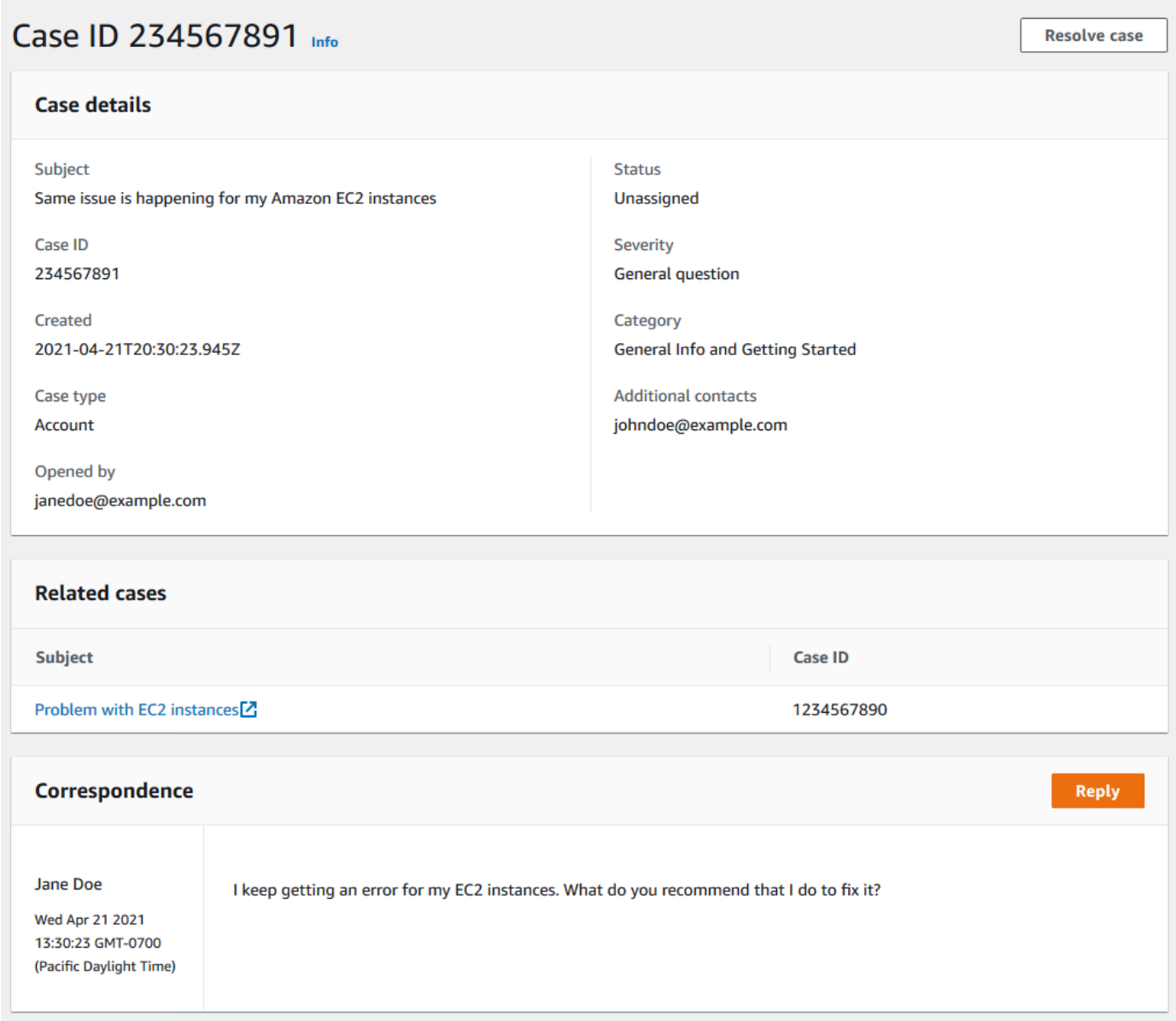


5. Follow the same steps to create your case. See Creating a support case.

> **Note**
>
> By default, your related case has the same **Type**, **Category**, and **Severity** of the previous case. You can update the case details as needed.

6. Review your case details and choose **Submit**.

After you create your case, the previous case appears in the **Related cases** section, such as in the following example.



Case history

You can view case history information up to 24 months after you create a case.

Using AWS Support with an AWS SDK

AWS software development kits (SDKs) are available for many popular programming languages. Each SDK provides an API, code examples, and documentation that make it easier for developers to build applications in their preferred language.

About the Support Center Console API

The Support Center Console API enhances your experience with the Support Center Console. Examples of the functionality provided by the Support Center Console API include the following:

- The ability for you to create and update a draft of your support case
- The ability for the Support Center Console to display the current status of your account
- The ability for the Support Center Console to display dynamic help for the selected service and category

For a complete list of actions provided by the Support Center Console API, see the table in Adding IAM policies for the Support Center Console API operations.

> **Important**
>
> To continue to use the functions included in the Support Center Console API, you must add the Support Center Console operations to your AWS Identity and Access Management policies before June 1, 2026. After you create the IAM policies, update the AWS Support managed policy to include the `support-console:*` actions. For more information, see Adding IAM policies for the Support Center Console API operations.

Topics

Adding IAM policies for the Support Center Console API operations

Before June 1, 2026, you must create AWS Identity and Access Management policies for the Support Center Console API operations. If you don't create these policies by June 1, 2026, you will receive `AccessDenied` errors.

To add these operations to your IAM policies, see Create IAM policies (console) in the *AWS Identity and Access Management User Guide*.

The following table summarizes the console operations.

> **Note**
>
> These operations are for the console only. They're not available for use in the AWS SDK or the AWS CLI.

Operation	Access level	Description
GetAccountState	READ	Grants permission for the console to show the current account state.
GetAccountGovCloudEnabled	READ	Grants permission to determine if your account is GovCloud enabled.
GetCaseDraft	READ	Grants permission for the console to show the case draft that you previously created.
CreateCaseDraft	WRITE	Grants permission to create or update a case draft for the given case type.
DeleteCaseDraft	WRITE	Grants permission to delete a case draft for the given case type.
GetBanner	READ	Grants permission for the console to show the Support banner displayed during customer impacting events.
DescribeDynamicHelp	READ	Grants permission for the console to show dynamic help

Operation	Access level	Description
		resources for the selected service and category.
CreateContact	WRITE	Grants permission for the console to create an authentic ated contact for the selected contact type.
CheckSubscription	READ	Grants permission for the console to verify if your account has access to the selected product.
GetQuestionnaire	READ	Grants permission for the console to show the customer feedback questionnaire.
SaveFeedback	WRITE	Grants permission to save questionnaire feedback.

Note

If you have a custom VPN configuration, then your IAM policies must allow the Support Center Console API endpoint in the aws.sourceIP conditions. If the Support Center Console API endpoint isn't allowed, then your ClientIp address won't forward to the API correctly. The following table provides the Support Center Console API endpoints by AWS Region.

AWS Region	Support Center Console API endpoint
`https://api.us-east-1.prod.support-console.support.aws.dev`	US East (N. Virginia)

AWS Region	Support Center Console API endpoint
`https://api.us-west-2.prod.support-console.support.aws.dev`	US West (Oregon)
`https://api.eu-west-1.prod.support-console.support.aws.dev`	Europe (Ireland)

Testing Support Center Console API calls

To validate that API calls to the console work, open the AWS Support Center Console. If the calls aren't successful, then you see a banner outlining the errors.

You can use AWS CloudTrail to debug the API calls made to the Support Center Console. The CloudTrail event for the API call shows if you have missing IAM policies. You can also investigate IP address forwarding issues by comparing your browser's IP addresses to the client IP address in the CloudTrail event.

To view CloudTrail events for calls to the Support Center Console, complete the following steps:

1. Sign in to the AWS Management Console and open the CloudTrail console at https://eusc-de-east-1.console.amazonaws-eusc.eu/cloudtrail/.
2. In the navigation pane, choose **Event history**. You see a filtered list of events with the most recent events showing first. The default filter for events is **Read only**, set to **false**. To clear the filter, choose **X** at the right of the filter.
3. Choose the event source **support-console.amazonaws.com**. On the event details page, you can view details about the event, see any referenced resources, and view the event record.

About the AWS Support API

The AWS Support API provides access to some of the features in the AWS Support Center.

The API provides two different groups of operations:

- Support case management operations to manage the entire life cycle of your AWS support cases, from creating a case to resolving it
- AWS Trusted Advisor operations to access AWS Trusted Advisor checks

> **Note**
>
> You must have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan to use the AWS Support API. For more information, see Support.

For more information about the operations and data types provided by Support, see the AWS Support API Reference.

Topics

Support case management

You can use the API to perform the following tasks:

- Open a support case
- Get a list and detailed information about recent support cases
- Filter your search for support cases by dates and case identifiers, including resolved cases
- Add communications and file attachments to your cases, and add the email recipients for case correspondences. You can attach up to three files. Each file can be up to 5 MB
- Resolve your cases

The AWS Support API supports CloudTrail logging for support case management operations. For more information, see Logging AWS Support API calls with AWS CloudTrail.

For code examples that demonstrate how to manage the entire life cycle of a support case, see Code examples for Support using AWS SDKs..

AWS Trusted Advisor

You can use the Trusted Advisor operations to perform the following tasks:

- Get the names and identifiers for the Trusted Advisor checks
- Request that a Trusted Advisor check be run against your AWS account and resources
- Get summaries and detailed information for your Trusted Advisor check results
- Refresh your Trusted Advisor checks
- Get the status of each Trusted Advisor check

The AWS Support API supports CloudTrail logging for Trusted Advisor operations. For more information, see AWS Trusted Advisor information in CloudTrail logging.

You can use Amazon CloudWatch Events to monitor for changes to your check results for Trusted Advisor. For more information, see Monitoring AWS Trusted Advisor check results with Amazon EventBridge.

For example Java code that demonstrates how to use the Trusted Advisor operations, see Using Trusted Advisor as a web service.

Endpoints

Support is a global service. This means that any endpoint that you use will update your support cases in the Support Center Console.

For example, if you use the US East (N. Virginia) endpoint to create a case, you can use the US West (Oregon) or Europe (Ireland) endpoint to add a correspondence to the same case.

You can use the following endpoints for the Support API:

- US East (N. Virginia) – https://support.us-east-1.amazonaws.com
- US West (Oregon) – https://support.us-west-2.amazonaws.com

- Europe (Ireland) – https://support.eu-west-1.amazonaws.com

> **Important**
>
> - If you call the CreateCase operation to create test support cases, then we recommend that you include a subject line, such as **TEST CASE-Please ignore**. After you're done with your test support case, call the ResolveCase operation to resolve it.
> - To call the AWS Trusted Advisor operations in the AWS Support API, you must use the US East (N. Virginia) endpoint. Currently, the US West (Oregon) and Europe (Ireland) endpoints don't support the Trusted Advisor operations.

For more information about AWS endpoints, see AWS Support endpoints and quotas in the *Amazon Web Services General Reference*.

Support in AWS SDKs

The AWS Command Line Interface (AWS CLI), and the AWS Software Development Kits (SDKs) include support for the Support API.

For a list of languages that support the AWS Support API, choose an operation name, such as CreateCase, and in the See Also section, choose your preferred language.

AWS Support Plans

AWS offers the following AWS Support Plans for you to choose from based on your business needs.

- Basic
- Developer
- Business
- Enterprise On-Ramp
- AWS Enterprise Support

> **Important**
>
> The new AWS Support plans aren't available in your AWS Region. Your existing Developer Support, Business Support, Enterprise Support or Enterprise On-Ramp plan remains in effect.

Topics

Features of AWS Support Plans

Basic Support offers assistance for account and billing questions and service quota increases. The other plans offer a number of technical support cases with pay-by-the-month pricing and no long-term contracts.

All AWS customers automatically have 24x7 access to these features of Basic Support:

- One-on-one responses to account and billing questions

- Support forums
- Service health checks
- Documentation, technical papers, and best practice guides

Customers with a Developer Support plan have access to these additional features:

- Best practice guidance
- Client-side diagnostic tools
- Building-block architecture support: guidance on how to use AWS products, features, and services together
- Supports an unlimited number of support cases that can be opened by any user with permissions.

In addition, customers with a Business, Enterprise, or Enterprise On-Ramp plan have access to these features:

- Use-case guidance – What AWS products, features, and services to use to best support your specific needs.
- AWS Trusted Advisor – A feature of Support, which inspects customer environments and identifies opportunities to save money, close security gaps, and improve system reliability and performance. You can access all Trusted Advisor checks.
- The AWS Support API to interact with Support Center and Trusted Advisor. You can use the AWS Support API to automate support case management and Trusted Advisor operations.
- Third-party software support – Help with Amazon Elastic Compute Cloud (Amazon EC2) instance operating systems and configuration. Also, help with the performance of the most popular third-party software components on AWS. Third-party software support isn't available for customers on Basic or Developer Support plans.
- Supports an unlimited number of AWS Identity and Access Management (IAM) users who can open technical support cases.

In addition, customers with an Enterprise On-Ramp or Enterprise Support plan have access to these features:

- Application architecture guidance – Consultative guidance on how services fit together to meet your specific use case, workload, or application.

- Infrastructure event management – Short-term engagement with AWS Support to get a deep understanding of your use case. After analysis, provide architectural and scaling guidance for an event.
- Technical account manager – Work with a technical account manager (TAM) for your specific use cases and applications.
- Management business reviews.

For more information about features and pricing for each support plan, see AWS Support and Compare AWS Support plans. Some features, such as 24x7 phone and chat support, aren't available in all languages.

> **Note**
>
> If you work with an AWS partner and want to learn more about Partner-led Support, see AWS Partner-Led Support

What is AWS Unified Operations

> **Note**
>
> AWS Unified Operations is available in the commercial AWS Regions only.

AWS Unified Operations combines proven expertise with AI-powered insights to help reduce operational and security risks, resolve issues faster, and help you architect more resilient cloud solutions from the start—accelerating your most critical cloud initiatives. Designated AWS specialists act as an extension of your team through your preferred collaboration channels—conducting workload reviews, providing strategic guidance, and optimizing performance through deep contextual knowledge. With 24/7 security and performance monitoring, we detect and mitigate incidents early while reducing alert volume. A five-minute, context-aware response for critical incidents further shortens resolution time and helps maintain peak operational performance.

AWS Unified Operations pricing

AWS Unified Operations pricing is based on your specific requirements and workload complexity. For detailed pricing information, see AWS Support Plan Pricing.

Contents

- Unified Operations Getting started: What you can expect from AWS

Benefits of Unified Operations

Unified Operations offers several key benefits.

- **Designated AWS experts:** Your AWS team includes designated Technical Account Managers (TAMs), AWS Domain Specialist Engineer, and Senior Billing and Account Specialist. Your AWS team integrates with your collaboration tools like Slack or Microsoft Teams.
- **Deep technical guidance:** Your AWS team helps you strengthen your resiliency through application-specific deep dives, readiness assessments, guided testing support, and custom runbooks tailored to your environment. Your AWS team provides workload focused financial management and aligns costs strategically with your business objectives.
- **Migrations, events, and launch support:** Accelerate your critical cloud migrations and business events with designated AWS engineers who help mitigate risks proactively, guiding you from planning through execution. Real-time assistance for scheduled events, increasing migration velocity and successful launches.
- **24x7 Proactive workload monitoring:** Onboard application and infrastructure level alarms from your existing AWS and third-party tools to detect early warning signs and drive proactive mitigation with your team.
- **Engage AWS incident managers within 5 mins for business-critical system down issues:** Receive proactive support from Incident Management Engineers within 5 minutes of alarms, workload alerts, or business-critical system down issues.
- **Context-specific case response:** Access to context-aware support engineers to drive incident resolution. Runbooks and incident response playbooks customized for your workflows, streamlining problem diagnosis and resolution for your specific business needs.
- **Security guidance and support:** Proactive monitoring of security events with automated triage and investigation, along with 24/7 access to the AWS Customer Incident Response Team to prepare for, respond to, and recover from security events in your AWS environment.

Unified Operations Team

AWS Unified Operations brings together a designated team of specialized experts who work together to support your cloud journey. Each role is carefully designed to provide comprehensive coverage across technical, financial, operational, and security dimensions of your cloud

environment. From strategic guidance to technical support, from financial optimization to rapid incident response, these experts act as an extension of your team, available 24/7 to ensure your cloud operations run efficiently and securely.

Contents

Technical Account Manager

Your designated cloud strategist, the Technical Account Manager (TAM) orchestrates the overall engagement and drives business outcomes. TAMs lead strategic planning, conduct quarterly business reviews, provide resiliency guidance, and coordinate across specialized teams to ensure your cloud initiatives succeed. They serve as your primary point of contact for escalations and strategic decisions.

Domain Specialist Engineer

A Domain Specialist Engineer (DSE) is a technical expert who deeply understands your specific workload architecture and AWS services. DSEs conduct critical workload reviews, create technical documentation, develop troubleshooting guides, and provide ongoing technical consultation. They analyze incidents to prevent recurrence and maintain global knowledge sharing to make sure that there is consistent support quality.

Senior Billing and Account Specialist

Your Senior Billing and Account Specialist (SBAS) is a designated financial optimization expert who helps balance performance with cost. They manage cost optimization strategies, oversee reserved instance and savings plan portfolios, conduct financial business reviews, and provide detailed spend analytics to maximize your cloud investment efficiency.

Incident Management Engineer

Incident Management Engineers (IME) are rapid response specialists who coordinate critical incident resolution. IMEs provide 5-minute response times, orchestrate technical teams during incidents, manage stakeholder communications. During active incidents, IMEs conduct real-time assessments of incident handling and response effectiveness, upon request they document the sequence of events, decisions made, and immediate outcomes. They observe and evaluate the execution of response protocols, team coordination, and the application of existing playbooks while the incident is still unfolding.

Migration Specialist

On-demand experts who guide critical transitions like launches and migrations. They validate architectures, create detailed execution plans, provide real-time monitoring during events, and conduct post-event analysis to capture learnings and optimize future operations.

AWS Customer Incident Response Team

The AWS Customer Incident Response Team (CIRT) are security experts providing 24/7 specialized assistance for security events. They monitor security findings, provide guided response within minutes of detection, and enhance your security operations capabilities through expert investigation support and best practices guidance.

Specialist Support Engineer

Specialist Support Engineers (SSE) are highly experienced technical experts using advanced contextual tools to deliver precise support solutions. They leverage AI-powered systems and deep technical knowledge to quickly understand your environment and resolve complex technical challenges.

Unified Operations life cycle

The Unified Operations Support plan contains different phases, from pre-onboarding, through continuous improvement, that help you get the most our of your cloud environment. This topic covers key points of each phase.

Contents

- Unified Operations onboarding
- Unified Operations Pre-event or migration planning
- Unified Operations Event migration or cutover
- Unified Operations Post Go-live or event
- Unified Operations Workload incident management
- Unified Operations Post incident
- Unified Operations Continuous improvement

Unified Operations pre-onboarding

During the pre-onboarding phases, AWS gathers information from you needed to start the onboarding process, including the following information:

Environment discovery and technical validation

- Understanding of your workload architecture and your key AWS services.
- Your future planning needs, such as migration or events.
- Pre-requisites, such as a list of your AWS accounts and AWS Regions.
- Your specific business needs.

Unified Operations onboarding

Onboarding kickoff

- Meet the team (customer and AWS allocated resources).

Onboarding workshop

- Identify critical workloads.
- Conduct deep architecture review of the workloads.
- Review roles and responsibilities (RACI review – AWS and your roles and responsibilities).
- Review Incident and Change management workflows (ITSM).
- Communication protocol - tooling and processes, escalation path, on-call schedules

- Identify and define the critical alarms (in Amazon CloudWatch, your 3rd party APM or custom monitoring tool).
- Build runbooks for critical alarms.

Service onboarding

- Your AWS account onboarding to AWS Security Incident Response.
- Critical alarm onboarding to rapid Incident Management.

Unified Operations Pre-event or migration planning

Pre-event or migration planning in AWS Unified Operations includes the following key elements.

- **Context gathering :** Workload discovery and architecture.
- **Operational Readiness Review (ORR):** Systematic assessment against AWS best practices and runbooks to identify potential issues before they impact the event.
- **Risk assessments:** Identify, list potential risks, and mitigation plans.
- **Review Events:** Secure the cut-over or migration event support.

Unified Operations Event migration or cutover

Event migration or cutover in Unified Operations includes the following key elements.

- **Real-time bridge to resolution:** AWS experts join your communication channels to monitor events and resolve issues during critical business periods.
- **24/7 comprehensive support:** Round-the-clock expert assistance with immediate guidance for resource scaling needs. .
- **AWS engineering:** Work directly with engineers who understand your business, delivering tailored solutions.

Unified Operations Post Go-live or event

The post go-live or post event process in Unified Operations includes the following key elements:

- Spin-down engagement and event-specific resources.

- Conduct event reviews.
- Update runbooks and documentation based on learnings.
- Perform retrospectives to identify areas for improvement.

Unified Operations Workload incident management

Workload incident management includes the following key elements:

- AWS Support case creation.
- Engagement with you and an AWS Incident Management Engineer (IME) for context-aware incident management.
- Joining or creation of incident bridge call.
- Monitoring of AWS service health and large-scale event (LSE).
- Rapid recovery of critical applications or workloads.

Unified Operations Post incident

Post-incident analysis for critical incidents is conducted by the assigned Domain Specialist Engineer (DSE). DSE takes a broader analytical approach after resolution, conducting comprehensive root cause analysis to identify any gaps in processes or tooling. The DSE transforms insights into actionable improvements by updating response playbooks, recommending preventive measures, and suggesting architectural enhancements to help prevent similar incidents in the future.

Unified Operations Continuous improvement

Continuous improvement includes the following key elements:

- Update Critical Workload Reviews, proposing AWS service-specific recommendations and resilience guidance .
- Review old and new cases on a continuous basis, and provide troubleshooting guides for identified technical issues.
- Apply lessons and oversee implementations and testing.
- Discuss technical issues, configurations, past, and upcoming projects and milestones.
- Review new feature implementation plans with risk assessments and performance optimizations.
- Conduct Monthly or Quarterly Business Review (MBR / QBR).

Getting started with Unified Operations

This topic discusses the steps to onboard to AWS Unified Operations.

Contents

Unified Operations Getting started: Prerequisites

The following items are required to onboard to AWS Unified Operations

A signed AWS Unified Operations contract. For more information, contact your AWS sales representative.

- Identified business needs, such as migration, modernization, events, target uptime, and so on.
- A list of your workloads.
- A list of your AWS accounts and associated AWS Regions.
- Identified stakeholders across Application, Architecture, Operations, and Security teams.

Unified Operations Getting started: Onboard critical alarms to rapid incident management

To help quickly notify you of critical incidents, complete the following steps to onboard your alarms to AWS Incident Detection and Response

1. Define and configure your critical alarms for rapid incident management. For detailed information, see Define and configure alarms in Incident Detection and Response in the *Incident Detection and Response User Guide*.

a. For steps to set up alarms using Amazon CloudWatch, see Define and configure alarms in Incident Detection and Response in the *Incident Detection and Response User Guide*. For AWS recommendations on critical alarm types for various AWS services, see Incident Detection and Response (IDR). Contact your AWS Unified Operations team if you want AWS to automate the creation of critical AWS alarms for your tagged AWS resources.

b. To redirect or ingest critical alarms from 3rd party APM tools with direct Amazon EventBridge integration, such as DataDog, NewRelic, and so on, see Ingest alarms from APMs that have direct integration with Amazon EventBridge in the *AWS Incident Detection and Response User Guide*. You must deploy a set of AWS resources (AWS Lambda and Amazon EventBridge event bus rules) to transform and redirect your alarm (event) to AWS Incident Detection and Response. Your AWS Unified Operations team can help provide the CloudFormation template to install these resources.

c. Redirect or ingest critical alarms from your custom monitoring tool through a 3rd party APM tool that doesn't have direct integration with Amazon EventBridge, such as DataDog, NewRelic, and so on. For more information, see Ingest alarms from APMs that have direct integration with Amazon EventBridge in the *AWS Incident Detection and Response User Guide*. You must deploy a set of AWS resources (API Gateway AWS Lambda functions, and Amazon EventBridge event bus rules) to transform and redirect your alarm (event) to AWS Incident Detection and Response. Your AWS Unified Operations team can help provide the CloudFormation template to install these resources.

2. Provide workload architecture details, point of contact information and runbook information on mitigation actions for critical alarms. To do this, complete the following steps:

   - Download and complete the AWS Incident Detection and Response Workload onboarding questionnaire for each critical workload or application and the Alarm ingestion questionnaire related to each unique workload.

     The information in these questionnaires helps the AWS team develop an incident remediation runbook. This runbook enables appropriate actions to be taken to quickly troubleshoot and remediate critical alarms before they cause business downtime. For examples and sample information, see Workload onboarding and alarm ingestion questionnaires in AWS Incident Detection and Response.

3. Provide access to onboard your critical alarms to AWS Incident Detection and Response

   a. Deploy the `AWSServiceRoleForHealth_EventProcessor` service-linked role (SLR) in your AWS account running the critical workload to be monitored by the AWS incident

management team. For more information, see Provision access for alert ingestion to AWS Incident Detection and Response.

> **Note**
>
> To assist your with onboarding of large AWS accounts, AWS can provide you with a AWS Command Line Interface script to fast track the provisioning of this SLR.

b. (Optional) If your alarms are in Amazon CloudWatch, make sure that the AWS Identity and Access Management user or role that's used for alarm testing (before go-live) has the `cloudwatch:SetAlarmState` IAM permission in your AWS account that's running the critical workload. This is needed for alarm testing (gameday) post onboarding. For more information, see Test onboarded workloads in AWS Incident Detection and Response.

4. Create a AWS Support case to subscribe a workload for rapid incident management. Note that your AWS account is automatically enabled for inbound rapid incident management, which means you can raise a case to the Unified Operations Incident Detection and Response queue through the Support Center Console, the AWS Command Line Interface, or the AWS SDK for quick action. For AWS to proactively monitor and create incidents with an outbound AWS Support case, create an AWS Support case for your critical workload. To do this, complete the following steps:

   a. Sign in to the AWS Support Center Console, select **Create case**, and then select **Technical support**.

   b. For **Service** select **Incident Detection and Response**.

   c. For **Category** select **Onboard new workload**.

   d. For **Severity** select **General guidance**.

   e. Attached the Workload and Alarm questionnaires that you completed in the previous step.

Unified Operations Getting started: How to request 5-minute incident response

AWS Unified Operations offers 5-minute incident response for your critical incidents. To request a 5-minute inbound response you can create a support case from a support interaction or use the legacy support case creation method. When you create your case, make sure that you enter the following information to ensure that your case receives a response within 5 minutes:

1. For **Case type**, choose **Technical**.

2. For **Service**, choose **AWS Incident Detection and Response**.
3. For **Category**, choose **Active Incident**.
4. For **Severity**, choose **Business-critical system down**.
5. In the **Description**, include the following information

    a. Technical information

    - Workload name
    - Affected AWS Resource ARN(s)

    b. Business information

    - Description of impact to the business
    - (Optional) Customer bridge details

Unified Operations Getting started: Plan for domain coverage

AWS Unified Operations provides specialized expertise through a domain-based coverage approach. Each domain is supported by a team of AWS Domain Specialists who provide the following services:

- **Specialized expertise** aligned to your specific technology areas.
- **Continuous coverage** with availability through your preferred collaboration tools (Slack or Microsoft Teams) during business days.
- **Proactive guidance** on architecture, best practices, and optimization opportunities.
- **Enhanced incident response** through deep domain knowledge and workload familiarity.
- **Consistent experience** maintained by a coordinated team rather than individuals.

This approach to domain coverage enables AWS specialists to maintain deep familiarity with your critical workloads while providing comprehensive support across your technology stack.

To select the domains, organizations maintain decision authority from a choice of 23 AWS Domains and consider the following factors in their decision:

- Primary AWS services running critical workloads
- Critical AWS service dependencies (such as Amazon EC2, Amazon EKS, or Amazon RDS)

- Major upcoming events requiring 24x7 support coverage (migrations, launches) planned within 3-6 months

This information, combined with guidance from your Technical Account Manager, enables precise alignment of domain expertise with your specific organizational needs, helping you maintain optimal support for mission-critical workloads.

Unified Operations Getting started: Onboard your account to proactive security incident management

Unified Operations entitles you to AWS Security Incident Response to help you quickly prepare for, respond to, and recover from security incidents, such as account takeovers, data breaches, and ransomware attacks. AWS Security Incident Response triages findings, escalates events, and manages critical cases, while also providing access to the AWS Customer Incident Response Team (CIRT) to investigate impacted resources. This access helps you to effectively mitigate and resolve security incidents, minimizing the impact on your operations. To onboard to this service feature, complete the following steps:

1. Create a centralized AWS account for AWS Security Incident Response. This AWS account will be used to configure all other AWS accounts that you want monitored, to manage your incident response team, and to create and view security events. We recommend that you to align this account with the account that you use for other security services such as Amazon GuardDuty and AWS Security Hub CSPM. You can use an AWS Organizations management account, or an AWS Organizations delegated administrator account as the Security Incident Response membership account. For more information, see Select a membership account in the *AWS Security Incident Response User Guide*.

    a. Choose basic membership details. For more information, see Setup membership details in the *AWS Security Incident Response User Guide*.

    b. Choose how you want to associate accounts with AWS Organizations. For more information, see Associate accounts with AWS Organizations in the *AWS Security Incident Response User Guide*.

    c. (Optional) You can optionally enable proactive response and alert triaging workflow to enable within your organization to monitor and investigate alerts generated from Amazon GuardDuty and AWS Security Hub CSPM integrations. For more information, see Setup proactive response and alert triaging workflows in the *AWS Security Incident Response User Guide*.

2. (Optional) Enable the proactive containment of a potential security incident. AWS can perform containment actions to quickly mitigate impact, such as isolating compromised hosts or rotating credentials. To turn on this feature, you must first grant the necessary permissions to the service. To do this, deploy an Step Functions StackSet.

Unified Operations Getting started: AWS expectations from you

For Unified Operations to deliver maximum value, we recommend the following collaborative approach:

Team engagement

- Identify subject matter experts from your team to collaborate with AWS engineers during onboarding and ongoing engagement.
- Participate in initial discovery calls and subsequent meetings to share architecture details and operational requirements.
- Establish regular touchpoints to review architecture updates or workload changes.

Operational integration

- Configure critical alarms in your account to enable effective incident management.
- Implement recommended action items provided by AWS specialists,
- Participate in gameday exercises to validate incident response processes.

This collaborative framework helps you maximize the value of Unified Operations, achieve your uptime goals, mitigate operational risks, and receive comprehensive support for your mission-critical workloads.

Unified Operations Getting started: What you can expect from AWS

When you onboard to Unified Operations, you can expect the following from AWS.

- Provide a team of designated AWS experts with deep technical expertise in the your workload domain and services.
- Offer proactive guidance, ongoing optimization, and continuous improvement recommendations to enhance workload performance and resiliency and accelerate path to migrations and modernization.

- Help provide rapid incident response, with context-aware engineers engaged within 5 minutes of a critical incident.
- Offer comprehensive support throughout the application lifecycle, from design and migration to production launch and long-term operations.
- Proactively monitor security threats with auto-triaging, reducing false positives, and raising incidents for potential security incidents.
- Assist in trouble and joint mitigation of AWS or your identified security incident.

Change AWS Support Plans

You can use the AWS Support Plans console to change your support plan for your AWS account. To change your support plan, you must have AWS Identity and Access Management(IAM) permissions. For more information, see Manage access to AWS Support Plans and AWS managed policies for AWS Support Plans.

To change your support plan

1. Sign in to the AWS Support Plans console at https://eusc-de-east-1.console.amazonaws-eusc.eu/support/plans/home.
2. (Optional) To compare support plans, on the **AWS Support Plans** page, choose **Compare all Support plans and features**.
3. (Optional) To view estimated costs for a support plan, choose **Pricing calculator**. In the **Pricing calculator**, select a support tier, enter an estimate of how much you expect to spend with AWS each month, and then choose **Calculate**.
4. To downgrade your Enterprise Support plan, reach out to your Technical Account Manager (TAM).

   To downgrade your Business Support+ plan, on the Manage Support Plans page choose **Review downgrade** in the Basic Support plan section.

   To upgrade to an AWS Enterprise Support or AWS Unified Operations plan, choose **Contact sales**.

   To upgrade to an AWS Business Support+ plan from Basic Support, complete the following steps:

   a. Choose **Get started** in the AWS Business Support+ section.

b. If you are onboarded to AWS Organizations and have the all-feature mode enabled, you can subscribe your entire organization to Business Support+. For information about all-feature mode, see Enabling all features for an organization with AWS Organizations in the *AWS Organizations User Guide*. To enroll at the organization level, select the **My organization** radio button. Review your **Consolidated Billing** estimate, then check the box to agree to the subscription terms. Only your organization's management account can subscribe your entire organization.

   Or, to enroll at the account level, select the **My account** radio button, then check the box to agree to the subscription terms.

c. Choose **Confirm upgrade** to complete your AWS Business Support+ plan subscription.

> **Note**
>
> If you sign up for a paid support plan, you're responsible for a minimum one month subscription of AWS Support. For more information, see the AWS Support FAQs.

Related information

For more information about AWS Support Plans, see the AWS Support FAQs. You can also choose **Contact us** from the Support Plans console.

To close your account, see Closing an Account in the *AWS Billing User Guide*.

Configure promotional plan expiration notifications

You can use AWS User Notifications to configure notifications to inform you when your support plan's promotional period is ending. You can subscribe to receive notifications by email, in the AWS Console Mobile Application, or in other chat channels of your choice.

Configure promotional support plan expiration notifications

1. Open User Notifications in the AWS Management Console:

   a. Choose the bell icon in the top navigation bar.

   b. Choose **Notification center**.

c. In the navigation pane, choose **Notification configuration**.

d. Choose **Create notification configuration**.

e. Select at least one **Configuration hub**. For more information, see Storing, processing, and replicating notifications using notification hubs in AWS User Notifications.

2. For **Event Rule**, enter the following information:

- For **AWS service name**, enter **Support Plans**.
- For **Event type**, enter **Support Plan Promotion Expiration**.
- For **Regions**, select the source AWS Regions where you want to receive notifications. For this option, choose US East (N. Virginia), US East (Ohio), US West (N. California), and US West (Oregon).

3. Configure aggregation settings to reduce notification frequency. We recommend that you set aggregation to **Receive within 5 minutes**.
4. Configure the delivery channels where you want to receive notifications. If you don't select a delivery channel, you can view notifications by selecting the bell icon in the AWS Management Console navigation bar.

For detailed instructions on creating user-configured notifications, see Step 1: creating a notification configuration in the AWS User Notifications User Guide.

View promotional plan notifications

Your notifications are delivered to the delivery channel that you chose during configuration. You can also view notifications by choosing the bell icon in the console navigation bar. The bell icon shows a red badge when new notifications are available.

For more information on viewing notifications, see Step 2: Viewing notifications in the *AWS User Notifications User Guide*.

Developer, Business, and Enterprise On-Ramp end of support

> **Important**
>
> This information applies to the commercial AWS Regions only.

For information on transitioning to a new plan, see the following information specific to your plan and the AWS Support Frequently Asked Questions.

> **Note**
>
> Developer Support, Business Support, and Enterprise On-Ramp will remain available in the AWS GovCloud (US) Region.

Developer Support plan end of support

End of Support Notice: Developer Support will be discontinued January 1, 2027. Customers with Developer Support can continue using their existing plan or choose to upgrade to Business Support+ anytime before January 1, 2027. Business Support+ delivers AI-powered assistance that understands the context of your operations, with 24/7 access to AWS experts at $29/month minimum per account. For more information see, Business Support+ plan details.

Business Support plan end of support

End of Support Notice: Business Support will be discontinued January 1, 2027. Customers with Business Support can continue using their existing plan or choose to upgrade to Business Support+ anytime before January 1, 2027. Business Support+ delivers AI-powered assistance that understands the context of your operations, with 24/7 access to AWS experts at $29/month minimum per account. For more information see, Business Support+ plan details

Enterprise On-Ramp end of support

End of Support Notice: On January 1, 2027, AWS will discontinue Enterprise On-Ramp. Throughout 2026, Enterprise On-Ramp customers will be automatically upgraded to AWS Enterprise Support during contract renewal or in periodic batches. Customers will receive an email notification a month before their upgrade. No further action is required. Enterprise Support provides designated TAM assignment, 15-minute response times, and AWS Security Incident Response available at no additional cost, all at a lower $5,000 minimum (reduced from $15,000). For more information, see AWS Enterprise Support plan details.

AWS Trusted Advisor

Trusted Advisor draws upon best practices learned from serving hundreds of thousands of AWS customers. Trusted Advisor inspects your AWS environment, and then makes recommendations when opportunities exist to save money, improve system availability and performance, or help close security gaps.

If you have a Basic or Developer Support plan, you can use the Trusted Advisor console to access all checks in the Service Limits category and selected checks in the Security and Fault tolerance categories. Automatic check updates aren't available in the Basic and Developer Support plans. You must manually refresh Trusted Advisor checks in the Security category. To manually refresh a check, do the following:

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. Select the **Refresh** button on the check that you want to refresh.

If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, you can use the Trusted Advisor console and the AWS Trusted Advisor API to access all Trusted Advisor checks. You also can use Amazon CloudWatch Events to monitor the status of Trusted Advisor checks. For more information, see Monitoring AWS Trusted Advisor check results with Amazon EventBridge.

You can access Trusted Advisor in the AWS Management Console. For more information about controlling access to the Trusted Advisor console, see Manage access to AWS Trusted Advisor.

For more information, see Trusted Advisor.

Topics

- Opt in AWS Compute Optimizer for Trusted Advisor checks
- Get started with AWS Trusted Advisor Priority
- AWS Trusted Advisor check reference
- Change log for AWS Trusted Advisor

Get started with Trusted Advisor Recommendations

You can use the Trusted Advisor Recommendations page of the Trusted Advisor console to review check results for your AWS account and then follow the recommended steps to fix any issues. For example, Trusted Advisor might recommend that you delete unused resources to reduce your monthly bill, such as an Amazon Elastic Compute Cloud (Amazon EC2) instance.

You can also use the AWS Trusted Advisor API to perform operations on your Trusted Advisor checks. For more information, see the AWS Trusted Advisor API Reference

Topics

- Sign in to the Trusted Advisor console
- View check categories
- View specific checks
- Filter your checks
- Refresh check results
- Download check results
- Organizational view
- Preferences

Sign in to the Trusted Advisor console

You can view the checks and the status of each check in the Trusted Advisor console.

> **Note**
>
> You must have AWS Identity and Access Management (IAM) permissions to access the Trusted Advisor console. For more information, see Manage access to AWS Trusted Advisor.

To sign in to the Trusted Advisor console

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. On the **Trusted Advisor Recommendations** page, view the summary for each check category:
   - **Action recommended (red)** – Trusted Advisor recommends an action for the check. For example, a check that detects a security issue for your IAM resources might recommend urgent steps.
   - **Investigation recommended (yellow)** – Trusted Advisor detects a possible issue for the check. For example, a check that reaches a quota for a resource might recommend ways to delete unused resources.
   - **Checks with excluded items (gray)** – The number of checks that have excluded items, such as resources that you want a check to ignore. For example, this might be Amazon EC2 instances that you don't want the check to evaluate.
3. You can do the following on the **Trusted Advisor Recommendations** page:
   - To refresh all checks in your account, choose **Refresh all checks**.
   - To create an .xls file that includes all check results, choose **Download all checks**.
   - Under **Checks summary**, choose a check category, such as **Security**, to view the results.
   - Under **Potential monthly savings**, you can view how much you can save for your account and the cost optimization checks for recommendations.
   - Under **Recent changes**, you can view changes to check statuses within the last 30 days. Choose a check name to view the latest results for that check or choose the arrow icon to view the next page.

View check categories

You can view the check descriptions and results for the following check categories:

- **Cost optimization** – Recommendations that can potentially save you money. These checks highlight unused resources and opportunities to reduce your bill.
- **Performance** – Recommendations that can improve the speed and responsiveness of your applications.
- **Security** – Recommendations for security settings that can make your AWS solution more secure.

- **Fault tolerance** – Recommendations that help increase the resiliency of your AWS solution. These checks highlight redundancy shortfalls and overused resources.
- **Service limits** – Checks the usage for your account and whether your account approaches or exceeds the limit (also known as quotas) for AWS services and resources.
- **Operational Excellence** – Recommendations to help you operate your AWS environment effectively, and at scale.

To view check categories

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. In the navigation pane, choose the check category.
3. On the category page, view the summary for each check category:
   - **Action recommended (red)** – Trusted Advisor recommends an action for the check.
   - **Investigation recommended (yellow)** – Trusted Advisor detects a possible issue for the check.
   - **No problems detected (green)** – Trusted Advisor doesn't detect an issue for the check.
   - **Excluded items (gray)** – The number of checks that have excluded items, such as resources that you want a check to ignore.
4. For each check, choose the refresh icon
   ( )
   to refresh this check.
5. Choose the download icon
   ( )
   to create an .xls file that includes the results for this check.

View specific checks

Expand a check to view the full check description, your affected resources, any recommended steps, and links to more information.

To view a specific check

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. In the navigation pane, choose a check category.
3. Choose the check name to view the description and the following details:
   - **Alert Criteria** – Describes the threshold when a check will change status.
   - **Recommended Action** – Describes the recommended actions for this check.
   - **Additional Resources** – Lists related AWS documentation.
   - A table that lists the affected items in your account. You can include or exclude these items from check results.
4. (Optional) To exclude items so that they don't appear in check results:
   a. Select an item and choose **Exclude & Refresh**.
   b. To view all excluded items, choose **Excluded items**.
5. (Optional) To include items so that the check evaluates them again:
   a. Choose **Excluded items**, select an item, and then choose **Include & Refresh**.
   b. To view all included items, choose **Included items**.
6. Choose the settings icon ().
   In the **Preferences** dialog box, you can specify the number of items or the properties to display, and then choose **Confirm**.

Filter your checks

On the check category pages, you can specify which check results that you want to view. For example, you might filter by checks that have detected errors in your account so that you can investigate urgent issues first.

If you have checks that evaluate items in your account, such as AWS resources, you can use tag filters to only show items that have the specified tag.

To filter your checks

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. In the navigation pane or the **Trusted Advisor Recommendations** page, choose the check category.
3. For **Search by keyword**, enter a keyword from the check name or description to filter your results.
4. For the **View** list, specify which checks to view:
   - **All checks** – List all checks for this category.
   - **Action recommended** – List checks that recommend that you take action. These checks are highlighted in red.
   - **Investigation recommended** – List checks that recommend that you take possible action. These checks are highlighted in yellow.
   - **No problems detected** – List checks that don't have any issues. These checks are highlighted in green.
   - **Checks with excluded items** – List checks that you specified to exclude items from the check results.
5. If you added tags to your AWS resources, such as Amazon EC2 instances or AWS CloudTrail trails, you can filter your results so that the checks only show items that have the specified tag.

   For **Filter by tag**, enter a tag key and value, and then choose **Apply filter**.
6. In the table for the check, the check results only show items that have the specified key and value.
7. To clear the filter by tags, choose **Reset**.

Related information

For more information about tagging for Trusted Advisor, see the following topics:

- AWS Support enables tagging capabilities for Trusted Advisor
- Tagging AWS resources in the *AWS General Reference*

Refresh check results

You can refresh checks to get the latest results for your account. If you have a Developer or Basic Support plan, you can sign in to the Trusted Advisor console to refresh the checks. If you have an AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, Trusted Advisor automatically refreshes the checks in your account on a weekly basis.

To refresh Trusted Advisor checks

1. Navigate to the AWS Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor.
2. On the **Trusted Advisor Recommendations** or a check category page, choose **Refresh all checks**.

You can also refresh specific checks in the following ways:

- Choose the refresh icon () for an individual check.
- Use the RefreshTrustedAdvisorCheck API operation.

Notes

- Trusted Advisor automatically refreshes some checks several times a day, such as the **AWS Well-Architected high risk issues for reliability** check. It might take a few hours for changes to appear in your account. For these automatically refreshed checks, you can't choose the refresh icon () to manually refresh your results.
- If you enabled AWS Security Hub CSPM for your account, you can't use the Trusted Advisor console to refresh Security Hub CSPM controls. For more information, see Refresh your Security Hub CSPM findings.

Download check results

You can download check results to get an overview of Trusted Advisor in your account. You can download results for all checks or a specific check.

To download check results from Trusted Advisor Recommendations

1. Navigate to the AWS Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor.
   - To download all check results, in the **Trusted Advisor Recommendations** or a check category page, choose **Download all checks**.
   - To download a check result for a specific check, choose the check name, and then choose the download icon ().
2. Save or open the .xls file. The file contains the same summary information from the Trusted Advisor console, such as the check name, description, status, affected resources, and so on.

Organizational view

You can set up the organizational view feature to create a report for all member accounts in your AWS organization. For more information, see Organizational view for AWS Trusted Advisor.

Preferences

On the **Manage Trusted Advisor** page, you can disable Trusted Advisor.

On the **Notifications** page, you can configure your weekly email messages for the check summary. See Set up notification preferences.

On the **Your organization** page, you can enable or disable trusted access with AWS Organizations. This is required for the Organizational view for AWS Trusted Advisor feature and Trusted Advisor Priority.

Set up notification preferences

Specify who can receive the weekly Trusted Advisor email messages for check results and the language. You receive an email notification about your check summary for Trusted Advisor Recommendations once a week.

The email notifications for Trusted Advisor Recommendations don't include results for Trusted Advisor Priority. For more information, see Manage Trusted Advisor Priority notifications.

To set up notification preferences

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. In the navigation pane, under **Preferences**, choose **Notifications**.
3. For **Recommendations**, select whom to notify for your check results. You can add and remove contacts from the Account Settings page in the AWS Billing and Cost Management console.
4. For **Language**, choose the language for the email message.
5. Choose **Save your preferences**.

Set up organizational view

If you set up your account with AWS Organizations, you can create reports for all member accounts in your organization. For more information, see Organizational view for AWS Trusted Advisor.

Disable Trusted Advisor

When you disable this service, Trusted Advisor won't perform any checks on your account. Anyone who tries to access the Trusted Advisor console or use the API operations will receive an access denied error message.

To disable Trusted Advisor

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. In the navigation pane, under **Preferences**, choose **Manage Trusted Advisor**.
3. Under **Trusted Advisor**, turn off **Enabled**. This action disables Trusted Advisor for all checks in your account.
4. You can then manually delete the from your account. For more information, see Deleting a service-linked role for Trusted Advisor.

Related information

For more information about Trusted Advisor, see the following topics:

- How do I start using Trusted Advisor?
- AWS Trusted Advisor check reference

Get started with the Trusted Advisor API

The AWS Trusted Advisor API Reference is intended for programmers that need detailed information about the Trusted Advisor API operations and data types. This API provides access to Trusted Advisor recommendations for your account or all the accounts within your AWS Organization. The Trusted Advisor API uses HTTP methods that returns results in JSON format.

Note

- You must have an AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan to use the Trusted Advisor API.
- If you call the AWS Trusted Advisor API from an account that doesn't have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, then you receive an Access Denied exception. For more information about changing your support plan, see AWS Support.

You can use the AWS Trusted Advisor API to get a list of checks and their descriptions, recommendations, and resources for recommendations. You can also update the lifecycle of recommendations. To manage recommendations, use the following API operations:

- Use the ListChecks, ListRecommendations, GetRecommendation, and ListRecommendationResources API operations to view recommendations and corresponding accounts and resources.
- Use The UpdateRecommendationLifecycle API operation to update the lifecycle of a recommendation that's managed by Trusted Advisor Priority.
- Use The BatchUpdateRecommendationResourceExclusion API operation to include or exclude one or more resources from your Trusted Advisor results.
- The ListOrganizationRecommendations, GetOrganizationRecommendation, ListOrganizationRecommendationResources, ListOrganizationRecommendationAccounts, and UpdateOrganizationRecommendationLifecycle API calls support only recommendations that are managed by Trusted Advisor Priority. These recommendations are also referred to as prioritized recommendations. You can view and manage your prioritized recommendations from

a management or delegated admin account if you have activated Trusted Advisor Priority. If Priority isn't activated, then you receive an Access Denied exception when you make requests.

For more information, see AWS Trusted Advisor in the AWS Support User Guide.

For authentication of requests, see the Signature Version 4 Signing Process.

Using Trusted Advisor as a web service

The AWS Support service enables you to write applications that interact with AWS Trusted Advisor. This topic shows you how to get a list of Trusted Advisor checks, refresh one of them, and then get the detailed results from the check. These tasks are demonstrated in Java. For information about support for other languages, see Tools for Amazon Web Services.

Topics

Get the list of available Trusted Advisor checks

The following Java code snippet creates an instance of an Support client that you can use to call all Trusted Advisor API operations. Next, the code gets the list of Trusted Advisor checks and their corresponding `CheckId` values by calling the DescribeTrustedAdvisorChecks API operation. You can use this information to build user interfaces that enable users to select the check they want to run or refresh.

```
private static AWSSupport createClient()
{
    return AWSSupportClientBuilder.defaultClient();
}
// Get the List of Available Trusted Advisor Checks
public static void getTAChecks() {
    // Possible language parameters: "en" (English), "ja" (Japanese), "fr" (French),
 "zh" (Chinese)
```

```
    DescribeTrustedAdvisorChecksRequest request = new
 DescribeTrustedAdvisorChecksRequest().withLanguage("en");
    DescribeTrustedAdvisorChecksResult result =
 createClient().describeTrustedAdvisorChecks(request);
    for (TrustedAdvisorCheckDescription description : result.getChecks()) {
        // Do something with check description.
        System.out.println(description.getId());
        System.out.println(description.getName());
    }
}
```

Refresh the list of available Trusted Advisor checks

The following Java code snippet creates an instance of an Support client that you can use to refresh Trusted Advisor data.

```
// Refresh a Trusted Advisor Check
// Note: Some checks are refreshed automatically, and they cannot be refreshed by using
 this operation.
// Specifying the check ID of a check that is automatically refreshed causes an
 InvalidParameterValue error.
public static void refreshTACheck(final String checkId) {
    RefreshTrustedAdvisorCheckRequest request = new
 RefreshTrustedAdvisorCheckRequest().withCheckId(checkId);
    RefreshTrustedAdvisorCheckResult result =
 createClient().refreshTrustedAdvisorCheck(request);
    System.out.println("CheckId: " + result.getStatus().getCheckId());
    System.out.println("Milliseconds until refreshable: " +
 result.getStatus().getMillisUntilNextRefreshable());
    System.out.println("Refresh Status: " + result.getStatus().getStatus());
}
```

Poll a Trusted Advisor check for status changes

After you submit the request to run a Trusted Advisor check to generate the latest status data, you use the DescribeTrustedAdvisorCheckRefreshStatuses API operation to request the progress of the check's run, and when new data is ready for the check.

The following Java code snippet gets the status of the check requested in the following section, using the value corresponding in the `CheckId` variable. In addition, the code demonstrates several other uses of the Trusted Advisor service:

1. You can call `getMillisUntilNextRefreshable` by traversing the objects contained in the `DescribeTrustedAdvisorCheckRefreshStatusesResult` instance. You can use the value returned to test whether you want your code to proceed with refreshing the check.
2. If `timeUntilRefreshable` equals zero, you can request a refresh of the check.
3. Using the status returned, you can continue to poll for status changes; the code snippet sets the polling interval to a recommended ten seconds. If the status is either `enqueued` or `in_progress`, the loop returns and requests another status. If the call returns `successful`, the loop terminates.
4. Finally, the code returns an instance of a `DescribeTrustedAdvisorCheckResultResult` data type that you can use to traverse the information produced by the check.

Note: Use a single refresh request before polling for the status of the request.

```
// Retrieves TA refresh statuses. Multiple checkId's can be submitted.
public static List<TrustedAdvisorCheckRefreshStatus> getTARefreshStatus(final String...
 checkIds) {
    DescribeTrustedAdvisorCheckRefreshStatusesRequest request =
            new
 DescribeTrustedAdvisorCheckRefreshStatusesRequest().withCheckIds(checkIds);
    DescribeTrustedAdvisorCheckRefreshStatusesResult result =
            createClient().describeTrustedAdvisorCheckRefreshStatuses(request);
    return result.getStatuses();
}
// Retrieves a TA check status, and checks to see if it has finished processing.
public static boolean isTACheckStatusInTerminalState(final String checkId) {
    // Since we only submitted one checkId to getTARefreshStatus, just retrieve the
 only element in the list.
    TrustedAdvisorCheckRefreshStatus status = getTARefreshStatus(checkId).get(0);
    // Valid statuses are:
    // 1. "none", the check has never been refreshed before.
    // 2. "enqueued", the check is waiting to be processed.
    // 3. "processing", the check is in the midst of being processed.
    // 4. "success", the check has succeeded and finished processing - refresh data is
 available.
    // 5. "abandoned", the check has failed to process.
    return status.getStatus().equals("abandoned") ||
 status.getStatus().equals("success");
}
// Enqueues a Trusted Advisor check refresh. Periodically polls the check refresh
 status for completion.
```

```
public static TrustedAdvisorCheckResult getFreshTACheckResult(final String checkId)
 throws InterruptedException {
    refreshTACheck(checkId);
    while(!isTACheckStatusInTerminalState(checkId)) {
        Thread.sleep(10000);
    }
    return getTACheckResult(checkId);
}
// Retrieves fresh TA check data whenever possible.
// Note: Some checks are refreshed automatically, and they cannot be refreshed by using
 this operation. This method
// is only functional for checks that can be refreshed using the
 RefreshTrustedAdvisorCheck operation.
public static void pollForTACheckResultChanges(final String checkId) throws
 InterruptedException {
    String checkResultStatus = null;
    do {
        TrustedAdvisorCheckResult result = getFreshTACheckResult(checkId);
        if (checkResultStatus != null && !checkResultStatus.equals(result.getStatus()))
 {
            break;
        }
        checkResultStatus = result.getStatus();
        // The rule refresh has completed, but due to throttling rules the checks may
 not be refreshed again
        // for a short period of time.
        // Since we only submitted one checkId to getTARefreshStatus, just retrieve the
 only element in the list.
        TrustedAdvisorCheckRefreshStatus refreshStatus =
 getTARefreshStatus(checkId).get(0);
        Thread.sleep(refreshStatus.getMillisUntilNextRefreshable());
    } while(true);
    // Signal that a TA check has changed check result status here.
}
```

Request a Trusted Advisor check result

After you select the check for the detailed results that you want, you submit a request by using the DescribeTrustedAdvisorCheckResult API operation.

> **Tip**
>
> The names and descriptions for Trusted Advisor checks are subject to change. We recommend that you specify the check ID in your code to uniquely identify a check. You can use the DescribeTrustedAdvisorChecks API operation to get the check ID.

The following Java code snippet uses the `DescribeTrustedAdvisorChecksResult` instance referenced by the variable `result`, which was obtained in the preceding code snippet. Rather than defining a check interactively through a user interface, After you submit the request to run the snippet submits a request for the first check in the list to be run by specifying an index value of 0 in each `result.getChecks().get(0)` call. Next, the code defines an instance of `DescribeTrustedAdvisorCheckResultRequest`, which it passes to an instance of `DescribeTrustedAdvisorCheckResultResult` called `checkResult`. You can use the member structures of this data type to view the results of the check.

```
// Request a Trusted Advisor Check Result
public static TrustedAdvisorCheckResult getTACheckResult(final String checkId) {
    DescribeTrustedAdvisorCheckResultRequest request = new
 DescribeTrustedAdvisorCheckResultRequest()
            // Possible language parameters: "en" (English), "ja" (Japanese),
 "fr" (French), "zh" (Chinese)
            .withLanguage("en")
            .withCheckId(checkId);
    DescribeTrustedAdvisorCheckResultResult requestResult =
 createClient().describeTrustedAdvisorCheckResult(request);
    return requestResult.getResult();
}
```

Note: Requesting a Trusted Advisor Check Result doesn't generate updated results data.

Show details of a Trusted Advisor check

The following Java code snippet iterates over the `DescribeTrustedAdvisorCheckResultResult` instance returned in the previous section to get a list of resources flagged by the Trusted Advisor check.

```
// Show ResourceIds for flagged resources.
for (TrustedAdvisorResourceDetail flaggedResource :
```

```
    result1.getResult().getFlaggedResources())
{
    System.out.println(
        "The resource for this ResourceID has been flagged: " +
        flaggedResource.getResourceId());
}
```

Organizational view for AWS Trusted Advisor

Organizational view lets you view Trusted Advisor checks for all accounts in your AWS Organizations. After you enable this feature, you can create reports to aggregate the check results for all member accounts in your organization. The report includes a summary of check results and information about affected resources for each account. For example, you can use the reports to identify which accounts in your organization are using AWS Identity and Access Management (IAM) with the IAM Use check or whether you have recommended actions for Amazon Simple Storage Service (Amazon S3) buckets with the Amazon S3 Bucket Permissions check.

Topics

Prerequisites

You must meet the following requirements to enable organizational view:

- Your accounts must be members of an AWS Organizations.
- Your organization must have all features enabled for Organizations. For more information, see Enabling all features in your organization in the *AWS Organizations User Guide*.

- The management account in your organization must have an AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan. You can find your support plan from the AWS Support Center or from the Support plans page. See Compare AWS Support plans.
- You must sign in as a user in the management account (or assumed equivalent role). Whether you sign in as an IAM user or an IAM role, you must have a policy with the required permissions. See Using IAM policies to allow access to organizational view.

Enable organizational view

After you meet the prerequisites, follow these steps to enable organizational view. After you enable this feature, the following happens:

- Trusted Advisor is enabled as a *trusted service* in your organization. For more information, see Enabling trusted access with other AWS services in the *AWS Organizations User Guide*.
- The `AWSServiceRoleForTrustedAdvisorReporting` service-linked-role is created for you in the management account in your organization. This role includes the permissions that Trusted Advisor needs to call Organizations on your behalf. This service-linked role is locked, and you can't delete it manually. For more information, see Using service-linked roles for Trusted Advisor.

You enable organizational view from the Trusted Advisor console.

To enable organizational view

1. Sign in as an administrator in the organization's management account and open the AWS Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor.
2. In the navigation pane, under **Preferences**, choose **Your organization**.
3. Under **Enable trusted access with AWS Organizations**, turn on **Enabled**.

> **Note**
>
> Enabling organizational view for the management account doesn't provide the same checks for all member accounts. For example, if your member accounts all have Basic Support, those accounts won't have the same checks available as your management account. The AWS Support plan determines which Trusted Advisor checks are available for an account.

Refresh Trusted Advisor checks

Before you create a report for your organization, we recommend that you refresh the statuses of your Trusted Advisor checks. You can download a report without refreshing your Trusted Advisor checks, but your report might not have the latest information.

If you have an AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, Trusted Advisor automatically refreshes the checks in your account on a weekly basis.

> **Note**
>
> If you have accounts in your organization that have a Developer or Basic support plan, a user for those accounts must sign in to the Trusted Advisor console to refresh the checks. You can't refresh checks for all accounts from the organization's management account.

To refresh Trusted Advisor checks

1. Navigate to the AWS Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor.
2. On the **Trusted Advisor Recommendations** page, choose the **Refresh all checks**. This refreshes all checks in your account.

You can also refresh specific checks in the following ways:

- Use the RefreshTrustedAdvisorCheck API operation.
- Choose the refresh icon () for an individual check.

Create organizational view reports

After you enable organizational view, you can create reports so that you can view Trusted Advisor check results for your organization.

You can create up to 50 reports. If you create reports beyond this quota, Trusted Advisor deletes the earliest report. You can't recover deleted reports.

To create organizational view reports

1. Sign in to the organization's management account and open the AWS Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor.
2. In the navigation pane, choose **Organizational View**.
3. Choose **Create report**.
4. By default, the report includes all AWS Regions, check categories, checks, and resource statuses. On the **Create report** page, you can use the filter options to customize your report. For example, you can clear the **All** option for **Region**, and then specify the individual Regions to include in the report.

   a. Enter a **Name** for the report.

   b. For **Format**, choose **JSON** or **CSV**.

   c. For **Region**, specify the AWS Regions or choose **All**.

   d. For **Check category**, choose the check category or choose **All**.

   e. For **Checks**, choose the specific checks for that category or choose **All**.

   > **Note**
   >
   > The **Check category** filter overrides the **Checks** filter. For example, if you choose the **Security** category and then choose a specific check name, your report includes all check results for that category. To create a report for only specific checks, keep the default **All** value for **Check category** and then choose your check names.

   f. For **Resource status**, choose the status to filter, such as **Warning**, or choose **All**.
5. For **AWS Organizations**, select the organizational units (OUs) to include in your report. For more information about OUs, see Managing organizational units in the *AWS Organizations User Guide*.
6. Choose **Create report**.

Example : Create report filter options

The following example creates a JSON report for the following:

- Three AWS Regions
- All **Security** and **Performance** checks

In the following example, the report includes the **support-team** organizational unit and one AWS account that are part of the organization.

Notes

- The amount of time it takes to create the report depends on the number of accounts in the organization and the number of resources in each account.
- You can't create more than one report at a time unless the current report has been running for more than 6 hours.
- Refresh the page if you don't see the report appear on the page.

View the report summary

After the report is ready, you can view the report summary from the Trusted Advisor console. This lets you quickly view the summary of your check results across your organization.

To view the report summary

1. Sign in to the organization's management account and open the AWS Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor.
2. In the navigation pane, choose **Organizational View**.
3. Choose the report name.
4. On the **Summary** page, view the check statuses for each category. You can also choose **Download report**.

Download an organizational view report

After your report is ready, download it from the Trusted Advisor console. The report is a .zip file that contains three files:

- `summary.json` – Contains a summary of the check results for each check category.
- `schema.json` – Contains the schema for the specified checks in the report.
- A resources file (.json or .csv) – Contains detailed information about the check statuses for resources in your organization.

To download an organizational view report

1. Sign in to the organization's management account and open the AWS Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor.
2. In the navigation pane, choose **Organizational View**.

   The **Organizational View** page displays the available reports to download.
3. Select a report, choose **Download report**, and then save the file. You can only download one report at a time.
4. Unzip the file.
5. Use a text editor to open the `.json` file or a spreadsheet application to open the `.csv` file.

> **Note**
>
> You might receive multiple files if your report is 5 MB or larger.

Example : summary.json file

The `summary.json` file shows the number of accounts in the organization and the statuses of the checks in each category.

Trusted Advisor uses the following color code for check results:

- `Green` – Trusted Advisor doesn't detect an issue for the check.
- `Yellow` – Trusted Advisor detects a possible issue for the check.
- `Red` – Trusted Advisor detects an error and recommends an action for the check.
- `Blue` – Trusted Advisor can't determine the status of the check.

In the following example, two checks are `Red`, one is `Green`, and one is `Yellow`.

```
{
    "numAccounts": 3,
    "filtersApplied": {
        "accountIds": ["123456789012","111122223333","111111111111"],
        "checkIds": "All",
        "categories": [
            "security",
```

```
            "performance"
        ],
        "statuses": "All",
        "regions": [
            "us-west-1",
            "us-west-2",
            "us-east-1"
        ],
        "organizationalUnitIds": [
            "ou-xa9c-EXAMPLE1",
            "ou-xa9c-EXAMPLE2"
        ]
    },
    "categoryStatusMap": {
        "security": {
            "statusMap": {
                "ERROR": {
                    "name": "Red",
                    "count": 2
                },
                "OK": {
                    "name": "Green",
                    "count": 1
                },
                "WARN": {
                    "name": "Yellow",
                    "count": 1
                }
            },
            "name": "Security"
        }
    },
    "accountStatusMap": {
        "123456789012": {
            "security": {
                "statusMap": {
                    "ERROR": {
                        "name": "Red",
                        "count": 2
                    },
                    "OK": {
                        "name": "Green",
                        "count": 1
                    },
```

```
                    "WARN": {
                        "name": "Yellow",
                        "count": 1
                    }
                },
                "name": "Security"
            }
        }
    }
}
```

Example : schema.json file

The `schema.json` file includes the schema for the checks in the report. The following example includes the IDs and properties for the IAM Password Policy (Yw2K9puPzl) and IAM Key Rotation (DqdJqYeRm5) checks.

```
{
    "Yw2K9puPzl": [
        "Password Policy",
        "Uppercase",
        "Lowercase",
        "Number",
        "Non-alphanumeric",
        "Status",
        "Reason"
    ],
    "DqdJqYeRm5": [
        "Status",
        "IAM User",
        "Access Key",
        "Key Last Rotated",
        "Reason"
    ],
    ...
}
```

Example

The `resources.csv` file includes information about resources in the organization. This example shows some of the data columns that appear in the report, such as the following:

- Account ID of the affected account

- The Trusted Advisor check ID
- The resource ID
- Timestamp of the report
- The full name of the Trusted Advisor check
- The Trusted Advisor check category
- The account ID of the parent organizational unit (OU) or root

The resources file only contains entries if a check result exists at the resource level. You might not see checks in the report for the following reasons:

- Some checks, such as **MFA on Root Account**, don't have resources and won't appear in the report. Checks without resources appear in the `summary.json` file instead.
- Some checks only show resources if they are `Red` or `Yellow`. If all resources are `Green`, they might not appear in your report.
- If an account isn't enabled for a service that requires the check, the check might not appear in the report. For example, if you're not using Amazon Elastic Compute Cloud Reserved Instances in your organization, the Amazon EC2 Reserved Instance Lease Expiration check won't appear in your report.
- The account hasn't refreshed check results. This might happen when users with a Basic or Developer support plan sign in to the Trusted Advisor console for the first time. If you have an AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, it can take up to one week from account sign up for users to see check results. For more information, see Refresh Trusted Advisor checks.
- If only the organization's management account enabled recommendations for checks, the report won't include resources for other accounts in the organization.

For the resources file, you can use common software such as Microsoft Excel to open the .csv file format. You can use the .csv file for one-time analysis of all checks across all accounts in your organization. If you want to use your report with an application, you can download the report as a .json file instead.

The .json file format provides more flexibility than the .csv file format for advanced use cases such as aggregation and advanced analytics with multiple datasets. For example, you can use a SQL interface with an AWS service such as Amazon Athena to run queries on your reports. You can also

use Amazon Quick Suite to create dashboards and visualize your data. For more information, see Using other AWS services to view Trusted Advisor reports.

Disable organizational view

Follow this procedure to disable organizational view. You must sign in to the organization's management account or assume a role with the required permissions to disable this feature. You can't disable this feature from another account in the organization.

After you disable this feature, the following happens:

- Trusted Advisor is removed as a trusted service in Organizations.
- The `AWSServiceRoleForTrustedAdvisorReporting` service-linked role is unlocked in the organization's management account. This means you can delete it manually, if needed.
- You can't create, view, or download reports for your organization. To access previously created reports, you must reenable organizational view from the Trusted Advisor console. See Enable organizational view.

To disable organizational view for Trusted Advisor

1. Sign in to the organization's management account and open the AWS Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor.
2. In the navigation pane, choose **Preferences**.
3. Under **Organizational View**, choose **Disable organizational view**.

After you disable organizational view, Trusted Advisor no longer aggregates checks from other AWS accounts in your organization. However, the `AWSServiceRoleForTrustedAdvisorReporting` service-linked role remains on the organization's management account until you delete it through the IAM console, IAM API, or AWS Command Line Interface (AWS CLI). For more information, see Deleting a service-linked role in the *IAM User Guide*.

> **Note**
>
> You can use other AWS services to query and visualize your data for organizational view reports. For more information, see the following resources:

- View AWS Trusted Advisor recommendations at scale with AWS Organizations in the *AWS Management & Governance Blog*
- Using other AWS services to view Trusted Advisor reports

Using IAM policies to allow access to organizational view

You can use the following AWS Identity and Access Management (IAM) policies to allow users or roles in your account access to organizational view in AWS Trusted Advisor.

Example : Full access to organizational view

The following policy allows full access to the organizational view feature. A user with these permissions can do the following:

- Enable and disable organizational view
- Create, view, and download reports

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "ReadStatement",
            "Effect": "Allow",
            "Action": [
                "organizations:ListAccountsForParent",
                "organizations:ListAccounts",
                "organizations:ListRoots",
                "organizations:DescribeOrganization",
                "organizations:ListOrganizationalUnitsForParent",
                "organizations:ListAWSServiceAccessForOrganization",
                "trustedadvisor:DescribeAccount",
                "trustedadvisor:DescribeChecks",
                "trustedadvisor:DescribeCheckSummaries",
                "trustedadvisor:DescribeAccountAccess",
                "trustedadvisor:DescribeOrganization",
                "trustedadvisor:DescribeReports",
```

```
                "trustedadvisor:DescribeServiceMetadata",
                "trustedadvisor:DescribeOrganizationAccounts",
                "trustedadvisor:ListAccountsForParent",
                "trustedadvisor:ListRoots",
                "trustedadvisor:ListOrganizationalUnitsForParent"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CreateReportStatement",
            "Effect": "Allow",
            "Action": [
                "trustedadvisor:GenerateReport"
            ],
            "Resource": "*"
        },
        {
            "Sid": "ManageOrganizationalViewStatement",
            "Effect": "Allow",
            "Action": [
                "organizations:EnableAWSServiceAccess",
                "organizations:DisableAWSServiceAccess",
                "trustedadvisor:SetOrganizationAccess"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CreateServiceLinkedRoleStatement",
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
reporting.trustedadvisor.amazonaws.com/AWSServiceRoleForTrustedAdvisorReporting"
        }
    ]
}
```

Example : Read access to organizational view

The following policy allows read-only access to organizational view for Trusted Advisor. A user with these permissions can only view and download existing reports.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "ReadStatement",
            "Effect": "Allow",
            "Action": [
                "organizations:ListAccountsForParent",
                "organizations:ListAccounts",
                "organizations:ListRoots",
                "organizations:DescribeOrganization",
                "organizations:ListOrganizationalUnitsForParent",
                "organizations:ListAWSServiceAccessForOrganization",
                "trustedadvisor:DescribeAccount",
                "trustedadvisor:DescribeChecks",
                "trustedadvisor:DescribeCheckSummaries",
                "trustedadvisor:DescribeAccountAccess",
                "trustedadvisor:DescribeOrganization",
                "trustedadvisor:DescribeReports",
                "trustedadvisor:ListAccountsForParent",
                "trustedadvisor:ListRoots",
                "trustedadvisor:ListOrganizationalUnitsForParent"
            ],
            "Resource": "*"
        }
    ]
}
```

You can also create your own IAM policy. For more information, see Creating IAM Policies in the *IAM User Guide*.

Note

If you enabled AWS CloudTrail in your account, the following roles can appear in your log entries:

- `AWSServiceRoleForTrustedAdvisorReporting` – The service-linked role that Trusted Advisor uses to access accounts in your organization.

- `AWSServiceRoleForTrustedAdvisor` – The service-linked role that Trusted Advisor uses to access services in your organization.

For more information about service-linked roles, see Using service-linked roles for Trusted Advisor.

Using other AWS services to view Trusted Advisor reports

Follow this tutorial to upload and view your data by using other AWS services. In this topic, you create an Amazon Simple Storage Service (Amazon S3) bucket to store your report and an CloudFormation template to create resources in your account. Then, you can use Amazon Athena to analyze or run queries for your report or Quick Suite to visualize that data in a dashboard.

For information and examples for visualizing your report data, see the View AWS Trusted Advisor recommendations at scale with AWS Organizations in the *AWS Management & Governance Blog*.

Prerequisites

Before you start this tutorial, you must meet the following requirements:

- Sign in as an AWS Identity and Access Management (IAM) user with administrator permissions.
- Use the US East (N. Virginia) AWS Region to quickly set up your AWS services and resources.
- Create an Quick Suite account. For more information, see Getting Started with Data Analysis in Quick Suite in the *Amazon Quick Suite User Guide*.

Upload the report to Amazon S3

After you download your `resources.json` report, upload the file to Amazon S3. You must use a bucket in the US East (N. Virginia) Region.

To upload the report to an Amazon S3 bucket

1. Sign in to the AWS Management Console at https://eusc-de-east-1.console.amazonaws-eusc.eu/.
2. Use the **Region selector** and choose the US East (N. Virginia) Region.
3. Open the Amazon S3 console at https://eusc-de-east-1.console.amazonaws-eusc.eu/s3/.

4. From the list of buckets, choose an S3 bucket, and then copy the name. You use the name in the next procedure.
5. On the *bucket-name* page, choose **Create folder**, enter the name **folder1**, and then choose **Save**.
6. Choose the **folder1**.
7. In **folder1**, choose **Upload** and choose the `resources.json` file.
8. Choose **Next**, keep the default options, and then choose **Upload**.

> **Note**
>
> If you upload a new report to this bucket, rename the `.json` files each time you upload them so that you don't override the existing reports. For example, you can add the timestamp to each file, such as `resources-timestamp.json`, `resources-timestamp2.json`, and so on.

Create your resources using AWS CloudFormation

After you upload your report to Amazon S3, upload the following YAML template to CloudFormation. This template tells CloudFormation what resources to create for your account so that other services can use the report data in the S3 bucket. The template creates resources for IAM, AWS Lambda, and AWS Glue.

To create your resources with CloudFormation

1. Download the trusted-advisor-reports-template.zip file.
2. Unzip the file.
3. Open the template file in a text editor.
4. For the `BucketName` and `FolderName` parameters, replace the values for *your-bucket-name-here* and *folder1* with the bucket name and folder name in your account.
5. Save the file.
6. Open the CloudFormation console at https://eusc-de-east-1.console.amazonaws-eusc.eu/cloudformation.
7. If you haven't already, in the **Region selector**, choose the US East (N. Virginia) Region.
8. In the navigation pane, choose **Stacks**.

9. Choose **Create stack** and choose **With new resources (standard)**.
10. On the **Create stack** page, under **Specify template**, choose **Upload a template file**, and then choose **Choose file**.
11. Choose the YAML file and choose **Next**.
12. On the **Specify stack details** page, enter a stack name such as **`Organizational-view-Trusted-Advisor-reports`**, and choose **Next**.
13. On the **Configure stack options** page, keep the default options, and then choose **Next**.
14. On the **Review `Organizational-view-Trusted-Advisor-reports`** page, review your options. At the bottom of the page, select the check box for **I acknowledge that CloudFormation might create IAM resources**.
15. Choose **Create stack**.

    The stack takes about 5 minutes to create.
16.

Query the data in Amazon Athena

After you have your resources, you can view the data in Athena. Use Athena to create queries and analyze the results of the report, such as looking up specific check results for accounts in the organization.

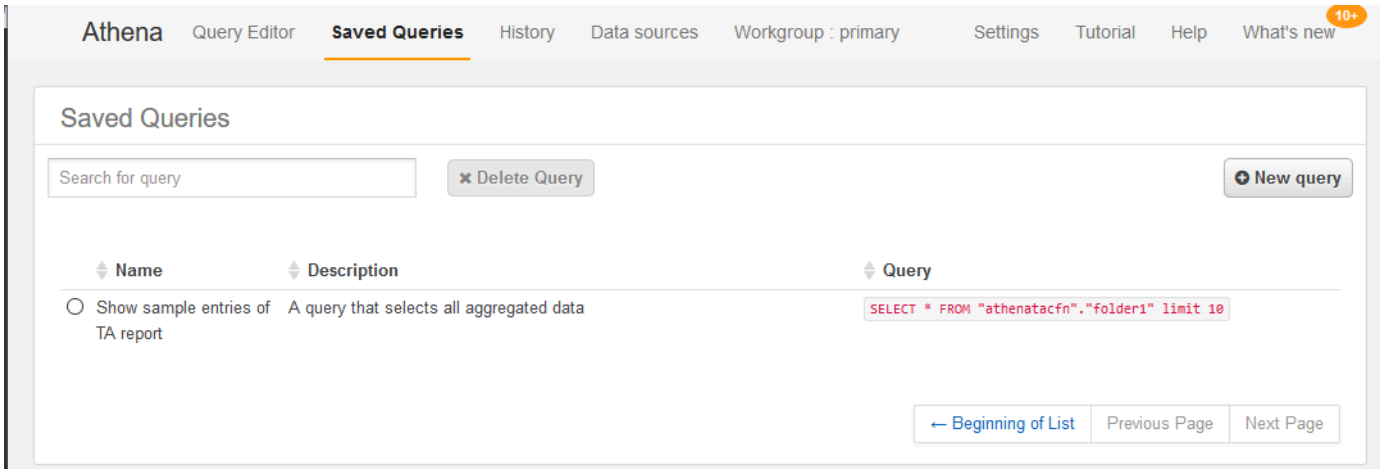
Notes

- Use the US East (N. Virginia) Region.
- If you're new to Athena, you must specify a query result location before you can run a query for your report. We recommend that you specify a different S3 bucket for this location. For more information, see Specifying a query result location in the *Amazon Athena User Guide*.

To query the data in Athena

1. Open the Athena console at https://eusc-de-east-1.console.amazonaws-eusc.eu/athena/.
2. If you haven't already, in the **Region selector**, choose the US East (N. Virginia) Region.
3. Choose **Saved Queries** and in search field, enter **`Show sample`**.

4. Choose the query that appears, such as **Show sample entries of TA report**.



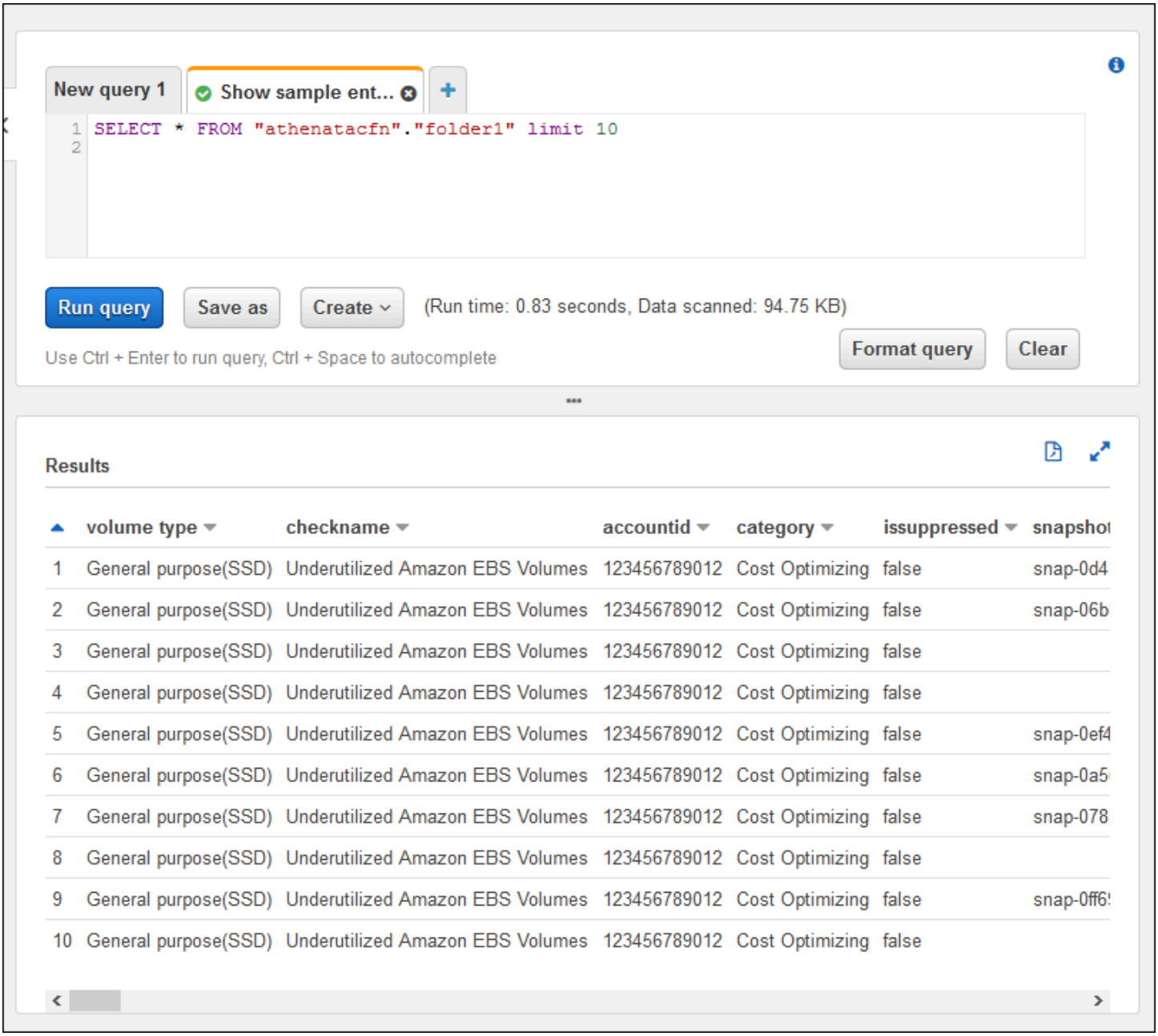
   The query should look like the following.

```
SELECT * FROM "athenatacfn"."folder1" limit 10
```

5. Choose **Run query**. Your query results appear.

Example : Athena query

The following example shows 10 sample entries from the report.



	volume type	checkname	accountid	category	issuppressed	snapshot
1	General purpose(SSD)	Underutilized Amazon EBS Volumes	123456789012	Cost Optimizing	false	snap-0d4
2	General purpose(SSD)	Underutilized Amazon EBS Volumes	123456789012	Cost Optimizing	false	snap-06b
3	General purpose(SSD)	Underutilized Amazon EBS Volumes	123456789012	Cost Optimizing	false	
4	General purpose(SSD)	Underutilized Amazon EBS Volumes	123456789012	Cost Optimizing	false	
5	General purpose(SSD)	Underutilized Amazon EBS Volumes	123456789012	Cost Optimizing	false	snap-0ef4
6	General purpose(SSD)	Underutilized Amazon EBS Volumes	123456789012	Cost Optimizing	false	snap-0a5
7	General purpose(SSD)	Underutilized Amazon EBS Volumes	123456789012	Cost Optimizing	false	snap-078
8	General purpose(SSD)	Underutilized Amazon EBS Volumes	123456789012	Cost Optimizing	false	
9	General purpose(SSD)	Underutilized Amazon EBS Volumes	123456789012	Cost Optimizing	false	snap-0ff6
10	General purpose(SSD)	Underutilized Amazon EBS Volumes	123456789012	Cost Optimizing	false	

For more information, see Running SQL Queries Using Amazon Athena in the *Amazon Athena User Guide*.

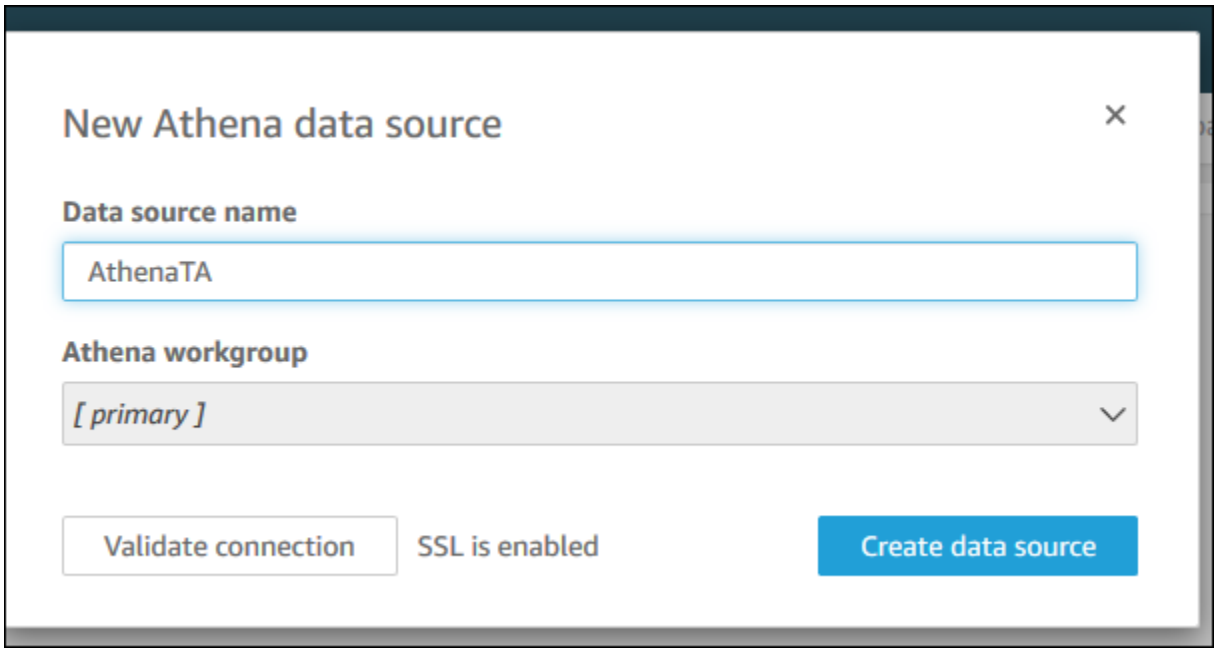
Create a dashboard in Quick Suite

You can also set up Quick Suite so that you can view your data in a dashboard and visualize your report information.

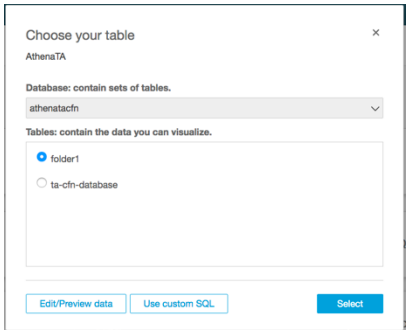
> **Note**
>
> You must use the US East (N. Virginia) Region.

To create a dashboard in Quick Suite

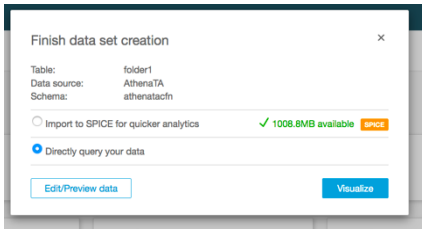
1. Navigate to the Quick Suite console and sign in to your account.
2. Choose **New analysis**, **New dataset**, and then choose **Athena**.
3. In the **New Athena data source** dialog box, enter a data source name such as **AthenaTA**, and then choose **Create data source**.



4. In the **Choose your table** dialog box, choose the **athenatacfn** table, choose **folder1**, and then choose **Select**.



5. In the **Finish data set creation** dialog box, choose **Directly query your data**, and then choose **Visualize**.

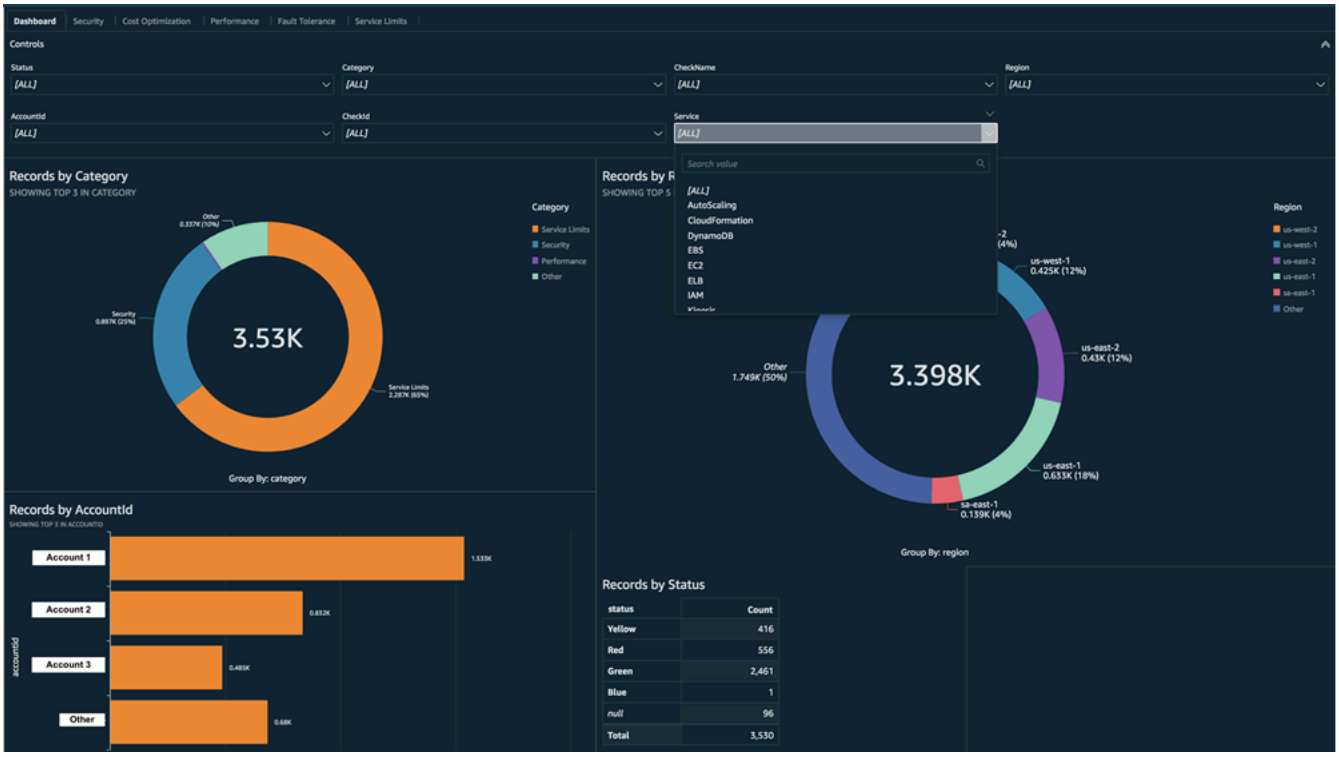


You can now create a dashboard in Quick Suite. For more information, see Working with Dashboards in the *Amazon Quick Suite User Guide*.

Example : Quick Suite dashboard

The following example dashboard shows information about the Trusted Advisor checks, such as the following:

- Affected account IDs
- Summary by AWS Regions
- Check categories
- Check statuses
- Number of entries in the report for each account



> **Note**
>
> If you have permission errors while creating your dashboard, make sure that Quick Suite can use Athena. For more information, see I Can't Connect to Amazon Athena in the *Amazon Quick Suite User Guide*.

For more information and examples for visualizing your report data, see the View AWS Trusted Advisor recommendations at scale with AWS Organizations in the *AWS Management & Governance Blog*.

Troubleshooting

If you have issues with this tutorial, see the following troubleshooting tips.

I'm not seeing the latest data in my report

When you create a report, the organizational view feature doesn't automatically refresh the Trusted Advisor checks in your organization. To get the latest check results, refresh the checks for the management account and each member account in the organization. For more information, see Refresh Trusted Advisor checks.

I have duplicate columns in the report

The Athena console might show the following error in your table if your report has duplicate columns.

```
HIVE_INVALID_METADATA: Hive metadata for table folder1 is invalid: Table descriptor contains duplicate columns
```

For example, if you added a column in your report that already exists, this can cause issues when you try to view the report data in the Athena console. You can follow these steps to fix this issue.

Find duplicate columns

You can use the AWS Glue console to view the schema and quickly identify if you have duplicate columns in your report.

To find duplicate columns

1. Open the AWS Glue console at https://eusc-de-east-1.console.amazonaws-eusc.eu/glue/.

2. If you haven't already, in the **Region selector**, choose the US East (N. Virginia) Region.
3. In the navigation pane, choose **Tables**.
4. Choose your folder name, such as ***`folder1`***, and then under **Schema**, view the values for **Column name**.

   If you have a duplicate column, you must upload a new report to your Amazon S3 bucket. See the following Upload a new report section.

Upload a new report

After you identify the duplicate column, we recommend that you replace the existing report with a new one. This ensures that the resources created from this tutorial use the latest report data from your organization.

To upload a new report

1. If you haven't already, refresh your Trusted Advisor checks for the accounts in your organization. See Refresh Trusted Advisor checks.
2. Create and download another JSON report in the Trusted Advisor console. See Create organizational view reports. You must use a JSON file for this tutorial.
3. Sign in to the AWS Management Console and open the Amazon S3 console at https://eusc-de-east-1.console.amazonaws-eusc.eu/s3/.
4. Choose your Amazon S3 bucket and choose the *`folder1`* folder.
5. Select the previous *`resources`*`.json` reports and choose **Delete**.
6. In the **Delete objects** page, under **Permanently delete objects?**, enter **`permanently delete`**, and then choose **Delete objects**.
7. In your S3 bucket, choose **Upload** and then specify the new report. This action automatically updates your Athena table and AWS Glue crawler resources with the latest report data. It can take a few minutes to refresh your resources.
8. Enter a new query in the Athena console. See Query the data in Amazon Athena.

> **Note**
>
> If you still have issues with this tutorial, you can create a technical support case in the AWS Support Center.

View AWS Trusted Advisor checks powered by AWS Config

AWS Config is a service that continually assesses, audits, and evaluates your resource configurations for your desired settings. AWS Config provides managed rules, which are predefined, customizable compliance checks that AWS Config uses to evaluate if your AWS resources comply with common best practices.

The AWS Config console guides you through the configuration and activation of managed rules. You can also use the AWS Command Line Interface (AWS CLI) or AWS Config API to pass the JSON code that defines your configuration of a managed rule. You can customize the behavior of a managed rule to suit your needs. You can customize the rule's parameters to define attributes that your resources must have to comply with the rule. To learn more about enabling AWS Config, see the AWS Config Developer Guide.

AWS Config managed rules power a set of Trusted Advisor checks across all categories. When you enable certain managed rules, the corresponding Trusted Advisor checks are automatically enabled. To see which Trusted Advisor checks are powered by specific AWS Config managed rules, see AWS Trusted Advisor check reference.

The AWS Config powered checks are available to customers with an AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan. If you enable AWS Config and you have one of these AWS Support plans, then you automatically see recommendations powered by corresponding deployed AWS Config managed rules.

> **Note**
>
> Results for these checks are automatically refreshed based on change-triggered updates to AWS Config managed rules. Refresh requests are not allowed. Currently, you can’t exclude resources from these checks.

Troubleshooting

If you have issues with this integration, see the following troubleshooting information.

Contents

- I deployed the same AWS Config managed rule twice, what will I see in Trusted Advisor?
- I turned off recording for AWS Config in an AWS Region. What will I see in Trusted Advisor?

I just enabled recording and managed rules for AWS Config, but I don't see corresponding Trusted Advisor checks.

After the AWS Config rule generates evalution results, you see the results in Trusted Advisor in near real-time. If you have issues with this feature, create a technical support case in the AWS Support Center.

I deployed the same AWS Config managed rule twice, what will I see in Trusted Advisor?

You see separate entries in the Trusted Advisor check results for each managed rule that you install.

I turned off recording for AWS Config in an AWS Region. What will I see in Trusted Advisor?

If you turned off resource recording for AWS Config in an AWS Region, then Trusted Advisor no longer receives data for corresponding managed rules and checks in that Region. Existing managed rule results remain in AWS Config and in Trusted Advisor until AWS Config expires, based on the recorder retention policy. If you delete a managed rule, then the Trusted Advisor check data usually deletes in near real-time.

Viewing AWS Security Hub CSPM controls in AWS Trusted Advisor

After you enable AWS Security Hub CSPM for your AWS account, you can view your security controls and their findings in the Trusted Advisor console. You can use Security Hub CSPM controls to identify security vulnerabilities in your account in the same way that you can use Trusted Advisor checks. You can view the check's status, the list of affected resources, and then follow Security Hub CSPM recommendations to address your security issues. You can use this feature to find security recommendations from Trusted Advisor and Security Hub CSPM in one convenient location.

Notes

- From Trusted Advisor, you can view controls in the AWS Foundational Security Best Practices security standard *except* for controls that have the Category: Recover > Resilience. For a list of supported controls, see AWS Foundational Security Best Practices controls in the *AWS Security Hub CSPM User Guide*.

  For more information about the Security Hub CSPM categories, see Control categories.
- Trusted Advisor onboarded Security Hub CSPM controls up to September 26, 2024. Controls released after September 26, 2024 are not yet onboarded to Trusted Advisor. You can find controls released after that date in the Security Hub CSPM log.

Topics

Prerequisites

You must meet the following requirements to enable the Security Hub CSPM integration with Trusted Advisor:

- You must have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan for this feature. You can find your support plan from the AWS Support Center or from the Support plans page. For more information, see Compare AWS Support plans.
- You must enable resource recording in AWS Config for the AWS Regions that you want for your Security Hub CSPM controls. For more information, see Enabling and configuring AWS Config.
- You must enable Security Hub CSPM and select the **AWS Foundational Security Best Practices v1.0.0** security standard. If you haven't done so already, see Setting up AWS Security Hub CSPM in the *AWS Security Hub CSPM User Guide*.

> **Note**
>
> If you already completed these prerequisites, you can skip to View your Security Hub CSPM findings.

About AWS Organizations accounts

If you already completed the prerequisites for a management account, this integration is enabled automatically for all member accounts in your organization. Individual member accounts don't need to contact Support to enable this feature. However, member accounts in your organization must enable Security Hub CSPM if they want to see their findings in Trusted Advisor.

If you want to disable this integration for a specific member account, see Disable this feature for AWS Organizations accounts.

View your Security Hub CSPM findings

After you enable Security Hub CSPM for your account, it can take up to 24 hours for your Security Hub CSPM findings to appear in the **Security** page of the Trusted Advisor console.

To view your Security Hub CSPM findings in Trusted Advisor

1. Navigate to the Trusted Advisor console, and then choose the **Security** category.
2. In the **Search by keyword** field, enter the control name or description in the field.

   > **Tip**
   >
   > For **Source**, you can choose **AWS Security Hub CSPM** to filter for Security Hub CSPM controls.

3. Choose the Security Hub CSPM control name to view the following information:

   - **Description** – Describes how this control checks your account for security vulnerabilities.
   - **Source** – Whether the check comes from AWS Trusted Advisor or AWS Security Hub CSPM. For Security Hub CSPM controls, you can find the control ID.
   - **Alert Criteria** – The status of the control. For example, if Security Hub CSPM detects an important issue, the status might be **Red: Critical or High**.

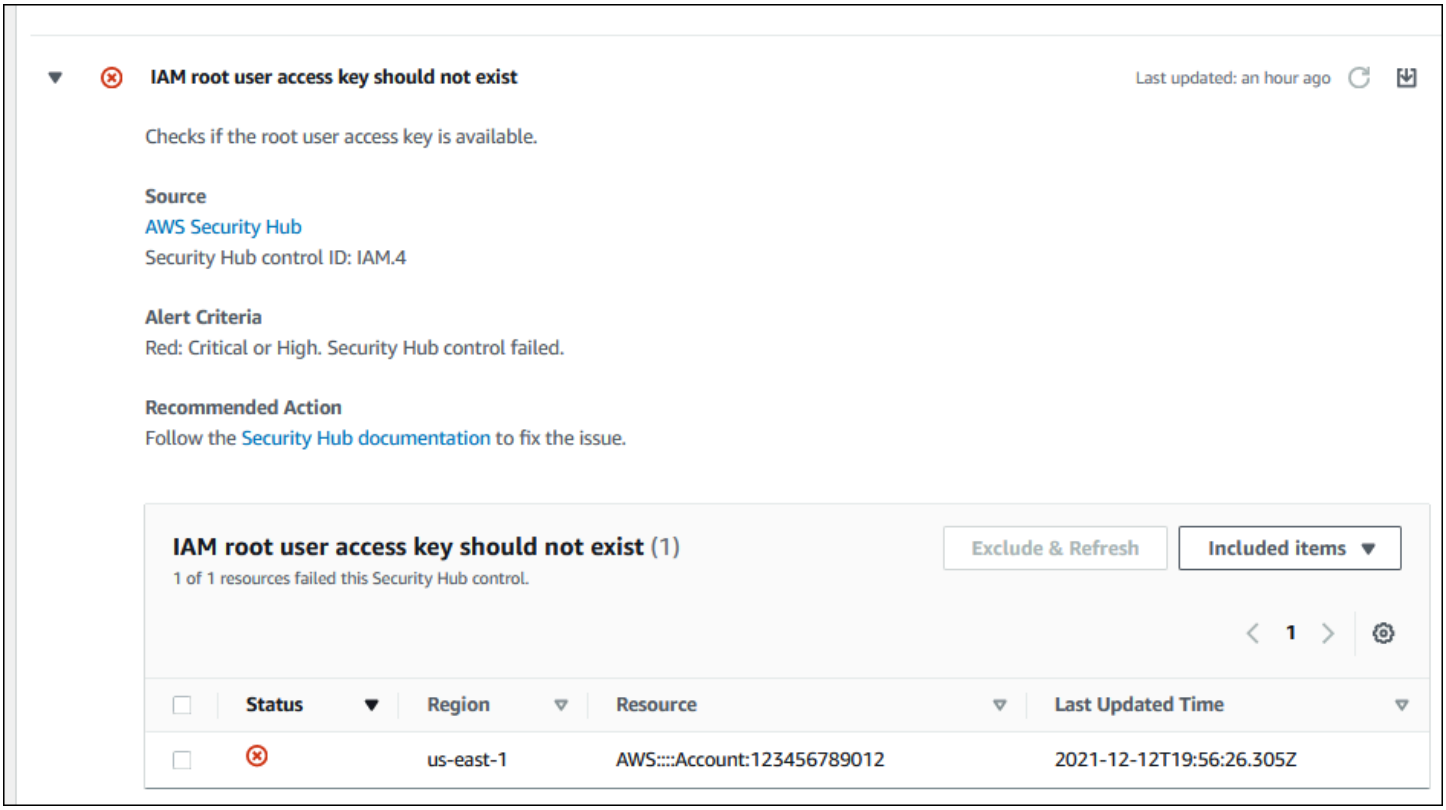
- **Recommended Action** – Use the Security Hub CSPM documentation link to find the recommended steps to fix the issue.
- **Security Hub CSPM resources** – You can find the resources in your account where Security Hub CSPM has detected an issue.

Notes

- You must use Security Hub CSPM to exclude resources from your findings. Currently, you can't use the Trusted Advisor console to exclude items from Security Hub CSPM controls. For more information, see Setting the workflow status for findings.
- The organizational view feature supports this integration with Security Hub CSPM. You can view your findings for your Security Hub CSPM controls across your organization, and then create and download reports. For more information, see Organizational view for AWS Trusted Advisor.

Example Example : Security Hub CSPM control for IAM user access key should not exist

The following is an example finding for a Security Hub CSPM control in the Trusted Advisor console.



Refresh your Security Hub CSPM findings

After you enable a security standard, it can take up to two hours for Security Hub CSPM to have findings for your resources. It can then take up to 24 hours for that data to appear in the Trusted Advisor console. If you recently enabled the **AWS Foundational Security Best Practices v1.0.0** security standard, check the Trusted Advisor console again later.

Note

- The refresh schedule for each Security Hub CSPM control is *periodic* or *change triggered*. Currently, you can't use the Trusted Advisor console or the AWS Support API to refresh your Security Hub CSPM controls. For more information, see Schedule for running security checks.
- You must use Security Hub CSPM if you want to exclude resources from your findings. Currently, you can't use the Trusted Advisor console to exclude items from Security Hub CSPM controls. For more information, see Setting the workflow status for findings.

Disable Security Hub CSPM from Trusted Advisor

Follow this procedure if you don't want your Security Hub CSPM information to appear in the Trusted Advisor console. This procedure only disables the Security Hub CSPM integration with Trusted Advisor. It won't affect your configurations with Security Hub CSPM. You can continue to use the Security Hub CSPM console to view your security controls, resources, and recommendations.

To disable the Security Hub CSPM integration

1. Contact AWS Support and request to disable the Security Hub CSPM integration with Trusted Advisor.

   After AWS Support disables this feature, Security Hub CSPM no longer sends data to Trusted Advisor. Your Security Hub CSPM data will be removed from Trusted Advisor.
2. If you want to enable this integration again, contact AWS Support.

Disable this feature for AWS Organizations accounts

If you already completed the previous procedure for a management account, Security Hub CSPM integration is automatically removed from all member accounts in your organization. Individual member accounts in your organization don't need to contact AWS Support separately.

If you're a member account in an organization, you can contact Support to remove this feature from only your account.

Troubleshooting

If you're having issues with this integration, see the following troubleshooting information.

Contents

- I see multiple AWS Regions for the same affected resource for a Security Hub CSPM check
- I turned off Security Hub CSPM or AWS Config in a Region
- My control is archived in Security Hub CSPM, but I still see the findings in Trusted Advisor
- I still can't view my Security Hub CSPM findings

I don't see Security Hub CSPM findings in the Trusted Advisor console

Verify that you completed the following steps:

- You have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan.
- You enabled resource recording in AWS Config within the same Region as Security Hub CSPM.
- You enabled Security Hub CSPM and selected the **AWS Foundational Security Best Practices v1.0.0** security standard.
- New controls from Security Hub CSPM are added as checks in Trusted Advisor within two to four weeks. See the note.

For more information, see the Prerequisites.

I configured Security Hub CSPM and AWS Config correctly, but my findings are still missing

It can take up to two hours for Security Hub CSPM to have findings for your resources. It can then take up to 24 hours for that data to appear in the Trusted Advisor console. Check the Trusted Advisor console again later.

Notes

- Only your findings for controls in the AWS Foundational Security Best Practices security standard will appear in Trusted Advisor *except* for controls that have the **Category: Recover > Resilience**.
- If there's a service issue with Security Hub CSPM or Security Hub CSPM isn't available, it can take up to 24 hours for your findings to appear in Trusted Advisor. Check the Trusted Advisor console again later.

I want to disable specific Security Hub CSPM controls

Security Hub CSPM sends your data to Trusted Advisor automatically. If you disable a Security Hub CSPM control or no longer have resources for that control, your findings won't appear in Trusted Advisor.

You can sign in to the Security Hub CSPM console and verify if your control is enabled or disabled.

If you disable a Security Hub CSPM control or disable all controls for the AWS Foundational Security Best Practices security standard, your findings are archived within the next five days. This five-day period to archive is approximate and best effort only, and isn't guaranteed. When your findings are archived, they are removed from Trusted Advisor.

For more information, see the following topics:

- Disabling and enabling individual controls
- Disabling or enabling a security standard

I want to find my excluded Security Hub CSPM resources

From the Trusted Advisor console, you can choose your Security Hub CSPM control name, and then choose the **Excluded items** option. This option displays all resources that are suppressed in Security Hub CSPM.

If the workflow status for a resource is set to SUPPRESSED, then that resource is an excluded item in Trusted Advisor. You can't suppress Security Hub CSPM resources from the Trusted Advisor console. To do so, use the Security Hub CSPM console. For more information, see Setting the workflow status for findings.

I want to enable or disable this feature for a member account that belongs to an AWS organization

By default, member accounts inherit the feature from the management account for AWS Organizations. If the management account has enabled the feature, then all accounts in the organization will also have the feature. If you have a member account and want to make specific changes for your account, you must contact AWS Support.

I see multiple AWS Regions for the same affected resource for a Security Hub CSPM check

Some AWS services are global and aren't specific to a Region, such as IAM and Amazon CloudFront. By default, global resources such as Amazon S3 buckets appear in the US East (N. Virginia) Region.

For Security Hub CSPM checks that evaluate resources for global services, you might see more than one item for affected resources. For example, if the `Hardware MFA should be enabled for the root user` check identifies that your account hasn't activated this feature, then you will see multiple Regions in the table for the same resource.

You can configure Security Hub CSPM and AWS Config so that multiple Regions won't appear for the same resource. For more information, see AWS Foundational Best Practices controls that you might want to disable.

I turned off Security Hub CSPM or AWS Config in a Region

If you stop resource recording with AWS Config or disable Security Hub CSPM in an AWS Region, Trusted Advisor no longer receives data for any controls in that Region. Trusted Advisor removes your Security Hub CSPM findings within 7-9 days. This time frame is best effort and isn't guaranteed. For more information, see Disabling Security Hub CSPM.

To disable this feature for your account, see Disable Security Hub CSPM from Trusted Advisor.

My control is archived in Security Hub CSPM, but I still see the findings in Trusted Advisor

When the `RecordState` status changes to `ARCHIVED` for a finding, Trusted Advisor deletes the finding for that Security Hub CSPM control from your account. You might still see the finding in Trusted Advisor for up to 7-9 days before it's deleted. This time frame is best effort and isn't guaranteed.

I still can't view my Security Hub CSPM findings

If you still have issues with this feature, you can create a technical support case in the AWS Support Center.

Opt in AWS Compute Optimizer for Trusted Advisor checks

Compute Optimizer is a service that analyzes the configuration and utilization metrics of your AWS resources. This service reports whether your resources are correctly configured for efficiency and reliability. It also suggests improvements you can implement to improve workload performance. With Compute Optimizer, you view the same recommendations in your Trusted Advisor checks.

You can opt in either your AWS account only, or all member accounts that are part of an organization in AWS Organizations. For more information, see Getting started in the *AWS Compute Optimizer User Guide*.

Once you opt in for Compute Optimizer, the following checks receive data from your Lambda functions and Amazon EBS volumes. It can take up to 12 hours to generate the findings and optimization recommendations. It can then take up to 48 hours to view your results in Trusted Advisor for the following checks:

Cost optimization

- Amazon EBS over-provisioned volumes
- AWS Lambda over-provisioned functions for memory size

Performance

- Amazon EBS under-provisioned volumes
- AWS Lambda under-provisioned functions for memory size

Notes

- Results for these check are automatically refreshed at least once daily, and refresh requests are not allowed. It might take a few hours for changes to appear. You can use the Trusted Advisor console to exclude resources from checks that automatically refresh. You can use the BatchUpdateRecommendationResourceExclusion API to exclude resources from any check except Trusted Advisor Priority recommendation resources.
- Trusted Advisor already has the Underutilized Amazon EBS Volumes and the Overutilized Amazon EBS Magnetic Volumes checks.

After you opt in with Compute Optimizer, we recommend that you use the new Amazon EBS over-provisioned volumes and Amazon EBS under-provisioned volumes checks instead.

Related information

For more information, see the following topics:

- Viewing Amazon EBS volume recommendations in the *AWS Compute Optimizer User Guide*
- Viewing Lambda function recommendations in the *AWS Compute Optimizer User Guide*
- Configuring Lambda function memory in the *AWS Lambda Developer Guide*
- Request modifications to your Amazon EBS volumes in the *Amazon EC2 User Guide*

Get started with AWS Trusted Advisor Priority

Trusted Advisor Priority helps you secure and optimize your AWS account to follow AWS best practices. With Trusted Advisor Priority, your AWS account team can proactively monitor your account and create prioritized recommendations when they identify opportunities for you.

For example, your account team can identify if your AWS account root user lacks multi-factor authentication (MFA). Your account team can create a recommendation so that you can take immediate action on a check, such as `MFA on Root Account`. The recommendation appears as an active **prioritized recommendation** on the Trusted Advisor Priority page of the Trusted Advisor console. You then follow the recommendations to resolve it.

Trusted Advisor Priority recommendations come from these two sources:

- AWS services – Services such as Trusted Advisor, AWS Security Hub CSPM, and AWS Well-Architected automatically create recommendations. Your account team shares these recommendations with you so that those recommendations appear in Trusted Advisor Priority.
- Your account team – Your account team can create manual recommendations.

Trusted Advisor Priority helps you focus on the most important recommendations. You and your account team can monitor the recommendation lifecycle, from the point when your account team shared the recommendation, up to the point when you acknowledge, resolve, or dismiss it.

You can use Trusted Advisor Priority to find recommendations for all member accounts in your organization.

Topics

Prerequisites

You must meet the following requirements to use Trusted Advisor Priority:

- You must have an AWS Enterprise Support or AWS Unified Operations plan.
- Your account must be part of an organization that has enabled all features in AWS Organizations. For more information, see Enabling all features in your organization in the *AWS Organizations User Guide*.
- Your organization must have enabled trusted access to Trusted Advisor. To enable trusted access, log in as the management account. Open the Your organization page in the Trusted Advisor console.
- You must be signed in to your AWS account to view Trusted Advisor Priority recommendations for your account.
- You must be signed in to the organization's management account or a delegated administrator account to view aggregated recommendations across your organization. For instructions on how to register delegated administrator accounts, see Register delegated administrators.

- You must have AWS Identity and Access Management (IAM) permissions to access Trusted Advisor Priority. For information on how to control access to Trusted Advisor Priority, see Manage access to AWS Trusted Advisor and AWS managed policies for AWS Trusted Advisor.

Enable Trusted Advisor Priority

Ask your account team to enable this feature for you. You must have an AWS Unified Operations plan and be the management account owner for your organization. If the Trusted Advisor Priority page in the console says that you need trusted access with AWS Organizations, then choose **Enable trusted access with AWS Organizations**. For more information, see the Prerequisites section.

View prioritized recommendations

After your account team enables Trusted Advisor Priority for you, you can view the latest recommendations for your AWS account.

To view your prioritized recommendations

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. On the **Trusted Advisor Priority** page, you can view the following items:

   If you're using an AWS Organizations Management or Delegated Administrator account, then switch to the **My Account** tab.

   - **Actions needed** – The number of recommendations that are pending a response or are in progress.
   - **Overview** – The following information:
     - Dismissed recommendations in the last 90 days
     - Resolved recommendations in the last 90 days
     - Recommendations without an update in over 30 days
     - Average time to resolve recommendations
3. On the **Active** tab, the **Active prioritized recommendations** show recommendations that your account team prioritized for you. The **Closed** tab shows resolved or dismissed recommendations.

   - To filter your results, use the following options:

- **Recommendation** – Enter keywords to search by name. This can be a check name, or a custom name that your account team created.
- **Status** – Whether the recommendation is pending a response, in progress, dismissed, or resolved.
- **Source** – The origin of a prioritized recommendation. The recommendation can come from AWS services, your AWS account team, or a planned service event.
- **Category** – The recommendation category, such as security or cost optimization.
- **Age** – When your account team shared the recommendation with you.

4. Choose a recommendation to learn more about its details, the affected resources, and the recommended actions. You can then acknowledge or dismiss the recommendation.

To view prioritized recommendations across all accounts in your AWS organization

Both the management account and the Trusted Advisor Priority delegated administrators can view recommendations aggregated across your organization.

> **Note**
>
> Member accounts don't have access to aggregated recommendations.

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. On the **Trusted Advisor Priority** page, make sure that you're on the **My Organization** tab.
3. To view recommendations for one account, select an account from the **Select an account from your organization** dropdown list. Or, you can view recommendations across all your accounts.

   On the **My Organization** tab, you can view the following items:

   - **Actions needed:** The number of recommendations across your organization that are pending a response or are in progress.
   - **Overview:** Shows the following items:

     - Dismissed recommendations in the last 90 days.

     - Resolved recommendations in the last 90 days.

- Recommendations without an update in over 30 days.

- The average time taken to resolve recommendations.

4. Under the **Active** tab, the **Active prioritized recommendations** section shows recommendations that your account team prioritized for you. The **Closed** tab shows resolved or dismissed recommendations.

   To filter your results, use the following options:

   - **Recommendation** – Enter keywords to search by name. This can be either a check name, or a custom name that your account team created.
   - **Status** – Whether the recommendation is pending a response, in progress, dismissed, or resolved.
   - **Source** – The origin of a prioritized recommendation. The recommendation can come from AWS services, your AWS account team, or a planned service event.
   - **Category** – The recommendation category, such as security or cost optimization.
   - **Age** – When your account team shared the recommendation with you.

5. Choose a recommendation to see additional details, affected accounts and resources, and the recommended actions. You can then acknowledge or dismiss the recommendation.

Acknowledge a recommendation

Under the **Active** tab, you can learn more about the recommendation and then decide if you want to acknowledge it.

To acknowledge a recommendation

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. If you're using an AWS Organizations Management or Delegated Administrator account, then switch to the **My Account** tab.
3. On the **Trusted Advisor Priority** page, under the **Active** tab, choose a recommendation name.
4. In the **Details** section, you can review the recommended actions to resolve the recommendation.
5. In the **Affected resources** section, you can review the affected resources and filter by *Status*.

6. Choose **Acknowledge**.
7. In the **Acknowledge recommendation** dialog box, choose **Acknowledge**.

   The recommendation status changes to **In progress**. Recommendations in progress or pending a response appear in the **Active** tab on the Trusted Advisor Priority page.
8. Follow the recommended actions to resolve the recommendation. For more information, see Resolve a recommendation.

To acknowledge a recommendation for all accounts in your AWS organization

The management account or the Trusted Advisor delegated administrators can acknowledge a recommendation for all of the affected accounts.

> **Note**
>
> Member accounts don't have access to aggregated recommendations.

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. On the **Trusted Advisor Priority** page, make sure that you're on the **My organization** tab.
3. In the **Active** tab, select a recommendation name.
4. Choose **Acknowledge**.
5. In the **Acknowledge recommendation** dialog box, choose **Acknowledge**.

   The recommendation status changes to **In progress**.
6. Follow the recommended actions to resolve the recommendation. For more information, see Resolve a recommendation.
7. To view the recommendation details, choose the recommendation name.

   In the **Details** section, you can review the following information about the recommendation:

   - An **Overview** of the recommendation and a **Details** section covering the recommendation actions to complete.

     A **Status summary** that shows recommendations across all affected accounts.

- In the **Affected accounts** section, you can review the affected resources across all your accounts. You can filter by **Account number** and **Status**.
- In the **Affected resources** section, you can review the affected resources across all your accounts. You can filter by **Account number** and **Status**.

Dismiss a recommendation

You can also dismiss a recommendation. This means that you acknowledge the recommendation, but you won't address it. You can dismiss a recommendation if it's not relevant to your account. For example, if you have a test AWS account that you plan to delete, you don't need to follow the recommended actions.

To dismiss a recommendation

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. If you're using an AWS Organizations Management or Delegated Administrator account, then switch to the **My Account** tab.
3. On the **Trusted Advisor Priority** page, under the **Active** tab, choose a recommendation name.
4. On the recommendation detail page, review the information about the affected resources.
5. If this recommendation doesn't apply for your account, choose **Dismiss**.
6. In the **Dismiss recommendation** dialog box, select a reason why you won't address the recommendation.
7. (Optional) Enter a note detailing why you're dismissing the recommendation. If you choose **Other**, you must enter a description in the **Note** section.
8. Choose **Dismiss**. The recommendation status changes to **Dismissed** and appears in the **Closed** tab on the Trusted Advisor Priority page.

To dismiss a recommendation for all the accounts in your AWS organization

The management account or the delgated administrator of Trusted Advisor Priority can dismiss a recommendation for all of their accounts.

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.

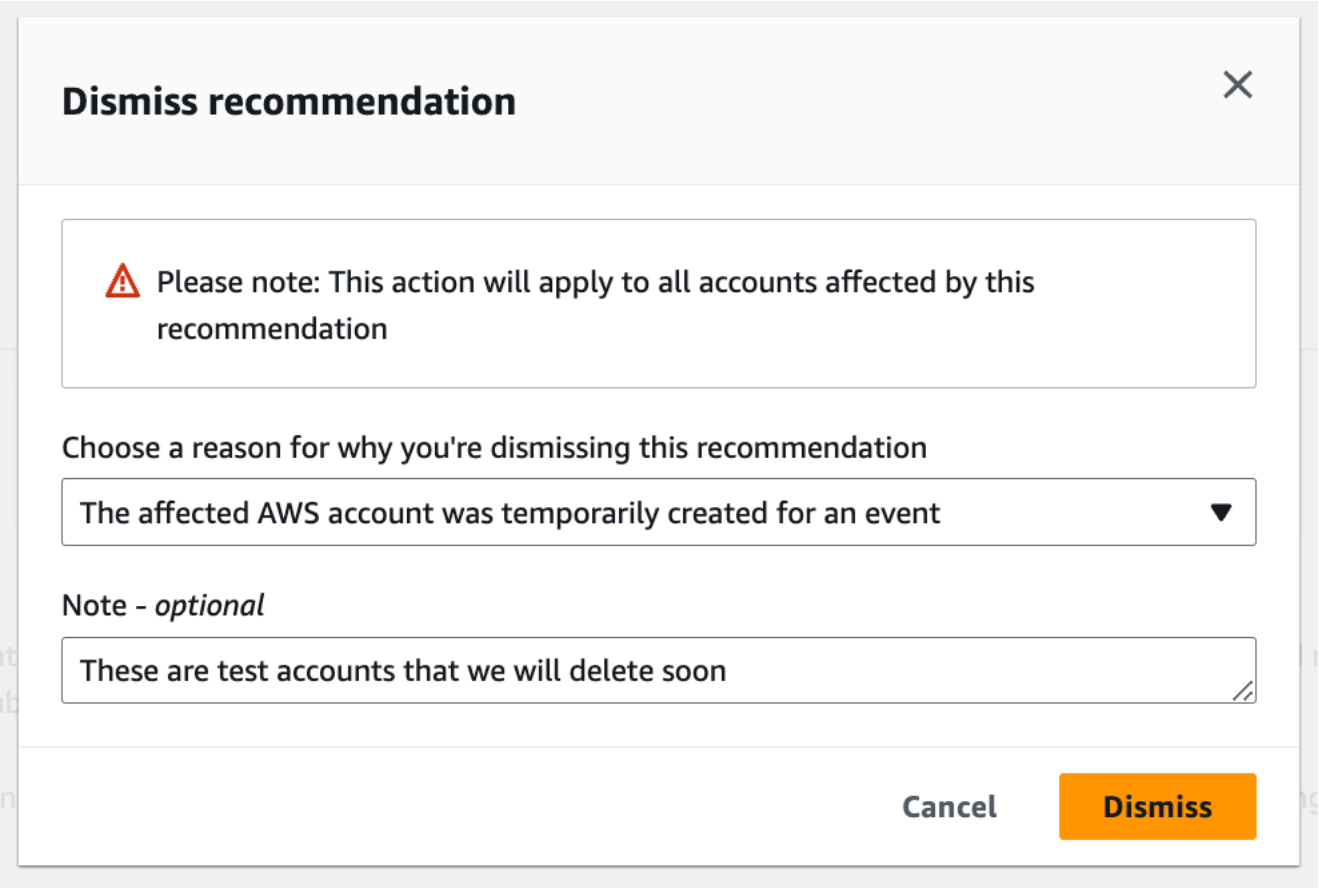
2. On the Trusted Advisor Priority page, make sure that you're on the **My Organization** tab.
3. In the **Active** tab, select a recommendation name.
4. If this recommendation doesn't apply for your account, then choose **Dismiss**.
5. In the **Dismiss recommendation** dialog box, select a reason why you won't address the recommendation.
6. (Optional) Enter a note detailing why you're dismissing the recommendation. If you choose **Other**, then you must enter a description in the **Note** section.
7. Choose **Dismiss**. The recommendation status changes to **Dismissed**. The recommendation appears in the **Closed** tab on the Trusted Advisor Priority page.

> **Note**
>
> You can choose the recommendation name and choose **View note** to find the reason for dismissal. If your account team dismissed the recommendation for you, their email address appears next to the note.

Trusted Advisor Priority also notifies your account team that you dismissed the recommendation.

Example : Dismiss a recommendation from Trusted Advisor Priority

The following example shows how you can dismiss a recommendation.



Resolve a recommendation

After you acknowledge the recommendation and complete the recommended actions, you can resolve the recommendation.

> **Tip**
>
> After you resolve a recommendation, you can't reopen it. If you want to revisit the recommendation again later, see Dismiss a recommendation.

To resolve a recommendation

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. On the Trusted Advisor Priority page, make sure that you're on the **My Organization** tab.
3. On the **Trusted Advisor Priority** page, select the recommendation, and then choose **Resolve**.

4. In the **Resolve recommendation** dialog box, choose **Resolve**. Resolved recommendations appear under the **Closed** tab on the Trusted Advisor Priority page. Trusted Advisor Priority notifies your account team that you resolved the recommendation.

To resolve a recommendation for all accounts in your AWS organization

The management account or the Trusted Advisor Priority delegated administrators can resolve a recommendation for all their accounts.

> **Note**
>
> Member accounts don't have access to aggregated recommendations.

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. If you're using an AWS Organizations Management or Delegated Administrator account, switch to the **My Account** tab.
3. In the **Active** tab, select a recommendation name.
4. If the recommendation doesn't apply for your account, choose **Resolve**.
5. In the **Resolve recommendation** dialog box, choose **Resolve**. Resolved recommendations appear under the **Closed** tab on the Trusted Advisor Priority page. Trusted Advisor Priority notifies your account team that you resolved the recommendation.

Reopen a recommendation

After you dismiss a recommendation, you or your account team can reopen the recommendation.

To reopen a recommendation

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. If you're using an AWS Organizations Management or Delegated Administrator account, then switch to the **My Account** tab.
3. On the **Trusted Advisor Priority** page, choose the **Closed** tab.

4. Under **Closed recommendations**, select a recommendation that was **Dismissed**, and then choose **Reopen**.
5. In the **Reopen recommendation** dialog box, describe why you're reopening the recommendation.
6. Choose **Reopen**. The recommendation status changes to **In progress** and appears under the **Active** tab.

> **Tip**
>
> You can choose the recommendation name and then choose **View note** to find the reason for reopening. If your account team reopened the recommendation for you, their name appears next to the note.

7. Follow the steps in the recommendation details.

To reopen a recommendation for all accounts in your AWS organization

The management account or the Trusted Advisor Priority delegated administrators can reopen a recommendation for all of their accounts.

> **Note**
>
> Member accounts don't have access to aggregated recommendations.

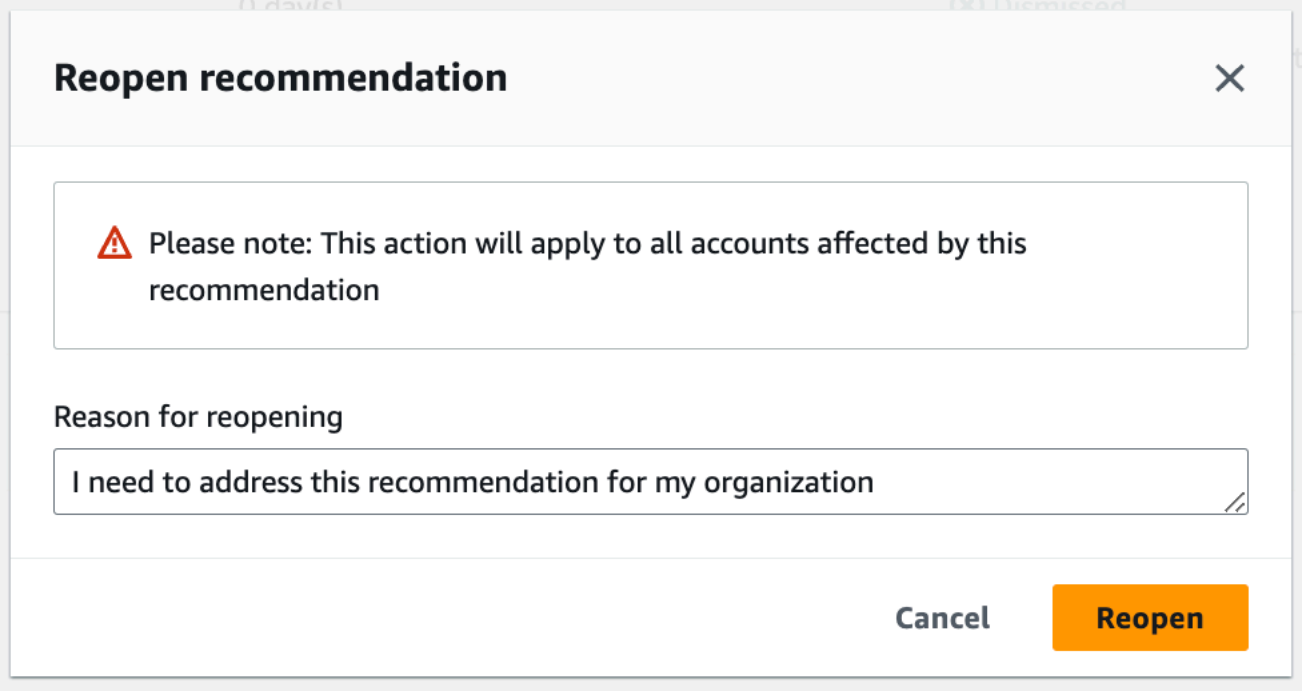
1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. On the Trusted Advisor Priority page, make sure that you're on the **My Organization** tab.
3. Under **Closed** recommendations, select a recommendation that was **Dismissed**, and then choose **Reopen**.
4. In the **Reopen recommendation** dialog box, describe why you're reopening the recommendation.
5. Choose **Reopen**. The recommendation status changes to **In progress** and appears under the **Active** tab.

> **Tip**
>
> You can choose the recommendation name and choose **View note** to find the reason for reopening. If your account team reopened the recommendation for you, their name appears next to the note.

6. Follow the steps in the recommendation details.

Example : Reopen a recommendation from Trusted Advisor Priority

The following example shows a recommendation that you want to reopen.



Download recommendation details

You can also download the results of a prioritized recommendation from Trusted Advisor Priority.

> **Note**
>
> Currently, you can download only one recommendation at a time.

To download a recommendation

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home.
2. On the **Trusted Advisor Priority** page, select the recommendation, and then choose **Download**.
3. Open the file to view the recommendation details.

Register delegated administrators

You can add member accounts that are part of your organization as delegated administrators. Delegated administrator accounts can review, acknowledge, resolve, dismiss, and reopen recommendations in Trusted Advisor Priority.

After you register an account, you must grant the delegated administrator the required AWS Identity and Access Management permissions to access Trusted Advisor Priority. For more information, see Manage access to AWS Trusted Advisor and AWS managed policies for AWS Trusted Advisor.

You can register up to five member accounts. Only the management account can add delegated administrators for the organization. You must be signed in to the organization's management account to register or deregister a delegated administrator.

To register a delegated administrator

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home as the management account.
2. In the navigation pane, under **Preferences**, choose **Your organization**.
3. Under **Delegated administrator**, choose **Register new account**.
4. In the dialog box, enter the member account ID, and then choose **Register**.
5. (Optional) To deregister an account, select an account and choose **Deregister**. In the dialog box, choose **Deregister** again.

Deregister delegated administrators

When you deregister a member account, that account no longer has the same access to Trusted Advisor Priority as the management account. Accounts that are no longer delegated administrators won't receive email notifications from Trusted Advisor Priority.

To deregister a delegated administrator

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home as the management account.
2. In the navigation pane, under **Preferences**, choose **Your organization**.
3. Under **Delegated administrator**, select an account and then choose **Deregister**.
4. In the dialog box, choose **Deregister**.

Manage Trusted Advisor Priority notifications

Trusted Advisor Priority delivers notifications through email. This email notification includes a summary of the recommendations that your account team prioritized for you. You can specify the frequency that you receive updates from Trusted Advisor Priority.

If you registered member accounts as delegated administrators, they can also set up their accounts to receive Trusted Advisor Priority email notifications.

Trusted Advisor Priority email notifications don't include check results for individual accounts and are separate from the weekly notification for Trusted Advisor Recommendations. For more information, see Set up notification preferences.

> **Note**
>
> Only the management account or delegated administrator can set up Trusted Advisor Priority email notifications.

To manage your Trusted Advisor Priority notifications

1. Sign in to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor/home as a management or delegated administrator account.

2. In the navigation pane, under **Preferences**, choose **Notifications**.
3. Under **Priority**, you can select the following options.

    a. **Daily** – Receive an email notification daily.

    b. **Weekly** – Receive an email notification once a week.

    c. Choose the notifications to receive:

    - Summary of prioritized recommendations
    - Resolution dates

4. For **Recipients**, select other contacts that you want to receive the email notifications. You can add and remove contacts from the Account Settings page in the AWS Billing and Cost Management console.
5. For **Language**, choose the language for the email notification.
6. Choose **Save your preferences**.

> **Note**
>
> Trusted Advisor Priority sends email notifications from the noreply@notifications.trustedadvisor.us-west-2.amazonaws.com address. You might need to verify that your email client doesn't identify these emails as spam.

Disable Trusted Advisor Priority

Contact your account team and ask that they disable this feature for you. After this feature is disabled, prioritized recommendations no longer appear in your Trusted Advisor console.

If you disable Trusted Advisor Priority and then enable it again later, you can still view the recommendations that your account team sent before you disabled Trusted Advisor Priority.

AWS Trusted Advisor check reference

You can view all Trusted Advisor check names, descriptions, and IDs in the following reference. You can also sign in to the Trusted Advisor console to view more information about the checks, recommended actions, and their statuses.

If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, you can also use the AWS Trusted Advisor API and the AWS Command Line Interface (AWS CLI) to access your checks. For more information, see the following topics:

- Get started with the Trusted Advisor API
- AWS Trusted Advisor API Reference

Cost optimization

You can use the following checks for the cost optimization category.

Check names

- Unassociated Elastic IP Addresses

Unassociated Elastic IP Addresses

Description

Checks for Elastic IP addresses (EIPs) that are not associated with a running Amazon Elastic Compute Cloud (Amazon EC2) instance.

EIPs are static IP addresses designed for dynamic cloud computing. Unlike traditional static IP addresses, EIPs mask the failure of an instance or Availability Zone by remapping a public IP address to another instance in your account. A nominal charge is imposed for an EIP that is not associated with a running instance.

> **Note**
>
> This check reports the resources that are flagged by the criteria and the total number of resources evaluated, including 0K resources. The resources table lists only the flagged resources.

Check ID

`Z4AUBRNSmz`

Alert Criteria

Yellow: An allocated Elastic IP address (EIP) is not associated with a running Amazon EC2 instance.

Recommended Action

Associate the EIP with a running active instance, or release the unassociated EIP. For more information, see Associating an Elastic IP Address with a Different Running Instance and Releasing an Elastic IP Address.

Additional Resources

Elastic IP Addresses

Report columns

- Region
- IP Address

Performance

Improve the performance of your service by checking your service quotas (formerly referred to as limits), so that you can take advantage of provisioned throughput, monitor for overutilized instances, and detect any unused resources.

You can use the following checks for the performance category.

Check names

- Amazon EBS Provisioned IOPS (SSD) Volume Attachment Configuration
- High CPU Utilization Amazon EC2 Instances
- Large Number of EC2 Security Group Rules Applied to an Instance
- Large Number of Rules in an EC2 Security Group
- Over-utilized Amazon EBS Magnetic Volumes

Amazon EBS Provisioned IOPS (SSD) Volume Attachment Configuration

Description

Checks for Provisioned IOPS (SSD) volumes that are attached to an Amazon EBS optimizable Amazon Elastic Compute Cloud (Amazon EC2) instance that is not EBS-optimized.

Provisioned IOPS (SSD) volumes in the Amazon Elastic Block Store (Amazon EBS) are designed to deliver the expected performance only when they are attached to an EBS-optimized instance.

Check ID

`PPkZrjsH2q`

Alert Criteria

Yellow: An Amazon EC2 instance that can be EBS-optimized has an attached Provisioned IOPS (SSD) volume but the instance is not EBS-optimized.

Recommended Action

Create a new instance that is EBS-optimized, detach the volume, and reattach the volume to your new instance. For more information, see Amazon EBS-Optimized Instances and Attaching an Amazon EBS Volume to an Instance.

Additional Resources

- Amazon EBS Volume Types
- Amazon EBS Volume Performance

Report columns

- Status
- Region/AZ
- Volume ID
- Volume Name
- Volume Attachment
- Instance ID
- Instance Type
- EBS Optimized

High CPU Utilization Amazon EC2 Instances

Description

Checks the Amazon Elastic Compute Cloud (Amazon EC2) instances that were running at any time during the last 14 days. An alert is sent if daily CPU utilization was greater than 90% on four or more days.

Consistent high utilization can indicate optimized, steady performance. However, it can also indicate that an application does not have enough resources. To get daily CPU utilization data, download the report for this check.

> **Note**
>
> This check reports the resources that are flagged by the criteria and the total number of resources evaluated, including 0K resources. The resources table lists only the flagged resources.

Check ID

`ZRxQlPsb6c`

Alert Criteria

Yellow: An instance had more than 90% daily average CPU utilization on at least 4 of the previous 14 days.

Recommended Action

Consider adding more instances. For information about scaling the number of instances based on demand, see What is Auto Scaling?

Additional Resources

- Monitoring Amazon EC2
- Instance Metadata and User Data
- Amazon CloudWatch User Guide
- Amazon EC2 Auto Scaling User Guide

Report columns

- Region/AZ
- Instance ID
- Instance Type
- Instance Name
- 14-Day Average CPU Utilization
- Number of Days over 90% CPU Utilization

Large Number of EC2 Security Group Rules Applied to an Instance

Description

Checks for EC2 instances that have a large number of security group rules. Performance can be degraded if an instance has a large number of rules.

> **Note**
>
> This check reports the resources that are flagged by the criteria and the total number of resources evaluated, including 0K resources. The resources table lists only the flagged resources.

Check ID

`j3DFqYTe29`

Alert Criteria

Yellow: An EC2-VPC instance has more than 50 security group rules.

Yellow: An EC2-Classic instance has more than 100 security group rules.

Recommended Action

Reduce the number of rules associated with an instance by deleting unnecessary or overlapping rules. For more information, see Deleting Rules from a Security Group.

Additional Resources

EC2 Security Groups

Report columns

- Region
- Instance ID
- Instance Name
- VPC ID
- Total Inbound Rules
- Total Outbound Rules

Large Number of Rules in an EC2 Security Group

Description

Checks each Amazon Elastic Compute Cloud (Amazon EC2) security group for an excessive number of rules.

If a security group has a large number of rules, performance can be degraded.

> **Note**
>
> This check reports the resources that are flagged by the criteria and the total number of resources evaluated, including 0K resources. The resources table lists only the flagged resources.

Check ID

`tfg86AVHAZ`

Alert Criteria

Yellow: An Amazon EC2-VPC security group has more than 50 rules.

Yellow: An Amazon EC2-Classic security group has more than 100 rules.

Recommended Action

Reduce the number of rules in a security group by deleting unnecessary or overlapping rules. For more information, see Amazon EC2 security groups for your EC2 instances.

Additional Resources

- Amazon EC2 Security Groups

Report columns

- Region
- Security Group Name
- Group ID
- Description
- Instance Count

- VPC ID
- Total Outbound Rules

Over-utilized Amazon EBS Magnetic Volumes

Description

Checks for Amazon Elastic Block Store (Amazon EBS) magnetic volumes that are potentially over-utilized and might benefit from a more efficient configuration.

A magnetic volume is designed for applications with moderate or bursty input/output (I/O) requirements, and the IOPS rate is not guaranteed. It delivers approximately 100 IOPS on average, with a best-effort ability to burst to hundreds of IOPS. For consistently higher IOPS, you can use a Provisioned IOPS (SSD) volume. For bursty IOPS, you can use a General Purpose (SSD) volume. For more information, see Amazon EBS Volume Types.

For a list of instance types that support EBS-optimized behavior, see Amazon EBS-Optimized Instances.

To get daily utilization metrics, download the report for this check. The detailed report shows a column for each of the last 14 days. If there is no active EBS volume, the cell is empty. If there is insufficient data to make a reliable measurement, the cell contains `N/A`. If there is sufficient data, the cell contains the daily median and the percentage of the variance in relation to the median (for example, `256 / 20%`)

Check ID

`k3J2hns32g`

Alert Criteria

Yellow: An Amazon EBS magnetic volume is attached to an instance that can be EBS-optimized or is part of a cluster compute network with a daily median of more than 95 IOPS, and varies by less than 10% of the median value for at least 7 of the past 14 days.

Recommended Action

For consistently higher IOPS, you can use a Provisioned IOPS (SSD) volume. For bursty IOPS, you can use a General Purpose (SSD) volume. For more information, see Amazon EBS Volume Types.

Additional Resources

- Amazon Elastic Block Store User Guide

Report columns

- Status
- Region
- Volume ID
- Volume Name
- Number of Days Over
- Max Daily Median
- Total Outbound Rules

Security

You can use the following checks for the security category.

> **Note**
>
> If you enabled Security Hub CSPM for your AWS account, you can view your findings in the Trusted Advisor console. For information, see Viewing AWS Security Hub CSPM controls in AWS Trusted Advisor.
> You can view all controls in the AWS Foundational Security Best Practices security standard *except* for controls that have the **Category: Recover > Resilience**. For a list of supported controls, see AWS Foundational Security Best Practices controls in the *AWS Security Hub CSPM User Guide*.

Check names

- Amazon RDS Security Group Access Risk
- AWS CloudTrail Logging
- IAM Access Key Rotation
- IAM Password Policy
- IAM Use
- Security Groups – Specific Ports Unrestricted
- Security Groups – Unrestricted Access

Amazon RDS Security Group Access Risk

Description

Checks security group configurations for Amazon Relational Database Service (Amazon RDS) and warns when a security group rule grants overly permissive access to your database. The recommended configuration for a security group rule is to allow access only from specific Amazon Elastic Compute Cloud (Amazon EC2) security groups or from a specific IP address.

> **Note**
>
> This check evaluates only security groups that are attached toAmazon RDS instances running outside on an Amazon VPC.

Check ID

`nNauJisYIT`

Alert Criteria

- Yellow: A DB security group rule references an Amazon EC2 security group that grants global access on one of these ports: 20, 21, 22, 1433, 1434, 3306, 3389, 4333, 5432, 5500.
- Red: A DB security group rule grants global access (the CIDR rule suffix is /0).
- Green: A DB security group doesn't include permissive rules.

Recommended Action

EC2-Classic was retired on August 15, 2022. It's recommend to move your Amazon RDS instances to a VPC and use Amazon EC2 security groups. For more information of moving your DB instance to a VPC see Moving a DB instance not in a VPC into a VPC.

If you are unable to migrate your Amazon RDS instances to a VPC, then review your security group rules and restrict access to authorized IP addresses or IP ranges. To edit a security group, use the AuthorizeDBSecurityGroupIngress API or the AWS Management Console. For more information, see Working with DB Security Groups.

Additional Resources

- Amazon RDS Security Groups
- Classless Inter-Domain Routing
- List of TCP and UDP port numbers

Report columns

- Status
- Region
- RDS Security Group Name
- Ingress Rule
- Reason

AWS CloudTrail Logging

Description

Checks your use of AWS CloudTrail. CloudTrail provides increased visibility into activity in your AWS account by recording information about AWS API calls made on the account. You can use these logs to determine, for example, what actions a particular user has taken during a specified time period, or which users have taken actions on a particular resource during a specified time period.

Because CloudTrail delivers log files to an Amazon Simple Storage Service (Amazon S3) bucket, CloudTrail must have write permissions for the bucket. If a trail applies to all Regions (the default when creating a new trail),the trail appears multiple times in the Trusted Advisor report.

Check ID

`vjafUGJ9H0`

Alert Criteria

- Yellow: CloudTrail reports log delivery errors for a trail.
- Red: A trail has not been created for a Region, or logging is turned off for a trail.

Recommended Action

To create a trail and start logging from the console, go to the AWS CloudTrail console.

To start logging, see Stopping and Starting Logging for a Trail.

If you receive log delivery errors, check to make sure that the bucket exists and that the necessary policy is attached to the bucket. See Amazon S3 Bucket Policy.

Additional Resources

- AWS CloudTrail User Guide

- Supported Regions
- Supported Services

Report columns

- Status
- Region
- Trail Name
- Logging Status
- Bucket Name
- Last Delivery Date

IAM Access Key Rotation

Description

Checks for active IAM access keys that have not been rotated in the last 90 days.

When you rotate your access keys regularly, you reduce the chance that a compromised key could be used without your knowledge to access resources. For the purposes of this check, the last rotation date and time is when the access key was created or most recently activated. The access key number and date come from the `access_key_1_last_rotated` and `access_key_2_last_rotated` information in the most recent IAM credential report.

Because the regeneration frequency of a credential report is restricted, refreshing this check might not reflect recent changes. For more information, see Getting Credential Reports for Your AWS account.

In order to create and rotate access keys, a user must have the appropriate permissions. For more information, see Allow Users to Manage Their Own Passwords, Access Keys, and SSH Keys.

Check ID

`DqdJqYeRm5`

Alert Criteria

- Green: The access key is active and has been rotated in the last 90 days.
- Yellow: The access key is active and has been rotated in the last 2 years, but more than 90 days ago.
- Red: The access key is active and has not been rotated in the last 2 years.

Recommended Action

Rotate access keys on a regular basis. See Rotating Access Keys and Managing Access Keys for IAM Users.

Additional Resources

- IAM Best Practices
- How to rotate access keys for IAM users

Report columns

- Status
- IAM user
- Access Key
- Key Last Rotated
- Reason

IAM Password Policy

Description

Checks the password policy for your account and warns when a password policy is not enabled, or if password content requirements have not been enabled.

Password content requirements increase the overall security of your AWS environment by enforcing the creation of strong user passwords. When you create or change a password policy, the change is enforced immediately for new users but does not require existing users to change their passwords.

Check ID

`Yw2K9puPzl`

Alert Criteria

- Green: A password policy is enabled with recommended content requirement enabled.
- Yellow: A password policy is enabled, but at least one content requirement is not enabled.

Recommended Action

If some content requirements are not enabled, consider enabling them. If no password policy is enabled, create and configure one. See Setting an Account Password Policy for IAM Users.

To access the AWS Management Console, IAM users need passwords. As a best practice, AWS highly recommends that instead of creating IAM users, you use federation. Federation allows users to use their existing corporate credentials to log into the AWS Management Console. Use IAM Identity Center to create or federate the user, and then assume an IAM role into an account.

To learn more about identity providers and federation, see Identity providers and federation in the IAM User Guide. To learn more about IAM Identity Center, see the IAM Identity Center User Guide.

Additional Resources

Managing Passwords

Report columns

- Password Policy
- Uppercase
- Lowercase
- Number
- Non-alphanumeric

IAM Use

Description

This check is intended to discourage the use of root access by checking for existence of at least one IAM user. You may ignore the alert if you are following the best practice of centralizing identities and configuring users in an Identity providers and federation or IAM Identity Center.

Check ID

`zXCkfM1nI3`

Alert Criteria

- Yellow: No IAM users have been created for this account.

Recommended Action

Create an IAM user or use IAM Identity Center to create additional users whose permissions are limited to perform specific tasks in your AWS environment.

Additional Resources

- What is IAM Identity Center

What Is IAM?

Report columns

- Password Policy
- Uppercase
- Lowercase
- Number
- Non-alphanumeric

Security Groups – Specific Ports Unrestricted

Description

Checks security groups for rules that allow unrestricted access (0.0.0.0/0) to specific ports.

Unrestricted access increases opportunities for malicious activity (hacking, denial-of-service attacks, loss of data). The ports with highest risk are flagged red, and those with less risk are flagged yellow. Ports flagged green are typically used by applications that require unrestricted access, such as HTTP and SMTP.

If you have intentionally configured your security groups in this manner, we recommend using additional security measures to secure your infrastructure (such as IP tables).

Note

This check only evaluates security groups that you create and their inbound rules for IPv4 addresses. Security groups created by AWS Directory Service are flagged as red or yellow, but they don't pose a security risk and can be excluded. For more information, see the Trusted Advisor FAQ.

Note

This check reports the resources that are flagged by the criteria and the total number of resources evaluated, including 0K resources. The resources table lists only the flagged resources.

Check ID

HCP4007jGY

Alert Criteria

- Green: Security Group provides unrestricted access on ports 80, 25, 443, or 465.
- Red: Security Group is attached to a resource and provides unrestricted access to port 20, 21, 22 , 1433, 1434, 3306, 3389, 4333, 5432, or 5500.
- Yellow: Security Group provides unrestricted access to any other port.
- Yellow: Security Group is not attached to any resource and provides unrestricted access.

Recommended Action

Restrict access to only those IP addresses that require it. To restrict access to a specific IP address, set the suffix to /32 (for example, 192.0.2.10/32). Be sure to delete overly permissive rules after creating rules that are more restrictive.

Review and delete unused security groups. You can use AWS Firewall Manager to centrally configure and manage security groups at scale across AWS accounts, For more information, see the AWS Firewall Manager documentation.

Consider using Systems Manager Sessions Manager for SSH (Port 22) and RDP (Port 3389) access to EC2 instances. With sessions manager, you can access your EC2 instances without enabling port 22 and 3389 in the security group.

Additional Resources

- Amazon EC2 Security Groups

  List of TCP and UDP port numbers
- Classless Inter-Domain Routing
- Working with Session Manager
- AWS Firewall Manager

Report columns

- Status
- Region
- Security Group Name
- Security Group ID
- Protocol

- From Port
- To Port
- Association

Security Groups – Unrestricted Access

Description

Checks security groups for rules that allow unrestricted access to a resource.

Unrestricted access increases opportunities for malicious activity (hacking, denial-of-service attacks, loss of data).

> **Note**
>
> This check evaluates only security groups that you create and their inbound rules for IPv4 addresses. Security groups created by AWS Directory Service are flagged as red or yellow, but they don't pose a security risk and can be excluded. For more information, see the Trusted Advisor FAQ.
>
> > **Note**
> >
> > This check reports the resources that are flagged by the criteria and the total number of resources evaluated, including 0K resources. The resources table lists only the flagged resources.

Check ID

`1iG5NDGVre`

Alert Criteria

- Green: A security group rule has a source IP address with a /0 suffix for ports 25, 80, or 443.
- Yellow: A security group rule has a source IP address with a /0 suffix for ports other than 25, 80, or 443 and security group is attached to a resource.
- Red: A security group rule has a source IP address with a /0 suffix for ports other than 25, 80, or 443 and security group is not attached to a resource.

Recommended Action

Restrict access to only those IP addresses that require it. To restrict access to a specific IP address, set the suffix to /32 (for example, 192.0.2.10/32). Be sure to delete overly permissive rules after creating rules that are more restrictive.

Review and delete unused security groups. You can use AWS Firewall Manager to centrally configure and manage security groups at scale across AWS accounts, For more information, see the AWS Firewall Manager documentation.

Consider using Systems Manager Sessions Manager for SSH (Port 22) and RDP (Port 3389) access to EC2 instances. With sessions manager, you can access your EC2 instances without enabling port 22 and 3389 in the security group.

Additional Resources

- Amazon EC2 Security Groups
- Classless Inter-Domain Routing
- Working with Session Manager
- AWS Firewall Manager

Report columns

- Status
- Region
- Security Group Name
- Security Group ID
- Protocol
- From Port
- To Port
- IP Range
- Association

Fault tolerance

You can use the following checks for the fault tolerance category.

Check names

- Amazon Aurora DB Instance Accessibility
- Amazon EBS Snapshots
- Amazon EC2 Availability Zone Balance
- Amazon RDS Backups
- Amazon RDS Multi-AZ
- VPN Tunnel Redundancy

Amazon Aurora DB Instance Accessibility

Description

Checks for cases where an Amazon Aurora DB cluster has both private and public instances.

When your primary instance fails, a replica can be promoted to a primary instance. If that replica is private, users who have only public access would no longer be able to connect to the database after failover. We recommend that all the DB instances in a cluster have the same accessibility.

Check ID

`xuy7H1avtl`

Alert Criteria

Yellow: The instances in an Aurora DB cluster have different accessibility (a mix of public and private).

Recommended Action

Modify the `Publicly Accessible` setting of the instances in the DB cluster so that they are all either public or private. For details, see the instructions for MySQL instances at Modifying a DB Instance Running the MySQL Database Engine.

Additional Resources

Fault Tolerance for an Aurora DB Cluster

Report columns

- Status
- Region
- Cluster

- Public DB Instances
- Private DB Instances
- Reason

Amazon EBS Snapshots

Description

Checks the age of the snapshots for your Amazon EBS volumes (either available or in-use). Failures can occur even if Amazon EBS volumes are replicated. Snapshots are persisted to Amazon S3 for durable storage and point-in-time recovery.

Check ID

`H7IgTzjTYb`

Alert Criteria

- Yellow: The most recent volume snapshot is between 7 and 30 days old.
- Red: The most recent volume snapshot is more than 30 days old.
- Red: The volume does not have a snapshot.

Recommended Action

Create weekly or monthly snapshots of your volumes. For more information, see Creating an Amazon EBS Snapshot.

To automate the creation of EBS snapshots, you can consider using AWS Backup or Amazon Data Lifecycle Manager.

Additional Resources

Amazon Elastic Block Store (Amazon EBS)

Amazon EBS Snapshots

AWS Backup

Amazon Data Lifecycle Manager

Report columns

- Status
- Region

- Volume ID
- Volume Name
- Snapshot ID
- Snapshot Name
- Snapshot Age
- Volume Attachment
- Reason

Amazon EC2 Availability Zone Balance

Description

Checks the distribution of Amazon Elastic Compute Cloud (Amazon EC2) instances across Availability Zones in a Region.

Availability Zones are distinct locations that are insulated from failures in other Availability Zones. This allows inexpensive, low-latency network connectivity between Availability Zones in the same Region. By launching instances in multiple Availability Zones in the same Region, you can help protect your applications from a single point of failure.

Check ID

`wuy7G1zxql`

Alert Criteria

- Yellow: The Region has instances in multiple zones, but the distribution is uneven (the difference between the highest and lowest instance counts in utilized Availability Zones is greater than 20%).
- Red: The Region has instances only in a single Availability Zone.

Recommended Action

Balance your Amazon EC2 instances evenly across multiple Availability Zones. You can do this by launching instances manually or by using Auto Scaling to do it automatically. For more information, see Launch Your Instance and Load Balance Your Auto Scaling Group.

Additional Resources

- Amazon EC2 Auto Scaling User Guide
- Placement groups for your Amazon EC2 instances

- Amazon EC2 instance types

Report columns

- Status
- Region
- Zone a Instances
- Zone b Instances
- Zone c Instances
- Zone e Instances
- Zone f Instances
- Reason

Amazon RDS Backups

Description

Checks for automated backups of Amazon RDS DB instances.

By default, backups are enabled with a retention period of one day. Backups reduce the risk of unexpected data loss and allow for point-in-time recovery.

> **Note**
>
> This check reports the resources that are flagged by the criteria and the total number of resources evaluated, including 0K resources. The resources table lists only the flagged resources.

Check ID

opQPADkZvH

Alert Criteria

Red: A DB instance has the backup retention period set to 0 days.

Recommended Action

Set the retention period for the automated DB instance backup to 1 to 35 days as appropriate to the requirements of your application. See Working With Automated Backups.

Additional Resources

Getting Started with Amazon RDS

Report columns

- Status
- Region/AZ
- DB Instance
- VPC ID
- Backup Retention Period

Amazon RDS Multi-AZ

Description

Checks for DB instances that are deployed in a single Availability Zone (AZ).

Multi-AZ deployments enhance database availability by synchronously replicating to a standby instance in a different Availability Zone. During planned database maintenance, or the failure of a DB instance or Availability Zone, Amazon RDS automatically fails over to the standby. This failover allows database operations to resume quickly without administrative intervention. Because Amazon RDS does not support Multi-AZ deployment for Microsoft SQL Server, this check does not examine SQL Server instances.

> **Note**
>
> This check reports the resources that are flagged by the criteria and the total number of resources evaluated, including 0K resources. The resources table lists only the flagged resources.

Check ID

`f2iK5R6Dep`

Alert Criteria

Yellow: A DB instance is deployed in a single Availability Zone.

Recommended Action

If your application requires high availability, modify your DB instance to enable Multi-AZ deployment. See High Availability (Multi-AZ).

Additional Resources

Regions and Availability Zones

Report columns

- Status
- Region/AZ
- DB Instance
- VPC ID
- Multi-AZ

VPN Tunnel Redundancy

Description

Checks the number of tunnels that are active for each of your Site-to-Site VPNs.

A VPN should have two tunnels configured at all times. This provides redundancy in case of outage or planned maintenance of the devices at the AWS endpoint. For some hardware, only one tunnel is active at a time. If a VPN has no active tunnels, charges for the VPN might still apply. For more information, see AWS Site-to-Site VPN User Guide.

Check ID

`S45wrEXrLz`

Alert Criteria

- Yellow: A VPN has one active tunnel (this is normal for some hardware).
- Yellow: A VPN has no active tunnels.

Recommended Action

Be sure that two tunnels are configured for your VPN connection, and that both are active if your hardware supports it. If you no longer need a VPN connection, you can delete it to avoid charges. For more information, see Your customer gateway device or Delete a Site-to-Site VPN connection.

Additional Resources

- AWS Site-to-Site VPN User Guide
- Create a target gateway

Report columns

- Status
- Region
- VPN ID
- VPC
- Virtual Private Gateway
- Customer Gateway
- Active Tunnels
- Reason

Service limits

See the following checks for the service limits (also known as quotas) category.

All checks in this category have the following descriptions:

Alert Criteria

- Yellow: 80% of limit reached.
- Red: 100% of limit reached.
- Blue: Trusted Advisor was unable to retrieve utilization or limits in one or more AWS Regions.

Recommended Action

If you expect to exceed a service limit, request an increase directly from the Service Quotas console. If Service Quotas doesn't support your service yet, you can open a support case in Support Center.

Report columns

- Status
- Service
- Region
- Limit Amount

- Current Usage

> **Note**
>
> - Values are based on a snapshot, so your current usage might differ. Quota and usage data can take up to 24 hours to reflect any changes. In cases where quotas have been recently increased, you might temporarily see utilization that exceeds the quota.

Check names

- CloudFormation Stacks
- DynamoDB Read Capacity
- DynamoDB Write Capacity
- IAM Group
- IAM Instance Profiles
- IAM Policies
- IAM Roles
- IAM Server Certificates
- IAM Users
- Kinesis Shards per Region
- RDS Cluster Parameter Groups
- RDS Cluster Roles
- RDS Clusters
- RDS DB Instances
- RDS DB Manual Snapshots
- RDS DB Parameter Groups
- RDS Event Subscriptions
- RDS Option Groups
- RDS Read Replicas per Master
- RDS Reserved Instances
- RDS Subnet Groups

- RDS Subnets per Subnet Group
- RDS Total Storage Quota

CloudFormation Stacks

Description

Checks for usage that is more than 80% of the CloudFormation stacks quota.

Check ID

`gW7HH017J9`

Additional Resources

CloudFormation quotas

DynamoDB Read Capacity

Description

Checks for usage that is more than 80% of the DynamoDB provisioned throughput limit for reads per AWS account.

Check ID

`6gtQddfEw6`

Additional Resources

DynamoDB quotas

DynamoDB Write Capacity

Description

Checks for usage that is more than 80% of the DynamoDB provisioned throughput limit for writes per AWS account.

Check ID

`c5ftjdfkMr`

Additional Resources

DynamoDB quotas

IAM Group

Description

Checks for usage that is more than 80% of the IAM group quota.

Check ID

sU7XX0l7J9

Additional Resources

IAM quotas

IAM Instance Profiles

Description

Checks for usage that is more than 80% of the IAM instance profiles quota.

Check ID

n07SS0l7J9

Additional Resources

IAM quotas

IAM Policies

Description

Checks for usage that is more than 80% of the IAM policies quota.

Check ID

pR7UU0l7J9

Additional Resources

IAM quotas

IAM Roles

Description

Checks for usage that is more than 80% of the IAM roles quota.

Check ID

oQ7TT0l7J9

Additional Resources

IAM quotas

IAM Server Certificates

Description

Checks for usage that is more than 80% of the IAM server certificates quota.

Check ID

rT7WW0l7J9

Additional Resources

IAM quotas

IAM Users

Description

Checks for usage that is more than 80% of the IAM users quota.

Check ID

qS7VV0l7J9

Additional Resources

IAM quotas

Kinesis Shards per Region

Description

Checks for usage that is more than 80% of the Kinesis shards per Region quota.

Check ID

`bW7HH0l7J9`

Additional Resources

Kinesis quotas

RDS Cluster Parameter Groups

Description

Checks for usage that is more than 80% of the RDS cluster parameter groups quota.

Check ID

`jtlIMO3qZM`

Additional Resources

Amazon RDS quotas

RDS Cluster Roles

Description

Checks for usage that is more than 80% of the RDS cluster roles quota.

Check ID

`7fuccf1Mx7`

Additional Resources

Amazon RDS quotas

RDS Clusters

Description

Checks for usage that is more than 80% of the RDS clusters quota.

Check ID

`gjqMBn6pjz`

Additional Resources

Amazon RDS quotas

RDS DB Instances

Description

Checks for usage that is more than 80% of the RDS DB instances quota.

Check ID

`XG0aXHpIEt`

Additional Resources

Amazon RDS quotas

RDS DB Manual Snapshots

Description

Checks for usage that is more than 80% of the RDS DB manual snapshots quota.

Check ID

`dV84wpqRUs`

Additional Resources

Amazon RDS quotas

RDS DB Parameter Groups

Description

Checks for usage that is more than 80% of the RDS DB parameter groups quota.

Check ID

`jEECYg2YVU`

Additional Resources

Amazon RDS quotas

RDS Event Subscriptions

Description

Checks for usage that is more than 80% of the RDS event subscriptions quota.

Check ID

`keAhfbH5yb`

Additional Resources

Amazon RDS quotas

RDS Option Groups

Description

Checks for usage that is more than 80% of the RDS option groups quota.

Check ID

`3Njm0DJQO9`

Additional Resources

Amazon RDS quotas

RDS Read Replicas per Master

Description

Checks for usage that is more than 80% of the RDS read replicas per master quota.

Check ID

pYW8UkYz2w

Additional Resources

Amazon RDS quotas

RDS Reserved Instances

Description

Checks for usage that is more than 80% of the RDS Reserved Instances quota.

Check ID

UUDv0a5r34

Additional Resources

Amazon RDS quotas

RDS Subnet Groups

Description

Checks for usage that is more than 80% of the RDS subnet groups quota.

Check ID

dYWBaXaaMM

Additional Resources

Amazon RDS quotas

RDS Subnets per Subnet Group

Description

Checks for usage that is more than 80% of the RDS subnets per subnet group quota.

Check ID

`jEhCtdJKOY`

Additional Resources

Amazon RDS quotas

RDS Total Storage Quota

Description

Checks for usage that is more than 80% of the RDS total storage quota.

Check ID

`P1jhKWEmLa`

Additional Resources

Amazon RDS quotas

Change log for AWS Trusted Advisor

See the following topic for recent changes to Trusted Advisor checks.

> **Note**
>
> If you use the Trusted Advisor console or the AWS Support API, deprecated checks won't appear in check results. If you use a deprecated check, such as specifying the check ID in an AWS Support API operation or your code, then you receive API call errors. Remove these checks to avoid errors.

For more information about the available checks, see the AWS Trusted Advisor check reference.

Change date	Check name	Change description
December 18, 2025	Updated Amazon S3 Bucket Versioning	Added a new **Alert criteria**: • Yellow: Trusted Advisor doesn't have access to validate versioning
December 17, 2025	Updated Amazon S3 Bucket Permissions	Updated the **Alert criteria** section.
November 21, 2025	Updated Application Load Balancer security group	Updated the Application Load Balancer security group alerts and recommendations.
November 17, 2025	Updated AWS STS global endpoint usage across AWS Regions check description	Updated the AWS STS global endpoint usage across AWS Regions check description to clarify when check results are refreshed.
October 15, 2025	Updated multiple check descriptions	A note was added to multiple check descriptions to indicate that the check reports all resources that are flagged by the criteria and the total number of resources evaluated, including 0K resources. The resources table lists only the flagged resources.
September 11, 2025	L4dfs2Q4C5: AWS Lambda functions using deprecated runtimes	Updated **Yellow** alert criterion to indicate that runtimes deprecating within at least 180 are included.

Change date	Check name	Change description
August 19, 2025	Pfx0RwqBli: Amazon S3 Bucket Permissions	Alert criteria updated: *Trusted Advisor does not have permission to check the policy or ACL, or the policy or ACL could not be evaluated for other reasons* changed from **Yellow** to **Red**.
July 03, 2025	c1dfprch15: Amazon EC2 instances with Ubuntu LTS end of standard support	Updated the note to indicate that this check refreshes at least once daily.
July 02, 2025	c1dvkm4z6b: Amazon ECS AWSLogs driver in blocking mode	Amazon ECS changed the default setting for awslogs driver logging configura tion parameter mode from `blocking` to `non-block ing` . The Yellow status description has been updated to reflect this change.
July 02, 2025	7DAFEmoDos: MFA on root account	Added information indicatin g that member account root user credentials can be deleted centrally, removing the need to manage MFA on root user credentials.
June 9, 2025	c1z7kmr17n: Amazon Aurora cost optimization recommend ations for DB cluster storage	New check
June 09, 2025	c15m0mgld3: AWS STS global endpoint usage across AWS Regions	Updated check: This check is now available for all AWS Support plans.

Change date	Check name	Change description
April 30, 2025	• N420c450f2: CloudFront Alternate Domain Names • N425c450f2: CloudFront Custom SSL Certificates in the IAM Certificate Store	Added a note indicating that this check applies to classic Amazon CloudFront distribut ions.
April 30, 2025	N415c450f2: CloudFront Header Forwarding and Cache Hit Ratio	Added a note indicating that this check applies to classic Amazon CloudFront distribut ions.
April 02, 2025	c1dfprch02: Amazon EFS Throughput Mode Optimizat ion	The description of this check has changed. For more information, see Amazon EC2 instances with Microsoft Windows Server end of support.
April 02, 2025	Qsdfp3A4L4: Amazon EC2 instances with Microsoft Windows Server end of support	The description of this check has changed. For more information, see Amazon EFS Throughput Mode Optimizat ion.

Older updates

The following AWS Security Hub CSPM checks are deprecated:

Check name	Check ID
S3.10 - S3 general purpose buckets with versioning enabled should have lifecycle configurations	`Hs4Ma3G211`

Check name	Check ID
S3.11 - S3 general purpose buckets should have event notifications enabled	Hs4Ma3G212
CodeBuild.5 - CodeBuild project environments should not have privileged mode enabled	Hs4Ma3G218
CloudFormation.1 - CloudFormation stacks should be integrated with Amazon Simple Notification Service (SNS)	Hs4Ma3G245
SNS.2 - Logging of delivery status should be enabled for notification messages sent to a topic	Hs4Ma3G263
Athena.1 - Athena workgroups should be encrypted at rest	Hs4Ma3G294

Added 1 new check

Trusted Advisor added 1 new check on November 22, 2024:

- 8604e947f2 - Application Load Balancer Security Groups

Updated 3 checks

Trusted Advisor updated 3 checks on November 7, 2024:

- b92b83d667 - ELB Target Imbalance
- 8CNsSllI5v - Auto Scaling Group Resources
- wuy7G1zxql - Amazon EC2 Availability Zone Balance

Added 4 checks

Trusted Advisor added 4 new checks on October 11, 2024:

- 07602fcad6 - IAM Access Analyzer - external access
- 528d6f5ee7 - GWLB - Endpoint AZ
- c2vlfg0jp6 - Inactive VPC interface endpoints
- c2vlfg0k35 - Inactive Gateway Load Balancer endpoints

Updated 3 checks

Trusted Advisor updated 3 checks on October 2, 2024:

- Check ID 7040ea389a moved from Cost Optimization pillar to the Fault Tolerance pillar
- Updated Check ID 7DAFEmoDos
- Updated Check ID Cmsvnj8db2

Added 9 new checks

Trusted Advisor added 9 new checks on August 23, 2024:

- c2vlfg0p86 - [IAM] - SAML 2.0 Identity Provider
- 7040ea389a - Network Firewall endpoint Cross-AZ Data Transfer
- c2vlfg0bfw - Low utilization Network Firewall
- c2vlfg0gqd - Network Firewall Multi-AZ
- c2vlfg0p1w - Application Load Balancer Target Groups encrypted protocol
- c2vlfg022t - [NAT Gateway] - Underutilized Resource
- c243hjzrhn - AWS Outposts Single Rack deployment
- b92b83d667 - ELB Target Imbalance
- 90046ff5b5 - MSK availability is limited to two zones

For more information, see the AWS Trusted Advisor check reference.

Updated 1 Security check and added 1 Security check

Trusted Advisor updated 1 Operational Excellence checks on August 22, 2024:

- c1fd6b96l4

Trusted Advisor added 1 Security checks on August 22, 2024:

- c2vlfg0f4h

For more information, see the AWS Trusted Advisor check reference.

Updated 6 Security checks

Trusted Advisor updated 6 Security checks on August 20, 2024:

- nNauJisYIT
- c9D319e7sG
- a2sEc6ILx
- HCP4007jGY
- 1iG5NDGVre
- Yw2K9puPzl

For more information, see the AWS Trusted Advisor check reference.

Updated 1 fault tolerance checks

Trusted Advisor updated the 1 fault tolerance check and 1 security on August 12, 2024:

- VPN Tunnel Redundancy
- Amazon RDS engine minor version upgrade is required

For more information, see the AWS Trusted Advisor check reference.

Updated 9 checks

Trusted Advisor updated the 9 checks on July 21, 2024:

- 7qGXsKIUw
- ZRxQlPsb6c
- N425c450f2
- 7DAFEmoDos

- Pfx0RwqBli
- H7IgTzjTYb
- C056F80cR3
- Yw2K9puPzl
- xSqX82fQu

For more information, see the AWS Trusted Advisor check reference.

Removed 5 checks and added 1 check

Trusted Advisor deprecated 3 Fault Tolerance checks, 1 Perfomance check, and 1 Security check on May 15, 2024:

- IAM Use
- ELB Cross-Zone Load Balancing
- Overutilized Amazon EBS Magnetic Volumes
- Large Number of EC2 Security Group Rules Applied to an Instance
- Large Number of Rules in an EC2 Security Group

Trusted Advisor added 1 new security check on May 15, 2024:

- Amazon S3 Server Access Logs Enabled

For more information, see the AWS Trusted Advisor check reference.

Removed fault tolerance checks

Trusted Advisor deprecated 3 Fault Tolerance check on April 25, 2024:

- Direct Connect Connection Redundancy
- Direct Connect Location Redundancy
- Direct Connect Virtual Interface Redundancy

For more information, see the AWS Trusted Advisor check reference.

New fault tolerance check

Trusted Advisor added 1 Fault Tolerance check on February 29, 2024:

- NLB - Internet-facing resource in private subnet

For more information, see the AWS Trusted Advisor check reference.

Updated fault tolerance and security checks

Trusted Advisor added 1 new Fault Tolerance check and amended 1 existing Fault tolerance and 1 Security check on March 28 2024:

- Added AWS Resilience Hub Application Component check
- Updated AWS Lambda VPC-enabled Functions without Multi-AZ Redundancy
- Updated AWS Lambda Functions Using Deprecated Runtimes

For more information, see the AWS Trusted Advisor check reference.

New fault tolerance check

Trusted Advisor added 1 Fault Tolerance check on January 31, 2024:

- Direct Connect Location Resiliency

For more information, see the AWS Trusted Advisor check reference.

Updated fault tolerance check

Trusted Advisor amended 1 Fault Tolerance check on January 08, 2024:

- Amazon RDS innodb_flush_log_at_trx_commit parameter is not 1

For more information, see the AWS Trusted Advisor check reference.

Updated security check

Trusted Advisor amended 1 Security check on December 21, 2023:

- AWS Lambda Functions Using Deprecated Runtimes

For more information, see the AWS Trusted Advisor check reference.

New security and performance checks

Trusted Advisor added 2 new Security checks and 2 new Performance checks on December 20, 2023:

- Amazon EFS clients not using data-in-transit encryption
- Amazon Aurora DB cluster under-provisioned for read workload
- Amazon RDS instance under-provisioned for system capacity
- Amazon EC2 instances with Ubuntu LTS end of standard support

For more information, see the AWS Trusted Advisor check reference.

New security check

Trusted Advisor added 1 new Security check on December 15, 2023:

- Amazon Route 53 mismatching CNAME records pointing directly to S3 buckets

For more information, see the AWS Trusted Advisor check reference.

New fault tolerance and cost optimization checks

Trusted Advisor added 2 new Fault Tolerance checks and 1 new Cost Optimization check on December 07, 2023:

- Amazon DocumentDB Single-AZ clusters
- Amazon S3 Incomplete Multipart Upload Abort Configuration
- Amazon ECS AWSLogs driver in blocking mode

For more information, see the AWS Trusted Advisor check reference.

Trusted Advisor check removal

Check name	Check category	Check ID
EBS volumes should be attached to EC2 instances	Security	`Hs4Ma3G119`
S3 buckets should have server-side encryption enabled	Security	`Hs4Ma3G167`
CloudFront distributions should have origin access identity enabled	Security	`Hs4Ma3G195`

Updates to the Trusted Advisor integration with AWS Security Hub CSPM

Trusted Advisor made the following update on November 17, 2022.

If you disable Security Hub CSPM or AWS Config for an AWS Region, Trusted Advisor now removes your control findings for that AWS Region within 7-9 days. Previously, the time frame to remove your Security Hub CSPM data from Trusted Advisor was 90 days.

For more information, see the following sections in the Troubleshooting topic:

- I turned off Security Hub CSPM or AWS Config in a Region
- My control is archived in Security Hub CSPM, but I still see the findings in Trusted Advisor

Update to the Trusted Advisor console

Trusted Advisor added the following change on November 16, 2022.

The Trusted Advisor Dashboard in the console is now Trusted Advisor Recommendations. The Trusted Advisor Recommendations page still shows the check results and the available checks for each category for your AWS account.

This name change only updates the Trusted Advisor console. You can continue to use the Trusted Advisor console and the Trusted Advisor operations in the Support API as usual.

For more information, see Get started with Trusted Advisor Recommendations.

Added Security Hub CSPM checks to Trusted Advisor

As of June 23, 2022, Trusted Advisor only supports Security Hub CSPM controls available through April 7, 2022. This release supports all controls in the AWS Foundational Security Best Practices security standard except for controls in the Category: Recover > Resilience. For more information, see Viewing AWS Security Hub CSPM controls in AWS Trusted Advisor.

For a list of supported controls, see AWS Foundational Security Best Practices controls in the *AWS Security Hub CSPM User Guide*.

Added checks from AWS Compute Optimizer

Trusted Advisor added the following checks on May 4, 2022.

Check name	Check category	Check ID
Amazon EBS over-provisioned volumes	Cost optimization	`COr6dfpM03`
Amazon EBS under-pro visioned volumes	Performance	`COr6dfpM04`
AWS Lambda over-prov isioned functions for memory size	Cost optimization	`COr6dfpM05`
AWS Lambda under-pro visioned functions for memory size	Performance	`COr6dfpM06`

You must opt in your AWS account for Compute Optimizer so that these checks can receive data from your Lambda and Amazon EBS resources. For more information, see Opt in AWS Compute Optimizer for Trusted Advisor checks.

Updated checks for AWS Direct Connect

Trusted Advisor updated the following checks on March 29, 2022.

Check name	Check category	Check ID
AWS Direct Connect Connection Redundancy	Fault tolerance	`0t121N1Ty3`
AWS Direct Connect Location Redundancy	Fault tolerance	`8M012Ph3U5`
AWS Direct Connect Virtual Interface Redundancy	Fault tolerance	`4g3Nt5M1Th`

- The value for the **Region** column now shows the AWS Region code instead of the full name. For example, resources in US East (N. Virginia) will now have the `us-east-1` value.
- The value for the **Time Stamp** column now appears in the RFC 3339 format, such as `2022-03-30T01:02:27.000Z`.
- Resources that don't have any detected problems will now appear in the check table. These resources will have a check mark icon () next to them.

  Previously, only resources that Trusted Advisor recommended that you investigate appeared in the table. These resources have a warning icon () next to them.

Updated check name for Amazon OpenSearch Service

Trusted Advisor updated the name for the Amazon OpenSearch Service Reserved Instance Optimization check on September 8, 2021.

The check recommendations, category, and ID are the same.

> **Note**
>
> If you use Trusted Advisor for Amazon CloudWatch metrics, the metric name for this check is also updated. For more information, see Creating Amazon CloudWatch alarms to monitor AWS Trusted Advisor metrics.

Added checks for AWS Lambda

Trusted Advisor added the following checks on March 8, 2021.

Check name	Check category	Check ID
AWS Lambda Functions with Excessive Timeouts	Cost optimization	`L4dfs2Q3C3`
AWS Lambda Functions with High Error Rates	Cost optimization	`L4dfs2Q3C2`
AWS Lambda Functions Using Deprecated Runtimes	Security	`L4dfs2Q4C5`
AWS Lambda VPC-enabled Functions without Multi-AZ Redundancy	Fault tolerance	`L4dfs2Q4C6`

For more information about how to use these checks with Lambda, see Example AWS Trusted Advisor workflow to view recommendations in the *AWS Lambda Developer Guide*.

Trusted Advisor check removal

Check name	Check category	Check ID
EC2 Elastic IP Addresses	Service limits	`aW9HH018J6`

Updated checks for Amazon Elastic Block Store

Trusted Advisor updated the unit of Amazon EBS volume from gibibyte (GiB) to tebibyte (TiB) for the following checks on March 5, 2021.

> **Note**
>
> If you use Trusted Advisor for Amazon CloudWatch metrics, the metric names for these five checks are also updated. For more information, see Creating Amazon CloudWatch alarms to monitor AWS Trusted Advisor metrics.

Check name	Check category	Check ID	Updated CloudWatch metric for ServiceLimit
EBS Cold HDD (sc1) Volume Storage	Service limits	gH5CC0e3J9	Cold HDD (sc1) volume storage (TiB)
EBS General Purpose SSD (gp2) Volume Storage	Service limits	dH7RR0l6J9	General Purpose SSD (gp2) volume storage (TiB)
EBS Magnetic (standard) Volume Storage	Service limits	cG7HH0l7J9	Magnetic (standard) volume storage (TiB)
EBS Provisioned IOPS SSD (io1) Volume Storage	Service limits	gI7MM0l7J9	Provisioned IOPS (SSD) storage (TiB)
EBS Throughput Optimized HDD (st1) Volume Storage	Service limits	wH7DD0l3J9	Throughput Optimized HDD (st1) volume storage (TiB)

Trusted Advisor check removal

> **Note**
>
> Trusted Advisor removed the following checks on November 18, 2020.

Checks removed on November 18, 2020	Check category	Check ID
EC2Config Service for EC2 Windows Instances	Fault tolerance	V77i0L1Bqz
ENA Driver Version for EC2 Windows Instances	Fault tolerance	TyfdMXG69d
NVMe Driver Version for EC2 Windows Instances	Fault tolerance	yHAGQJV9K5
PV Driver Version for EC2 Windows Instances	Fault tolerance	Wnwm9Il5bG

You can monitor your Amazon EC2 instances and verify they are up to date by using AWS Systems Manager Distributor, other third-party tools, or write your own scripts to return driver information for Windows Management Instrumentation (WMI).

Trusted Advisor check removal

Trusted Advisor removed the following check on February 18, 2020.

Check name	Check category	Check ID
Service Limits	Performance	eW7HH0l7J9

AWS Support App in Slack

You can use the AWS Support App to manage your AWS support cases in Slack. Invite your team members to chat channels, respond to case updates, and chat directly with support agents. Use the AWS Support App to manage support cases quickly in Slack.

Use the AWS Support App to do the following:

- Create, update, search for, and resolve support cases in Slack channels
- Attach files to support cases
- Share support case details with your team without leaving the Slack channel
- Start a live chat session with support agents

When you create, update, or resolve a support case in the AWS Support App, the case is also updated in the AWS Support Center Console. You don't need to sign in to the Support Center Console to manage your support cases separately.

Notes

- The response times for support cases are the same, whether you created the case from Slack or from the Support Center Console.
- You can create a support case for account and billing support, and technical support.

Topics

- Deleting a Slack channel configuration from the AWS Support App
- Deleting a Slack workspace configuration from the AWS Support App
- AWS Support App in Slack commands
- View AWS Support App correspondences in the AWS Support Center Console
- Creating AWS Support App in Slack resources with AWS CloudFormation

Prerequisites

You must meet the following requirements to use the AWS Support App in Slack:

- You have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan. You can find your support plan from the AWS Support Center Console or from the Support plans page. For more information, see Compare AWS Support plans.
- You have a Slack workspace and channel for your organization. You must be a Slack workspace administrator, or have permission to add apps to that Slack workspace. For more information, see the Slack Help Center.
- You sign in to the AWS account as an AWS Identity and Access Management (IAM) user or role with the required permissions. For more information, see Managing access to the AWS Support App widget.
- You will need to create an IAM role that has the required permissions to perform actions for you. The AWS Support App uses this role to make API calls to different services. For more information, see Managing access to the AWS Support App.

Topics

- Managing access to the AWS Support App widget
- Managing access to the AWS Support App

Managing access to the AWS Support App widget

You can attach an AWS Identity and Access Management (IAM) policy to grant an IAM user permission to configure the AWS Support App widget in the AWS Support Center Console.

For more information about how to add a policy to an IAM entity, see Adding IAM identity permissions (console) in the IAM User Guide.

> **Note**
>
> You can also sign in as the root user in your AWS account, but we don't recommend that you do this. For more information about root user access, see Safeguard your root user credentials and don't use them for everyday tasks in the *IAM User Guide*.

Example IAM policy

You can attach the following policy to an entity, such as an IAM user or group. This policy allows a user to authorize a Slack workspace and configure Slack channels in the Support Center Console.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "supportapp:GetSlackOauthParameters",
                "supportapp:RedeemSlackOauthCode",
                "supportapp:DescribeSlackChannels",
                "supportapp:ListSlackWorkspaceConfigurations",
                "supportapp:ListSlackChannelConfigurations",
                "supportapp:CreateSlackChannelConfiguration",
                "supportapp:DeleteSlackChannelConfiguration",
                "supportapp:DeleteSlackWorkspaceConfiguration",
                "supportapp:GetAccountAlias",
                "supportapp:PutAccountAlias",
                "supportapp:DeleteAccountAlias",
                "supportapp:UpdateSlackChannelConfiguration",
                "iam:ListRoles"
            ],
            "Resource": "*"
        }
    ]
}
```

Permissions required to connect the AWS Support App to Slack

The AWS Support App includes permission-only actions that don't directly correspond to an API operation. These actions are indicated in the Service Authorization Reference with [permission only].

The AWS Support App uses the following API actions to connect to Slack and then lists your *public* Slack channels in the AWS Support Center Console:

- `supportapp:GetSlackOauthParameters`
- `supportapp:RedeemSlackOauthCode`
- `supportapp:DescribeSlackChannels`

These API actions are not intended to be called by your code. Therefore, these API actions are not included in the AWS CLI and AWS SDKs.

Managing access to the AWS Support App

After you have permissions to the AWS Support App widget, you must also create an AWS Identity and Access Management (IAM) role. This role performs actions from other AWS services for you, such as the AWS Support API and Service Quotas.

You then attach an IAM policy to this role so that the role has the required permissions to complete these actions. You choose this role when you create your Slack channel configuration in the Support Center Console.

Users in your Slack channel have the same permissions that you grant to the IAM role. For example, if you specify read-only access to your support cases, then users in your Slack channel can view your support cases, but can't update them.

> **Important**
>
> When you request a live chat with a support agent and choose new private channel as your live chat channel preference, the AWS Support App creates a separate Slack channel. This Slack channel has the same permissions as the channel where you created the case or initiated the chat.

If you change the IAM role or the IAM policy, your changes apply to the Slack channel that you configured and to any new live chat Slack channels that the AWS Support App creates for you.

Follow these procedures to create your IAM role and policy.

Topics

Use an AWS managed policy or create a customer managed policy

To grant your role permissions, you can use either an AWS managed policy or a customer managed policy.

Tip

If you don't want to create a policy manually, we recommend that you use an AWS managed policy instead and skip this procedure. Managed policies automatically have the required permissions for the AWS Support App. You don't need to update the policies manually. For more information, see AWS managed policies for AWS Support App in Slack.

Follow this procedure to create a customer managed policy for your role. This procedure uses the JSON policy editor in the IAM console.

To create a customer managed policy for the AWS Support App

1. Sign in to the AWS Management Console and open the IAM console at https://eusc-de-east-1.console.amazonaws-eusc.eu/iam/.
2. In the navigation pane, choose **Policies**.
3. Choose **Create policy**.
4. Choose the **JSON** tab.
5. Enter your JSON, and then replace the default JSON in the editor. You can use the example policy.

6. Choose **Next: Tags**.
7. (Optional) You can use tags as key–value pairs to add metadata to the policy.
8. Choose **Next: Review**.
9. On the **Review policy** page, enter a **Name**, such as *AWSSupportAppRolePolicy*, and a **Description** (optional).
10. Review the **Summary** page to see the permissions that the policy allows and then choose **Create policy**.

This policy defines the actions that the role can take. For more information, see Creating IAM policies (console) in the *IAM User Guide*.

Example IAM policy

You can attach the following example policy to your IAM role. This policy allows the role to have full permissions to all required actions for the AWS Support App. After you configure a Slack channel with the role, any user in your channel has the same permissions.

> **Note**
>
> For a list of AWS managed policies, see AWS managed policies for AWS Support App in Slack.

You can update the policy to remove a permission from the AWS Support App.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "supportapp:GetSlackOauthParameters",
                "supportapp:RedeemSlackOauthCode",
                "supportapp:DescribeSlackChannels",
                "supportapp:ListSlackWorkspaceConfigurations",
                "supportapp:ListSlackChannelConfigurations",
                "supportapp:CreateSlackChannelConfiguration",
```

```
                "supportapp:DeleteSlackChannelConfiguration",
                "supportapp:DeleteSlackWorkspaceConfiguration",
                "supportapp:GetAccountAlias",
                "supportapp:PutAccountAlias",
                "supportapp:DeleteAccountAlias",
                "supportapp:UpdateSlackChannelConfiguration",
                "iam:ListRoles"
            ],
            "Resource": "*"
        }
    ]
}
```

For descriptions for each action, see the following topics in the *Service Authorization Reference*:

- Actions, resources, and condition keys for AWS Support
- Actions, resources, and condition keys for Service Quotas
- Actions, resources, and condition keys for AWS Identity and Access Management

Create an IAM role

After you have your policy, you must create an IAM role, and then attach the policy to that role. You choose this role when you create a Slack channel configuration in the Support Center Console.

To create a role for the AWS Support App

1. Sign in to the AWS Management Console and open the IAM console at https://eusc-de-east-1.console.amazonaws-eusc.eu/iam/.
2. In the navigation pane, choose **Roles**, and then choose **Create role**.
3. For **Select trusted entity**, choose **AWS service**.
4. Choose **AWS Support App**.
5. Choose **Next: Permissions**.
6. Enter the policy name. You can choose the AWS managed policy or choose a customer managed policy that you created, such as *AWSSupportAppRolePolicy*. Then select the check box next to the policy.
7. Choose **Next: Tags**.
8. (Optional) You can use tags as key–value pairs to add metadata to the role.

9. Choose **Next: Review**.
10. For **Role name**, enter a name, such as *AWSSupportAppRole*.
11. (Optional) For **Role description**, enter a description for the role.
12. Review the role and then choose **Create role**. You can now choose this role when you configure a Slack channel in the Support Center Console. See Configuring a Slack channel.

For more information, see Creating a role for an AWS service in the *IAM User Guide*.

Troubleshooting

See the following topics to manage access to the AWS Support App.

Contents

I want to restrict specific users in my Slack channel from specific actions

By default, users in your Slack channel have the same permissions specified in the IAM policy that you attach to the IAM role that you create. This means anyone in the channel has read or write access to your support cases, whether or not they have an AWS account or an IAM user.

We recommend the following best practices:

- Configure private Slack channels with the AWS Support App
- Only invite users to your channel who need access to your support cases
- Use an IAM policy that has the minimum required permissions to the AWS Support App. See AWS managed policies for AWS Support App in Slack.

When I configure a Slack channel, I don't see the IAM role that I created

If your IAM role doesn't appear in the **IAM role for the AWS Support App** list, this means that the role doesn't have the AWS Support App as a trusted entity, or that the role was deleted. You can update the existing role, or create another one. See Create an IAM role.

My IAM role is missing a permission

The IAM role that you create for your Slack channel needs permissions to perform the actions that you want. For example, if you want your users in Slack to create support cases, the role must have the `support:CreateCase` permission. The AWS Support App assumes this role to perform these actions for you.

If you receive an error about a missing permission from the AWS Support App, verify that the policy attached to your role has the required permission.

See the previous Example IAM policy.

A Slack error says that my IAM role isn't valid

Verify that you chose the correct role for your channel configuration.

To verify your role

1. Sign in to the AWS Support Center Console at https://eusc-de-east-1.console.amazonaws-eusc.eu/support/app#/config page.
2. Choose the channel that you configured with the AWS Support App.
3. From the **Permissions** section, find the IAM role name that you chose.
   - To change the role, choose **Edit**, choose another role, and then choose **Save**.
   - To update the role or the policy attached to the role, sign in to the IAM console.

The AWS Support App says that I'm missing an IAM role for Service Quotas

You must have the `AWSServiceRoleForServiceQuotas` role in your account to request quota increases from Service Quotas. If you receive an error about a missing resource, complete one of the following steps:

- Use the Service Quotas console to request a quota increase. After you make a successful request, Service Quotas creates this role for you automatically. Then, you can use the AWS Support App to request quota increases in Slack. For more information, see Requesting a quota increase.

- Update the IAM policy attached to your role. This grants the role permission to Service Quotas. The following section in the Example IAM policy allows the AWS Support App to create the Service Quotas role for you.

```
{
    "Effect": "Allow",
    "Action": "iam:CreateServiceLinkedRole",
    "Resource": "*",
    "Condition": {
        "StringEquals": {"iam:AWSServiceName": "servicequotas.amazonaws.com"}
     }
}
```

If you delete the IAM role that you configure for your channel, you must manually create the role or update the IAM policy to allow the AWS Support App to create one for you.

Authorize a Slack workspace

After you authorize your workspace and give the AWS Support App permission to access it, you then need an AWS Identity and Access Management (IAM) role for your AWS account. The AWS Support App uses this role to call API operations from AWS Support and Service Quotas for you. For example, the AWS Support App uses the role to call the `CreateCase` operation to create a support case for you in Slack.

Notes

- Your Slack channel inherits permissions from the IAM role. All users in the Slack channel have the same permissions that are specified in the IAM policy attached to the role. For example, if your IAM policy grants full read and write permissions to your support cases, anyone in your Slack channel can create, update, and resolve your support cases. If your IAM policy allows the role read-only permissions, then users in your Slack channel can only view your support cases.
- We recommend that you add the Slack workspaces and channels that you need to manage your support operations. We recommend that you configure private channels and only invite required users.

You must authorize each Slack workspace that you want to use for your AWS account. If you have multiple AWS accounts, you must sign in to each account and repeat the following procedure to authorize the workspace. If your account belongs to an organization in AWS Organizations and you want to authorize multiple accounts, skip to Authorize multiple accounts.

To authorize the Slack workspace for your AWS account

1. Sign in to the **AWS Support Center Console** and choose **Slack configuration**.
2. On the **Getting started** page, choose **Authorize workspace**.
3. If you're not already signed in to Slack, on the **Sign in to your workspace** page, enter your workspace name, and then choose **Continue**.
4. On the **AWS Support is requesting permission to access the your-workspace-name Slack** page, choose **Allow**.

   > **Note**
   >
   > If you can't allow Slack to access your workspace, make sure that you have permissions from your Slack administrator to add the AWS Support App to the workspace. See Prerequisites.

   On the **Slack configuration** page, your workspace name appears under **Workspaces**.
5. (Optional) To add more workspaces, choose **Authorize workspace** and repeat steps 3-4. You can add up to five workspaces to your account.
6. (Optional) By default, your AWS account ID number appears as the account name in your Slack channel. To change this value, under **Account name**, choose **Edit**, enter your account name, and then choose **Save**.

   > **Tip**
   >
   > Use a name that you and your team can easily recognize. The AWS Support App uses this name to identify your account in the Slack channel. You can update this name at any time.

   Your workspace and account name appear on the **Slack configuration** page.

Authorize multiple accounts

To authorize multiple AWS accounts to use Slack workspaces, you can use AWS CloudFormation or Terraform to create your AWS Support App resources.

Configuring a Slack channel

After you authorize your Slack workspace, you can configure your Slack channels to use the AWS Support App.

In the channel where you invite and add the AWS Support App, you can create and search for cases, and receive case notifications. This channel displays case updates, including newly created or resolved cases, added correspondence, and shared case details.

Your Slack channel inherits permissions from the IAM role. All users in the Slack channel have the same permissions that are specified in the IAM policy attached to the role. For example, if your IAM policy grants full read and write permissions to your support cases, anyone in your Slack channel can create, update, and resolve your support cases. If your IAM policy allows the role read-only permissions, then users in your Slack channel can only view your support cases.

You can add up to 20 channels per AWS account. Each Slack channel supports up to 100 AWS accounts. This means that only 100 accounts can add the same Slack channel to the AWS Support App. To minimize distractions, we recommend adding only the accounts necessary for managing your organization's support cases.

Each AWS account must configure a Slack channel separately in the AWS Support App. This way, the AWS Support App can access the support cases in that AWS account. If another AWS account in your organization already invited the AWS Support App to that Slack channel, skip to step 3.

> **Note**
>
> You can configure channels that are part of Slack Connect and channels that are shared with multiple workspaces. However, only the first workspace that configured the shared channel for an AWS account can use the AWS Support App. The AWS Support App returns an error message if you try to configure the same Slack channel for another workspace.

To configure a Slack channel

1. From your Slack application, choose the Slack channel that you want to use with the AWS Support App.
2. Complete the following steps to invite the AWS Support App to your channel:
   a. Open the context (right-click) menu on the channel name, and then choose **View channel details**.
   b. Choose the **Integrations** tab, and then choose **Add an App**.
   c. To search for the app, enter **AWS Support App**.
   d. Choose **Add** next to the **AWS Support App**.
3. Sign in to the [Support Center Console] and choose **Slack configuration**.
4. Choose **Add channel**.
5. On the **Add channel** page, under **Workspace**, choose the workspace name that you previously authorized. You can choose the refresh icon if the workspace name doesn't appear in the list.
6. Under **Slack channel**, for **Channel type**, choose one of the following:
   - **Public** – Under **Public channel**, choose the Slack channel that you invited the AWS Support App to (step 2). If your channel doesn't appear in the list, choose the refresh icon and try again.
   - **Private** – Under **Channel ID**, enter the ID or the URL of the Slack channel that you invited the AWS Support App to.

   > **Tip**
   >
   > To find the channel ID, open the context (right-click) menu for the channel name in Slack, and then choose **Copy**, and then choose **Copy link**. Your channel ID is the value that looks like *C01234A5BCD*.

7. Under **Channel configuration name**, enter a name that easily identifies your Slack channel configuration for the AWS Support App. This name appears only in your AWS account and doesn't appear in Slack. You can rename your channel configuration later.

   Your Slack channel type might look like the following example.
8. Under **Permissions**, for **IAM role for the AWS Support App in Slack**, choose a role that you created for the AWS Support App. Only roles that have the AWS Support App as a trusted entity appear in the list.

> **Note**
>
> If you haven't created a role or don't see your role in the list, see Managing access to the AWS Support App.

9. Under **Notifications**, specify how to get notified for cases.
   - **All cases** – Get notified for all case updates.
   - **High-severity cases** – Get notified for only cases that affect a production system or higher. For more information, see Choosing an initial support case severity level.
   - **None** – Don't get notified for case updates.
10. (Optional) If you choose **All cases** or **High-severity cases**, you must select at least one of the following options:
    - **New and reopened cases**
    - **Case correspondences**
    - **Resolved cases**

    The following channel receives case notifications for all case updates in Slack.
11. Review your configuration and choose **Add channel**. Your channel appears in the **Slack configuration** page.

Update your Slack channel configuration

After you configured your Slack channel, you can update them later to change the IAM role or case notification.

To update your Slack channel configuration

1. Sign in to the **Support Center Console** and choose **Slack configuration**.
2. Under **Channels**, choose the channel configuration that you want.
3. On the ***channelName*** page, you can do the following tasks:
   - Choose **Rename** to update your channel configuration name. This name only appears in your AWS account and won't appear in Slack.

- Choose **Delete** to delete the channel configuration from the AWS Support App. See Deleting a Slack channel configuration from the AWS Support App.
- Choose **Open in Slack** to open the Slack channel in your browser.
- Choose **Edit** to change the IAM role or notifications.

Creating support cases in a Slack channel

After you authorize your Slack workspace and add your Slack channel, you can create a support case in your Slack channel.

To create a support case in Slack

1. In your Slack channel, enter the following command:

   ```
   /awssupport create
   ```

2. In the **Create a support case** dialog box, do the following:

   a. If you configured more than one account for this Slack channel, for **AWS account**, choose the account ID. If you created an account name, this value appears next to the account ID. For more information, see Authorize a Slack workspace.

   b. For **Subject**, enter a title for the support case.

   c. For **Description**, describe the support case. Provide details, such as how you're using an AWS service and what troubleshooting steps you tried.

3. Choose **Next**.
4. On the **Create a support case** dialog box, specify the following options:

   a. Choose the **Issue type**.

   b. Choose the **Service**.

   c. Choose the **Category**.

   d. Choose the **Severity**.

   e. Review your case details and choose **Next**.

   The following example shows a technical support case for Alexa Services.

5. For **Contact language**, choose your preferred language for your support case.

> **Note**
>
> Japanese language support isn't available for live chat in Slack for account and billing cases.

6. For **Contact method**, choose **Email and Slack notifications** or **Live chat in Slack**.

   The following example shows how to choose a live chat in Slack.

   a. If you choose **Live chat in Slack**, choose **New private channel** or **Current channel** as your **Live chat channel preference**. **New private channel** will create a separate private channel for you to chat with the AWS Support agent, and **Current channel** will use a thread in the current channel for you to chat with the AWS Support agent.

   b. (Optional) If you choose **Live chat in Slack**, you can enter the names of other Slack members. For **New private channel**, the AWS Support App will automatically add you and selected members to the new channel. For **Current channel**, the AWS Support App will automatically tag you and selected members in the chat thread when the AWS Support agent joins.

   > **Important**
   >
   > - We recommend that you only add chat members that you want to have access to your support case details and chat history.
   > - If you start a new live chat session for an existing support case, the AWS Support App uses the same chat channel or thread that was used for a previous live chat. The AWS Support App also uses the same live chat channel preference that was used previously.
   > - The **Current channel** option is only available if the chat is requested from a private channel. We recommend that you only use this option if you want all channel members to have access to your chat.

7. (Optional) For **Additional contacts to notify**, enter email addresses to also receive updates about this support case. You can add up to 10 email addresses.
8. Choose **Review**.
9. In the Slack channel, review the case details. You can do the following:

- Choose **Edit** to change the case details.
- Add a file to your case. To do so, follow these steps:

  a. Choose **Attach file**, choose the **+** icon in Slack, and choose **Your computer**.

  b. Navigate to and choose your file.

  c. In the **Upload a file** dialog box, enter `@awssupport`, and press the send message  icon.

  > **Notes**
  >
  > - You can attach up to three files. Each file can be up to 5 MB.
  > - If you attach a file to your support case, you must submit your case within 1 hour. If you don't, you must add the files again.

- Choose **Share to channel** to share the case details with others in the Slack channel. You can use this option to share the case details with your team before you create the case.

10. Review your case details, and then choose **Create case**.

    The following example shows a technical support case for Alexa Services.

    After you create a support case, it might take a few minutes for your case details to appear.

11. When your support case is updated, you can choose **See details** to view your case information. You can then do the following:
    - Choose **Share to channel** to share the case details with others in the Slack channel.
    - Choose **Reply** to add a correspondence.
    - Choose **Resolve case**.

    > **Note**
    >
    > If you didn't choose to receive automatic case updates in Slack, you can search for the support case to find the **See details** option.

Replying to support cases in Slack

You can add updates to your case such as case details and attachments, and reply to responses from the support agent.

Note

- You can also use the AWS Support Center Console to reply to support agents. For more information, see Legacy experience: Updating, resolving, and reopening your case.
- You cannot add correspondences to cases from chat channels created by the AWS Support App. Live chat channels only send messages to agents during the live chat.

To reply to a support case in Slack

1. In your Slack channel, choose the case that you want to respond to. You can enter `/awssupport search` to find your support case.
2. Choose **See details** next to the case that you want.
3. At the bottom of the case details, choose **Reply**.

4. In the **Reply to case** dialog box, enter a brief description of the issue in the **Message** field. Then choose **Next**.
5. Choose your contact method. The available contact methods depend on your case type and support plan.
6. (Optional) For **Additional contacts to notify**, enter additional email addresses that you want to receive updates about this support case. You can add up to 10 email addresses.
7. Choose **Review**. You can then choose if you want to edit your reply, attach files, or share to the channel.
8. When you're ready to reply, choose **Send message**.
9. (Optional) To view previous correspondence for your case, choose **Previous correspondence**. To view shortened messages, choose **Show full message**.

Example : Reply to a case in Slack

Join a live chat session with Support

When you request a live chat for your case, you choose to either use a new chat channel or a thread in the current channel for you and the AWS Support agent. Use this chat channel or thread to communicate with the support agent and any others that you invited to the live chat.

Important

Anyone who joins a channel with a live chat can view details about the specific support case and the chat history. Its a best practice to add only users that require access to your support cases. Any member of a chat channel or thread can also participate in an active chat.

Note

Live chat channels and threads also receive notifications when a correspondence is added to the case outside of the live chat session. This occurs before, during, and after a chat session, so you can use a chat channel or thread to monitor all updates for a case. If you chose to use a new chat channel, use the configuration channel that you invited the AWS Support App to reply to these correspondences.

To join a live chat session with Support in a new channel

1. In the Slack application, navigate to the channel that the AWS Support App creates for you. The channel name includes your support case ID, such as *awscase-1234567890*.

   Note

   The AWS Support App adds a pinned message to the live chat channel that contains details about your support case. From the pinned message, you can end the chat or resolve the case. You can find all pinned messages in this channel under the channel name.

2. When the support agent joins the channel, you can chat about your support case. Until a support agent joins the channel, the agent won't see messages in that chat and the messages don't appear in your case correspondence.
3. (Optional) Add other members to the chat channel. By default, chat channels are private.
4. After the support agent joins the chat, the chat channel is active and the AWS Support App records the chat.

   You can chat with the agent about your support case and upload any file attachments to the channel. The AWS Support App automatically saves your files and chat log to your case correspondence.

   > **Note**
   >
   > When you chat with a support agent, note the following differences in Slack for the AWS Support App:
   >
   > - Support agents can't view shared messages or threads. To share text from a message or thread, enter the text as a new message.
   > - If you edit or delete a message, the agent still sees the original message. You must enter your new message again to show the revision.

   Example : Live chat session

   The following is an example of a live chat session with a support agent to fix a connectivity issue for two Amazon Elastic Compute Cloud (Amazon EC2) instances.
5. (Optional) To stop the live chat, choose **End chat**. The support agent leaves the channel and the AWS Support App stops recording the live chat. You can find the chat history attached to the case correspondence for this support case.
6. If the issue is resolved, you can choose **Resolve case** from the pinned message or enter `/awssupport resolve`.

   Example : End a live chat

   The following pinned message shows the case details about an Amazon EC2 instance. You can find the pinned messages under the Slack channel name.

Example : Correspondence notification in chat channel

The following is an example of a live chat channel receiving a notification when the another collaborator adds an update after the chat has ended.

The notification will indicate the chat status (requested, in progress, or ended) and whether the correspondence was added by an agent or by another collaborator. The Support App will also attempt to link back to the original Slack thread or channel where this chat was requested. You can reply to this case from that channel, or any other channel with access to this case.

To join a live chat session with Support in the current channel

1. In the Slack application, navigate to the thread in the current channel that the AWS Support App uses for the chat. In most cases, this will be the thread that started when the case was first created.
2. When the support agent joins the thread, you can chat about your support case. Until a support agent joins the thread, the agent won't see messages in that thread, and the messages won't appear in your case correspondence when the chat ends.

   > **Note**
   >
   > Messages sent to this channel outside of the chat thread are never seen by Support, even while a chat is active.

3. (Optional) Tag other channel members to notify them on the chat thread.
4. After the support agent joins the chat, the chat thread is active and the AWS Support App records the chat. Similar to the new chat channel option, you can chat with the agent about your support case and upload any file attachments to the thread. The AWS Support App automatically saves your files and chat log to your case correspondence.
5. (Optional) To stop the live chat, choose End chat from the initial message for this thread. The support agent leaves the thread and the AWS Support App stops recording the live chat. You can find the chat history attached to the case correspondence for this support case.
6. If the issue is resolved, you can choose Resolve case from the initial message for this thread.

Searching for support cases in Slack

From your Slack channel, you can search for support cases from your AWS account and from other accounts that configured the same channel and workspace. For example, if your account (123456789012) and your coworker's account (111122223333) have configured the same workspace and channels in the AWS Support Center Console, you can use the AWS Support App to search for each other's support cases.

To filter your search results, you can use the following options:

- Account ID
- Case ID
- Case status
- Contact language
- Date range

To search for a support case in Slack

1. In the Slack channel, enter the following command:

   ```
   /awssupport search
   ```

2. For the **I want to search for cases by:** option, choose one of the following:

   A. **Filter options** – You can filter cases with the following options:
   - **AWS account** – This list only appears if you have multiple accounts in the channel.
   - **Date range** – The date the case was created.
   - **Case status** – The current case status, such as **All open cases** or **Resolved**.
   - **Case created in** – The contact language for the case.

   B. **Case ID** – Enter the case ID. You can only enter one case ID at a time. If you have multiple accounts in the channel, choose the AWS account to search for the case.

3. Choose **Search**. Your search results appear in Slack.

Use your search results

After you receive your search results, you can do the following:

To use your search results

1. Choose **Edit Search** to change your previous filter options or case ID.
2. Choose **Share to channel** to share the search results with the channel.
3. Choose **See details** for more information about a case. You can choose **Show full message** to view the rest of the latest correspondence.
4. If you searched by **Filter options**, search results can return multiple cases. Choose **Next 5 results** or **Previous 5 results** to view the next or previous 5 cases.

Resolving a support case in Slack

If you don't need your support case anymore, or you fixed the issue, you can resolve a support case directly in Slack. This also resolves the case in the AWS Support Center Console. After you resolve a case, you can reopen the case later.

To resolve a support case in Slack

1. In your Slack channel, navigate to the support case. See Searching for support cases in Slack.
2. Choose **See details** for the case.
3. Choose **Resolve case**.
4. In the **Resolve case** dialog box, choose **Resolve case**. You can reopen a case in the Slack channel or from the Support Center Console.

Reopening a support case in Slack

After you resolve a support case, you can reopen the case from Slack.

To reopen a support case in Slack

1. Find the support case to reopen in Slack. See Searching for support cases in Slack.
2. Choose **See details**.
3. Choose **Reopen case**.
4. In the **Reopen case** dialog box, enter a brief description of the issue in the **Message** field.
5. Choose **Next**.
6. (Optional) Enter additional contacts.

7. Choose **Review**.
8. Review your case details, and then choose **Send message**. Your case reopens. If you requested a new live chat with a support agent, Slack uses the same chat channel or thread as the one that was used for a previous live chat. If you requested a live chat in a new channel and you haven't had one so far, a new chat channel opens. If you requested a live chat in the current channel and you haven't had one so far, a thread in the current channel is used.

Deleting a Slack channel configuration from the AWS Support App

You can delete a channel configuration from the AWS Support App if you don't need it. This action only removes the channel from the AWS Support App and the AWS Support Center Console. Your channel isn't deleted from Slack.

You can add up to 20 channels for your AWS account. If you already reached this quota, you must delete a channel before you can add another one.

To delete a Slack channel configuration

1. Sign in to the **Support Center Console** and choose **Slack configuration**.
2. On the **Slack configuration** page, under **Channels**, choose the channel name, and then choose **Delete**.
3. In the **Delete channel name** dialog box, choose **Delete**. You can add this channel to the AWS Support App again later.

Deleting a Slack workspace configuration from the AWS Support App

You can delete a workspace configuration from the AWS Support App if you don't need it. This action only removes the workspace from the AWS Support App and the AWS Support Center Console. Your workspace isn't deleted from Slack.

You can add up to 5 workspaces for your AWS account. If you already reached this quota, you must delete a Slack workspace before you can add another one.

> **Note**
>
> If you added channels from this workspace to the AWS Support App, you must first delete these channels before you can delete the workspace. See Deleting a Slack channel configuration from the AWS Support App.

To delete a Slack workspace configuration

1. Sign in to the **AWS Support Center Console** and choose **Slack configuration**.
2. On the **Slack configuration** page, under **Slack workspaces**, choose **Delete a workspace**.
3. In the **Delete Slack workspace** dialog box, choose the Slack workspace name, and then choose **Delete**. You can add the workspace to your AWS account again later.

AWS Support App in Slack commands

Slack channel commands

You can enter the following commands in the Slack channel where you invited the AWS Support App. This Slack channel name also appears as a configured channel in the AWS Support Center Console.

`/awssupport create` or `/awssupport create-case`

Create a support case.

`/awssupport search` or `/awssupport search-case`

Search for cases. You can search for support cases for the AWS accounts that configured the AWS Support App for the same Slack channel.

`/awssupport quota` or `/awssupport service-quota-increase`

Request a service quota increase.

Live chat channel commands

You can enter the following commands in the live chat channel. This is the channel that the AWS Support App creates for you if you choose a new channel for your chat with Support. Chat channels include your support case ID, such as *awscase-1234567890*.

> **Note**
>
> The following commands are not available when using a thread in the current channel for a live chat. Instead, use the buttons attached to the initial thread message to end a chat, invite a new agent, or resolve the case.

`/awssupport endchat`

Remove the support agent and end the live chat session.

`/awssupport invite`

Invite a new support agent to this channel.

`/awssupport resolve`

Resolve this support case.

View AWS Support App correspondences in the AWS Support Center Console

When you create, update, or resolve support cases for your account in the Slack channel, you can also sign in to the Support Center Console to view your cases. You can view the case correspondences to determine whether the case was updated in the Slack channel, view the chat history with a support agent, and find any attachments that you uploaded from Slack.

To view case correspondences from Slack

1. Sign in to the **AWS Support Center Console** for your account.
2. Choose your support case.
3. In the **Correspondence**, you can view whether the case was created and updated from the Slack channel.

Example : Support case

In the following screenshot, Jane Doe reopened a support case in Slack. This correspondence appears for the support case in the Support Center Console.

Creating AWS Support App in Slack resources with AWS CloudFormation

AWS Support App in Slack is integrated with AWS CloudFormation, a service that helps you to model and set up your AWS resources so that you can spend less time creating and managing your resources and infrastructure. You create a template that describes all the AWS resources that you want (such as your AccountAlias and SlackChannelConfiguration), and CloudFormation provisions and configures those resources for you.

When you use CloudFormation, you can reuse your template to set up your AWS Support App resources consistently and repeatedly. Describe your resources once, and then provision the same resources over and over in multiple AWS accounts and Regions.

AWS Support App and CloudFormation templates

To provision and configure resources for AWS Support App and related services, you must understand CloudFormation templates. Templates are formatted text files in JSON or YAML. These templates describe the resources that you want to provision in your CloudFormation stacks. If you're unfamiliar with JSON or YAML, you can use CloudFormation Designer to help you get started with CloudFormation templates. For more information, see What is CloudFormation Designer? in the *AWS CloudFormation User Guide*.

AWS Support App supports creating your AccountAlias and SlackChannelConfiguration in CloudFormation. For more information, including examples of JSON and YAML templates for the AccountAlias and SlackChannelConfiguration resources, see the AWS Support App resource type reference in the *AWS CloudFormation User Guide*.

Create Slack configuration resources for your organization

You can use CloudFormation templates to create the resources that you need for the AWS Support App. If you're the management account for your organization, you can use the templates to create these resources for your member accounts in AWS Organizations.

For example, you might use a template to create the same Slack workspace configuration for all accounts in the organization, but then use separate templates to create different Slack channel configurations for specific AWS accounts or organizational units (OUs). You can also use a template to create a Slack workspace configuration so that member accounts can then configure the Slack channels that they want for their AWS accounts.

You can choose whether to use CloudFormation templates or not. If you don't use CloudFormation templates, you can complete the following manual steps instead:

- Create the AWS Support App resources in the AWS Support Center Console.
- Create a support case with AWS Support to authorize multiple accounts to use the AWS Support App.
- Call the RegisterSlackWorkspaceForOrganization API operation to register a Slack workspace for your account. The CloudFormation stack calls this API operation for you.

Follow these procedures to upload the CloudFormation template to your organization. You can use the example templates from the AWS Support App resource type reference page.

The templates tell CloudFormation to create the following resources:

- A Slack channel configuration.
- A Slack workspace configuration.
- An IAM role with the `AWSSupportSlackAppCFNRole` name. The AWSSupportAppFullAccess AWS managed policy is attached.

Contents

Update your CloudFormation templates for Slack

To get started, use the following templates to create your stack. You must replace the templates with valid values for your Slack workspace and channel.

> **Note**
>
> We don't recommend the use of the template to create an AccountAlias resource for your organization. The AccountAlias resource uniquely identifies an AWS account in the AWS Support App. Your member accounts can enter an account name in the Support Center Console. For more information, see Authorize a Slack workspace.

To update your CloudFormation templates for Slack

1. If you're the management account for an organization, you must manually authorize a Slack workspace for your account before your member accounts can use CloudFormation to create the resources. If you haven't already done so, see Authorize a Slack workspace.
2. From the AWS Support App resource type reference page, copy the JSON or YAML template for the resource that you want.
3. In a text editor, paste the template into a new file.
4. In the template, specify the parameters that you want. At a minimum, replace the values for the following fields:

   - `TeamId` with your Slack workspace ID
   - `ChannelId` with the Slack channel ID
   - `ChannelName` with a name to identify the Slack channel configuration

   > **Tip**
   >
   > To find the workspace and channel IDs, open your Slack channel in a browser. In the URL, your workspace ID is the first identifier and the channel ID is the second. For example, in https://app.slack.com/client/T012ABCDEFG/C01234A5BCD, T012ABCDEFG is the workspace ID and C01234A5BCD is the channel ID.

5. Save the file as either a JSON or YAML file.

Create a stack for the management account

Next, you must create a stack for the management account in the organization. This step calls the RegisterSlackWorkspaceForOrganization API operation for you and authorizes the workspace with Slack.

> **Note**
>
> We recommend that you upload the Slack workspace configuration template that you updated in the previous procedure for the management account. You don't need to upload the Slack channel configuration template unless you're also configuring the management account to use the AWS Support App.

To create a stack for the management account

1. Sign in to the AWS Management Console as the management account for your organization.
2. Open the CloudFormation console at https://eusc-de-east-1.console.amazonaws-eusc.eu/cloudformation.
3. If you haven't already, in the **Region selector**, choose one of the following AWS Regions:
   - Europe (Frankfurt)
   - Europe (Ireland)
   - Europe (London)
   - US East (N. Virginia)
   - US East (Ohio)
   - US West (Oregon)
   - Asia Pacific (Singapore)
   - Asia Pacific (Tokyo)
   - Canada (Central)
4. Follow the procedure to create a stack. For more information, see Creating a stack on the CloudFormation console.

   After CloudFormation successfully creates the stack, you can use the same template to create a stack set for your organization.

Create a stack set for your organization

Next, use the same template for the Slack workspace configuration to create a stack set with `service-managed` permissions. You can use stack sets to create the stack for your entire organization or specify the OUs that you want. For more information, see Create a stack set.

This procedure also calls the RegisterSlackWorkspaceForOrganization API operation for you. This API operation authorizes the workspace with Slack for the member accounts.

To create a stack set for your organization

1. Sign in to the AWS Management Console as the management account for your organization.
2. Open the CloudFormation console at https://eusc-de-east-1.console.amazonaws-eusc.eu/cloudformation.
3. If you haven't already, in the **Region selector**, choose the same AWS Region that you used in the previous procedure.
4. In the navigation pane, choose **StackSets**.
5. Choose **Create StackSet**.
6. On the **Choose a template** page, keep the default options for the following options:
   - For **Permissions**, keep **Service-managed permissions**.
   - For **Prerequisite - Prepare template**, keep **Template is ready**.
7. Under **Specify template**, choose **Upload a template file**, and then choose **Choose file**.
8. Choose the file and then choose **Next**.
9. On the **Specify StackSet details** page, enter a stack name such as **`support-app-slack-workspace`**, enter a description, and then choose **Next**.
10. On the **Configure StackSet options** page, keep the default options and then choose **Next**.
11. On the **Set deployment options** page, for **Add stacks to stack set**, keep the default **Deploy new stacks** option.
12. For **Deployment targets**, choose if you want to create the stack for the entire organization or specific OUs. If you choose an OU, enter the OU ID.
13. For **Specify regions**, enter only *one* of the following AWS Regions:
    - Europe (Frankfurt)
    - Europe (Ireland)

- Europe (London)
- US East (N. Virginia)
- US East (Ohio)
- US West (Oregon)
- Asia Pacific (Singapore)
- Asia Pacific (Tokyo)
- Canada (Central)

Notes:

- To streamline your workflow, we recommend that you use the same AWS Region that you chose in step 3.
- Choosing more than one AWS Region can cause conflicts with creating your stack.

14. For **Deployment options**, for **Failure tolerance - optional**, enter the number of accounts where the stacks can fail before CloudFormation stops the operation. We recommend that you enter the number of accounts that you want to add, minus one. For example, if your specified OU has 10 member accounts, enter 9. This means that even if CloudFormation fails the operation 9 times, at least one account will succeed.
15. Choose **Next**.
16. On the **Review** page, review your options, and then choose **Submit**. You can check the status of your stack on the **Stack instances** tab.
17. (Optional) Repeat this procedure to upload a template for a Slack channel configuration. The example template also creates the IAM role and attaches an AWS managed policy. This role has the required permissions to access other services for you. For more information, see Managing access to the AWS Support App.

    If you don't create a stack set to create the Slack channel configuration, your member accounts can manually configure the Slack channel. For more information, see Configuring a Slack channel.

After CloudFormation creates the stacks, each member account can sign in to the Support Center Console and find their configured Slack workspaces and channels. They can then use the AWS Support App for their AWS account. See Creating support cases in a Slack channel.

Tip

If you need to upload a new template, we recommend that you use the same AWS Region that you specified before.

Learn more about CloudFormation

To learn more about CloudFormation, see the following resources:

- AWS CloudFormation
- AWS CloudFormation User Guide
- CloudFormation API Reference
- AWS CloudFormation Command Line Interface User Guide

Create AWS Support App resources by using Terraform

You can also use Terraform to create the AWS Support App resources for your AWS account. Terraform is an infrastructure-as-code tool that you can use for your cloud applications. You can use Terraform to create AWS Support App resources instead of deploying a CloudFormation stack to an account.

After you install Terraform, you can specify the AWS Support App resources that you want. Terraform calls the RegisterSlackWorkspaceForOrganization API operation to register a Slack workspace for you and creates your resources. You can then sign in the Support Center Console and find your configured Slack workspaces and channels.

Notes

- If you're the management account for an organization, you must manually authorize a Slack workspace for your account before your member accounts can use Terraform to create the resources. If you haven't already done so, see Authorize a Slack workspace.
- Unlike CloudFormation stack sets, you can't use Terraform to create the AWS Support App resources for an OU in your organization.
- You can also find the event history for these updates from Terraform in AWS CloudTrail. The `eventSource` for these events will be `cloudcontrolapi.amazonaws.com` and

`supportapp.amazonaws.com`. For more information, see Logging AWS Support App in Slack API calls using AWS CloudTrail.

Learn more

To learn more about Terraform, see the following topics:

- Terraform installation
- Terraform tutorial: Build infrastructure for AWS
- `awscc_support_app_account_alias`
- `awscc_supportapp_slack_workspace_configuration`
- `awscc_supportapp_slack_channel_configuration`

Security in AWS Support

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

Security is a shared responsibility between AWS and you. The 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 .
- **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 Support. The following topics show you how to configure Support to meet your security and compliance objectives. You also learn how to use other Amazon Web Services that help you to monitor and secure your Support resources.

Topics

Data protection in AWS Support

The AWS applies to data protection in Support. 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. You are also responsible for 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 data protection purposes, we recommend that you protect AWS account credentials and set up individual users with AWS IAM Identity Center or 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 require TLS 1.2 and recommend TLS 1.3.
- Set up API and user activity logging with AWS CloudTrail. For information about using CloudTrail trails to capture AWS activities, see Working with CloudTrail trails in the *AWS CloudTrail User Guide*.
- 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 sensitive data that is stored in Amazon S3.
- If you require FIPS 140-3 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) 140-3.

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form text fields such as a **Name** field. This includes when you work with AWS Support or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form text 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.

Important

In the case correspondence, never share sensitive information, such as credentials, credit cards, signed URLs, or personally identifiable information.

Security for your AWS Support cases

When you create a support case, you own the information that you include in your support case. AWS doesn't access your AWS account data without your permission. AWS doesn't share your information with third parties.

When you create a support case, note the following:

- AWS Support uses the permissions defined in the `AWSServiceRoleForSupport` service-linked role to call other AWS services that troubleshoot customer issues for you. For more information, see Using service-linked roles for AWS Support and AWS managed policy: AWSSupportServiceRolePolicy.
- You can view API calls to AWS Support that occurred in your AWS account. For example, you can view log information when someone in your account creates or resolves a support case. For more information, see Logging AWS Support API calls with AWS CloudTrail.
- You can use the AWS Support API to call the `DescribeCases` API. This API returns support case information, such as the case ID, the create and resolve date, and correspondences with the support agent. You can view case details for up to 24 months after the case was created. For more information, see DescribeCases in the *AWS Support API Reference*.
- Your support cases follow Compliance validation for AWS Support.
- When you create a support case, AWS doesn't gain access your account. If necessary, support agents use a screen-sharing tool to view your screen remotely and identify and troubleshoot problems. This tool is view-only. AWS Support can't act for you during the screen-share session. You must give consent to share a screen with a support agent. For more information, see the AWS Support FAQs.
- You can change your AWS Support plan to get the help that you need for your account. For more information, see Compare AWS Support Plans and Changing your AWS Support plan.

Identity and access management for AWS Support

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 Support 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 based on your role:

- **Service user** - request permissions from your administrator if you cannot access features (see Troubleshooting AWS Support identity and access)
- **Service administrator** - determine user access and submit permission requests (see How AWS Support works with IAM)
- **IAM administrator** - write policies to manage access (see AWS Support identity-based policy examples)

Authenticating with identities

Authentication is how you sign in to AWS using your identity credentials. You must be authenticated as the AWS account root user, an IAM user, or by assuming an IAM role.

You can sign in as a federated identity using credentials from an identity source like AWS IAM Identity Center (IAM Identity Center), single sign-on authentication, or Google/Facebook credentials. For more information about signing in, see How to sign in to your AWS account in the *AWS Sign-In User Guide*.

For programmatic access, AWS provides an SDK and CLI to cryptographically sign requests. For more information, see AWS Signature Version 4 for API requests in the *IAM User Guide*.

AWS account root user

When you create an AWS account, you begin with one sign-in identity called the AWS account *root user* that has complete access to all AWS services and resources. We strongly recommend that you don't use the root user for everyday tasks. For tasks that require root user credentials, see Tasks that require root user credentials in the *IAM User Guide*.

IAM users and groups

An *IAM user* is an identity with specific permissions for a single person or application. We recommend using temporary credentials instead of IAM users with long-term credentials. For more information, see Require human users to use federation with an identity provider to access AWS using temporary credentials in the *IAM User Guide*.

An *IAM group* specifies a collection of IAM users and makes permissions easier to manage for large sets of users. For more information, see Use cases for IAM users in the *IAM User Guide*.

IAM roles

An *IAM role* is an identity with specific permissions that provides temporary credentials. You can assume a role by switching from a user to an IAM role (console) or by calling an AWS CLI or AWS API operation. For more information, see Methods to assume a role in the *IAM User Guide*.

IAM roles are useful for federated user access, temporary IAM user permissions, cross-account access, cross-service access, and applications running on Amazon EC2. For more information, see Cross account resource access in IAM in the *IAM User Guide*.

Managing access using policies

You control access in AWS by creating policies and attaching them to AWS identities or resources. A policy defines permissions when associated with an identity or resource. AWS evaluates these policies when a principal makes a request. Most policies are stored in AWS as JSON documents. For more information about JSON policy documents, see Overview of JSON policies in the *IAM User Guide*.

Using policies, administrators specify who has access to what by defining which **principal** can perform **actions** on what **resources**, and under what **conditions**.

By default, users and roles have no permissions. An IAM administrator creates IAM policies and adds them to roles, which users can then assume. IAM policies define permissions regardless of the method used to perform the operation.

Identity-based policies

Identity-based policies are JSON permissions policy documents that you attach to an identity (user, group, or role). These policies control what actions identities can perform, on which resources, and under what conditions. To learn how to create an identity-based policy, see Define custom IAM permissions with customer managed policies in the *IAM User Guide*.

Identity-based policies can be *inline policies* (embedded directly into a single identity) or *managed policies* (standalone policies attached to multiple identities). To learn how to choose between managed and inline policies, see Choose between managed policies and inline policies in the *IAM User Guide*.

Other policy types

AWS supports additional policy types that can set the maximum permissions granted by more common policy types:

- **Permissions boundaries** – Set the maximum permissions that an identity-based policy can grant to an IAM entity. For more information, see Permissions boundaries for IAM entities in the *IAM User Guide*.
- **Service control policies (SCPs)** – Specify the maximum permissions for an organization or organizational unit in AWS Organizations. For more information, see Service control policies in the *AWS Organizations User Guide*.
- **Resource control policies (RCPs)** – Set the maximum available permissions for resources in your accounts. For more information, see Resource control policies (RCPs) in the *AWS Organizations User Guide*.
- **Session policies** – Advanced policies passed as a parameter when creating a temporary session for a role or federated user. 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 Support works with IAM

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

For information about how to manage access for Support using IAM, see Manage access for Support.

Topics

- Support identity-based policies
- Support IAM roles

Support identity-based policies

With IAM identity-based policies, you can specify allowed or denied actions and resources as well as the conditions under which actions are allowed or denied. Support supports specific actions. To learn about the elements that you use in a JSON policy, see IAM JSON policy elements reference in the *IAM User Guide*.

Actions

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

The `Action` element of a JSON policy describes the actions that you can use to allow or deny access in a policy. Include actions in a policy to grant permissions to perform the associated operation.

Policy actions in Support use the following prefix before the action: `support:`. For example, to grant someone permission to run an Amazon EC2 instance with the Amazon EC2 `RunInstances` API operation, you include the `ec2:RunInstances` action in their policy. Policy statements must include either an `Action` or `NotAction` element. Support defines its own set of actions that describe tasks that you can perform with this service.

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

```
"Action": [
      "ec2:action1",
```

```
      "ec2:action2"
```

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

```
"Action": "ec2:Describe*"
```

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

Examples

To view examples of Support identity-based policies, see AWS Support identity-based policy examples.

Support IAM roles

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

Using temporary credentials with Support

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

Support supports using temporary credentials.

Service-linked roles

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

Support supports service-linked roles. For details about creating or managing Support service-linked roles, see Using service-linked roles for AWS Support.

Service roles

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

Support supports service roles.

AWS Support identity-based policy examples

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

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

Topics

Policy best practices

Identity-based policies are very powerful. They determine whether someone can create, access, or delete Support resources in your account. When you create or edit identity-based policies, follow these guidelines and recommendations:

- **Get Started Using AWS Managed Policies** – To start using Support 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 Support console

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

To be sure that those entities can still use the Support console, also attach the following AWS managed policy to the entities. For more information, see Adding permissions to a user in the *IAM User Guide*:

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

Allow users to view their own permissions

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

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
```

```
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Using service-linked roles

AWS Support and AWS Trusted Advisor use AWS Identity and Access Management (IAM) service-linked roles. A service-linked role is a unique IAM role that is linked directly to Support and Trusted Advisor. In each case, the service-linked role is a predefined role. This role includes all the permissions that Support or Trusted Advisor require to call other AWS services on your behalf. The following topics explain what service-linked roles do and how to work with them in Support and Trusted Advisor.

Topics

- Using service-linked roles for AWS Support
- Using service-linked roles for Trusted Advisor

Using service-linked roles for AWS Support

AWS Support tools gather information about your AWS resources through API calls to provide customer service and technical support. To increase the transparency and auditability of support activities, Support uses an AWS Identity and Access Management (IAM) service-linked role.

The `AWSServiceRoleForSupport` service-linked role is a unique IAM role that is linked directly to Support. This service-linked role is predefined, and it includes the permissions that Support requires to call other AWS services on your behalf.

The `AWSServiceRoleForSupport` service-linked role trusts the `support.amazonaws.com` service to assume the role.

To provide these services, the role's predefined permissions give Support access to resource metadata, not customer data. Only Support tools can assume this role, which exists within your AWS account.

We redact fields that could contain customer data. For example, the `Input` and `Output` fields of the GetExecutionHistory for the AWS Step Functions API call aren't visible to Support. We use AWS KMS keys to encrypt sensitive fields. These fields are redacted in the API response and aren't visible to AWS Support agents.

> **Note**
>
> AWS Trusted Advisor uses a separate IAM service-linked role to access AWS resources for your account to provide best practice recommendations and checks. For more information, see Using service-linked roles for Trusted Advisor.

The `AWSServiceRoleForSupport` service-linked role enables all AWS Support API calls to be visible to customers through AWS CloudTrail. This helps with monitoring and auditing requirements, because it provides a transparent way to understand the actions that Support performs on your behalf. For information about CloudTrail, see the AWS CloudTrail User Guide.

Service-linked role permissions for Support

This role uses the `AWSSupportServiceRolePolicy` AWS managed policy. This managed policy is attached to the role and allows the role permission to complete actions on your behalf.

These actions might include the following:

- **Billing, administrative, support, and other customer services** – AWS customer service uses the permissions granted by the managed policy to perform a number of services as part of your support plan. These include investigating and answering account and billing questions, providing administrative support for your account, increasing service quotas, and offering additional customer support.

- **Processing of service attributes and usage data for your AWS account** – Support might use the permissions granted by the managed policy to access service attributes and usage data for your AWS account. This policy allows Support to provide billing, administrative, and technical support for your account. Service attributes include your account's resource identifiers, metadata tags, roles, and permissions. Usage data includes usage policies, usage statistics, and analytics.
- **Maintaining the operational health of your account and its resources** – Support uses automated tools to perform actions related to operational and technical support.

For more information about the allowed services and actions, see the AWSSupportServiceRolePolicy policy in the IAM console.

> **Note**
>
> AWS Support automatically updates the `AWSSupportServiceRolePolicy` policy once per month to add permissions for new AWS services and actions.

For more information, see AWS managed policies for AWS Support.

Creating a service-linked role for Support

You don't need to manually create the `AWSServiceRoleForSupport` role. When you create an AWS account, this role is automatically created and configured for you.

> **Important**
>
> If you used Support before it began supporting service-linked roles, then AWS created the `AWSServiceRoleForSupport` role in your account. For more information, see A new role appeared in my IAM account.

Editing and deleting a service-linked role for Support

You can use IAM to edit the description for the `AWSServiceRoleForSupport` service-linked role. For more information, see Editing a service-linked role in the *IAM User Guide*.

The `AWSServiceRoleForSupport` role is necessary for Support to provide administrative, operational, and technical support for your account. As a result, this role can't be deleted through

the IAM console, API, or AWS Command Line Interface (AWS CLI). This protects your AWS account, because you can't inadvertently remove necessary permissions for administering support services.

Customers onboarded to AWS Organizations and who have an Enterprise Support plan can delete the `AWSServiceRoleForSupport` service-linked role. Deleting this role restricts access to your resources by AWS Support engineers, limiting their ability to perform actions on your behalf. For more information, or to request to delete the `AWSServiceRoleForSupport` service-linked role, contact your Technical Account Manager (TAM).

For more information about the `AWSServiceRoleForSupport` role or its uses, contact Support.

Using service-linked roles for Trusted Advisor

AWS Trusted Advisor uses the AWS Identity and Access Management (IAM) service-linked role. A service-linked role is a unique IAM role that is linked directly to AWS Trusted Advisor. Service-linked roles are predefined by Trusted Advisor, and they include all the permissions that the service requires to call other AWS services on your behalf. Trusted Advisor uses this role to check your usage across AWS and to provide recommendations to improve your AWS environment. For example, Trusted Advisor analyzes your Amazon Elastic Compute Cloud (Amazon EC2) instance use to help you reduce costs, increase performance, tolerate failures, and improve security.

> **Note**
>
> AWS Support uses a separate IAM service-linked role for accessing your account's resources to provide billing, administrative, and support services. For more information, see Using service-linked roles for AWS Support.

For information about other services that support service-linked roles, see AWS services that work with IAM. 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.

Topics

- Service-linked role permissions for Trusted Advisor
- Manage permissions for service-linked roles
- Creating a service-linked role for Trusted Advisor
- Editing a service-linked role for Trusted Advisor
- Deleting a service-linked role for Trusted Advisor

Service-linked role permissions for Trusted Advisor

Trusted Advisor uses two service-linked roles:

- AWSServiceRoleForTrustedAdvisor – This role trusts the Trusted Advisor service to assume the role to access AWS services on your behalf. The role permissions policy allows Trusted Advisor read-only access for all AWS resources. This role simplifies getting started with your AWS account, because you don't have to add the necessary permissions for Trusted Advisor. When you open an AWS account, Trusted Advisor creates this role for you. The defined permissions include the trust policy and the permissions policy. You can't attach the permissions policy to any other IAM entity.

  For more information about the attached policy, see AWSTrustedAdvisorServiceRolePolicy.
- AWSServiceRoleForTrustedAdvisorReporting – This role trusts the Trusted Advisor service to assume the role for the organizational view feature. This role enables Trusted Advisor as a trusted service in your AWS Organizations organization. Trusted Advisor creates this role for you when you enable organizational view.

  For more information about the attached policy, see AWSTrustedAdvisorReportingServiceRolePolicy.

  You can use the organizational view to create reports for Trusted Advisor check results for all accounts in your organization. For more information about this feature, see Organizational view for AWS Trusted Advisor.

Manage permissions for service-linked roles

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. The following examples use the `AWSServiceRoleForTrustedAdvisor` service-linked role.

Example : Allow an IAM entity to create the `AWSServiceRoleForTrustedAdvisor` service-linked role

This step is necessary only if the Trusted Advisor account is disabled, the service-linked role is deleted, and the user must recreate the role to reenable Trusted Advisor.

You can add the following statement to the permissions policy for the IAM entity to create the service-linked role.

```
{
    "Effect": "Allow",
    "Action": [
        "iam:CreateServiceLinkedRole",
        "iam:PutRolePolicy"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/trustedadvisor.amazonaws.com/
AWSServiceRoleForTrustedAdvisor*",
    "Condition": {"StringLike": {"iam:AWSServiceName": "trustedadvisor.amazonaws.com"}}
}
```

Example : Allow an IAM entity to edit the description of the `AWSServiceRoleForTrustedAdvisor` service-linked role

You can only edit the description for the `AWSServiceRoleForTrustedAdvisor` role. You can add the following statement to the permissions policy for the IAM entity to edit the description of a service-linked role.

```
{
    "Effect": "Allow",
    "Action": [
        "iam:UpdateRoleDescription"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/trustedadvisor.amazonaws.com/
AWSServiceRoleForTrustedAdvisor*",
    "Condition": {"StringLike": {"iam:AWSServiceName": "trustedadvisor.amazonaws.com"}}
}
```

Example : Allow an IAM entity to delete the `AWSServiceRoleForTrustedAdvisor` service-linked role

You can add the following statement to the permissions policy for the IAM entity to delete a service-linked role.

```
{
    "Effect": "Allow",
    "Action": [
        "iam:DeleteServiceLinkedRole",
        "iam:GetServiceLinkedRoleDeletionStatus"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/trustedadvisor.amazonaws.com/
AWSServiceRoleForTrustedAdvisor*",
```

```
    "Condition": {"StringLike": {"iam:AWSServiceName": "trustedadvisor.amazonaws.com"}}
}
```

You can also use an AWS managed policy, such as AdministratorAccess, to provide full access to Trusted Advisor.

Creating a service-linked role for Trusted Advisor

You don't need to manually create the `AWSServiceRoleForTrustedAdvisor` service-linked role. When you open an AWS account, Trusted Advisor creates the service-linked role for you.

> **Important**
>
> If you were using the Trusted Advisor service before it began supporting service-linked roles, then Trusted Advisor already created the `AWSServiceRoleForTrustedAdvisor` role in your account. To learn more, see A new role appeared in my IAM account in the *IAM User Guide*.

If your account doesn't have the `AWSServiceRoleForTrustedAdvisor` service-linked role, then Trusted Advisor won't work as expected. This can happen if someone in your account disabled Trusted Advisor and then deleted the service-linked role. In this case, you can use IAM to create the `AWSServiceRoleForTrustedAdvisor` service-linked role, and then reenable Trusted Advisor.

To enable Trusted Advisor (console)

1. Use the IAM console, AWS CLI, or the IAM API to create a service-linked role for Trusted Advisor. For more information, see Creating a service-linked role.
2. Sign in to the AWS Management Console, and then navigate to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor.

   The **Disabled Trusted Advisor** status banner appears in the console.
3. Choose **Enable Trusted Advisor Role** from the status banner. If the required `AWSServiceRoleForTrustedAdvisor` isn't detected, the disabled status banner remains.

Editing a service-linked role for Trusted Advisor

You can't change the name of a service-linked role because various entities might reference the role. However, you can use the IAM console, AWS CLI, or the IAM API to edit the description of the role. For more information, see Editing a service-linked role in the *IAM User Guide*.

Deleting a service-linked role for Trusted Advisor

If you don't need to use the features or services of Trusted Advisor, you can delete the `AWSServiceRoleForTrustedAdvisor` role. You must disable Trusted Advisor before you can delete this service-linked role. This prevents you from removing permissions required by Trusted Advisor operations. When you disable Trusted Advisor, you disable all service features, including offline processing and notifications. Also, if you disable Trusted Advisor for a member account, then the separate payer account is also affected, which means you won't receive Trusted Advisor checks that identify ways to save costs. You can't access the Trusted Advisor console. API calls to Trusted Advisor return an access denied error.

You must recreate the `AWSServiceRoleForTrustedAdvisor` service-linked role in the account before you can reenable Trusted Advisor.

You must first disable Trusted Advisor in the console before you can delete the `AWSServiceRoleForTrustedAdvisor` service-linked role.

To disable Trusted Advisor

1. Sign in to the AWS Management Console and navigate to the Trusted Advisor console at https://eusc-de-east-1.console.amazonaws-eusc.eu/trustedadvisor.
2. In the navigation pane, choose **Preferences**.
3. In the **Service Linked Role Permissions** section, choose **Disable Trusted Advisor**.
4. In the confirmation dialog box, choose **OK** to confirm that you want to disable Trusted Advisor.

After you disable Trusted Advisor, all Trusted Advisor functionality is disabled, and the Trusted Advisor console displays only the disabled status banner.

You can then use the IAM console, the AWS CLI, or the IAM API to delete the Trusted Advisor service-linked role named `AWSServiceRoleForTrustedAdvisor`. For more information, see Deleting a service-linked role in the *IAM User Guide*.

AWS managed policies for AWS Support

An AWS managed policy is a standalone policy that is created and administered by AWS. AWS managed policies are designed to provide permissions for many common use cases so that you can start assigning permissions to users, groups, and roles.

Keep in mind that AWS managed policies might not grant least-privilege permissions for your specific use cases because they're available for all AWS customers to use. We recommend that you reduce permissions further by defining customer managed policies that are specific to your use cases.

You cannot change the permissions defined in AWS managed policies. If AWS updates the permissions defined in an AWS managed policy, the update affects all principal identities (users, groups, and roles) that the policy is attached to. AWS is most likely to update an AWS managed policy when a new AWS service is launched or new API operations become available for existing services.

For more information, see AWS managed policies in the *IAM User Guide*.

Topics

AWS managed policies for AWS Support

AWS Support has the following managed policies.

Contents

AWS managed policy: AWSSupportAccess

AWS Support uses the AWSSupportAccess AWS managed policy. This policy manages your support case lifecycle through the Support API. Enhanced functionality in the AWS Support Center Console is provided through the support-console API service. You can attach this policy to your IAM entities. For more information, see Service-linked role permissions for Support.

To view the permissions for this policy, see AWSSupportAccess in the *AWS Managed Policy Reference*.

AWS managed policy: AWSSupportServiceRolePolicy

AWS Support uses the AWSSupportServiceRolePolicy AWS managed policy. This managed policy is attached to the `AWSServiceRoleForSupport` service-linked role. The policy allows the service-linked role to complete actions on your behalf. You can't attach this policy to your IAM entities. For more information, see Service-linked role permissions for Support.

To view the permissions for this policy, see AWSSupportServiceRolePolicy in the *AWS Managed Policy Reference*.

For a list of changes to the policy, see AWS Support updates to AWS managed policies and Permission changes for AWSSupportServiceRolePolicy.

AWS Support updates to AWS managed policies

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

The following table describes important updates to the AWS Support managed policies since February 17, 2022.

AWS Support

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 145 new permissio ns to the following services	December 08, 2025

Change	Description	Date
	to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • AWS Amplify – To troublesh oot issues related to AWS Amplify. • AWS AppSync – To debug issues related to AWS AppSync. • AWS Outposts – To debug issues related to AWS Outposts. • AWS Clean Rooms – To troubleshoot issues related to AWS Clean Rooms. • AWS Compute Optimizer – To debug issues related to AWS Compute Optimizer. • Amazon Connect – To debug issues related to Amazon Connect. • Amazon DynamoDB – To debug issues related to Amazon DynamoDB. • Amazon EMR – To troublesh oot issues related to Amazon EMR. • Amazon Location Service – To troubleshoot issues related to Amazon Location Service;.	

Change	Description	Date
	• Amazon GuardDuty – To debug issues related to Amazon GuardDuty. • AWS Network Firewall – To debug issues related to AWS Network Firewall. • AWS HealthOmics – To troubleshoot issues related to AWS HealthOmics. • AWS Organizations – To debug issues related to AWS Organizations. • Amazon S3 – To debug issues related to Amazon S3. • Amazon S3 Tables – To debug issues related to Amazon S3 Tables. • Amazon S3 Vectors – To debug issues related to Amazon S3 Vectors. • Amazon SageMaker AI – To troubleshoot issues related to Amazon SageMaker AI. • AWS Security Hub CSPM – To troubleshoot issues related to AWS Security Hub CSPM. • Amazon SES – To debug issues related to Amazon SES.	

Change	Description	Date
	• AWS Signer – To debug issues related to AWS Signer. • AWS STS – To troubleshoot issues related to AWS STS.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 125 new permissio ns to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • Artificial intelligence for IT operations(AIOps) – To debug issues related to Artificial intelligence for IT operations(AIOps). • AWS Backup gateway – To troubleshoot issues related to AWS Backup gateway. • CloudFormation – To debug issues related to CloudForm ation. • Amazon Cognito Identity – To debug issues related to Amazon Cognito Identity. • Amazon Cognito – To troubleshoot issues related to Amazon Cognito. • AWS Backup – To debug issues related to AWS Backup. • Directory Service – To debug issues related to Directory Service. • Amazon EC2 – To debug issues related to Amazon EC2.	Sep 30, 2025

Change	Description	Date
	• AWS FIS – To troubleshoot issues related to AWS FIS. • Amazon FSx – To troubleshoot issues related to Amazon FSx. • AWS Global Accelerator – To debug issues related to AWS Global Accelerator. • AWS Identity Store – To debug issues related to AWS Identity Store. • AWS Invoicing – To debug issues related to AWS Invoicing. • AWS Lake Formation – To troubleshoot issues related to AWS Lake Formation. • AWS Network Firewall – To debug issues related to AWS Network Firewall. • Oracle Database@AWS – To troubleshoot issues related to Oracle Database@AWS. • Amazon S3 – To debug issues related to Amazon S3. • Amazon SES – To troubleshoot issues related to Amazon SES. • AWS IAM Identity Center – To troubleshoot issues	

Change	Description	Date
	related to AWS IAM Identity Center.	
AWSSupportAccess – Update to an existing policy	Added permissions for the support-console API to the AWSSupportAccess managed policy.	July 18, 2025

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 25 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administrative, and technical support: • CloudFormation – To debug issues related to CloudFormation. • AWS Config – To troubleshoot issues related to AWS Config. • Amazon OpenSearch Service – To debug issues related to Amazon OpenSearch Service. • AWS Glue – To debug issues related to AWS Glue. • AWS IAM – To troubleshoot issues related to AWS IAM. • Amazon Pinpoint – To troubleshoot issues related to Amazon Pinpoint. • AWS Outposts – To debug issues related to AWS Outposts. • AWS STS – To debug issues related to AWS STS. For more information, see Permission changes for	Jul 15, 2025

Change	Description	Date
	AWSSupportServiceRolePolicy .	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 257 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administrative, and technical support: • AWS App Runner – To debug issues related to AWS App Runner. • AWS AppSync – To troubleshoot issues related to AWS AppSync. • AWS Batch – To debug issues related to AWS Batch. • Amazon Bedrock – To troubleshoot issues related to Amazon Bedrock. • Amazon CloudFront – To debug issues related to Amazon CloudFront. • AWS CodePipeline – To troubleshoot issues related to AWS CodePipeline. • AWS Config – To troubleshoot issues related to AWS Config. • Amazon Connect – To debug issues related to Amazon Connect.	Jun 17, 2025

Change	Description	Date
	• AWS DataSync – To debug issues related to AWS DataSync. • Direct Connect – To troubleshoot issues related to the Direct Connect. • Amazon EC2 – To troubleshoot issues related to the Amazon EC2. • AWS Fault Injection Service – To debug issues related to the AWS Fault Injection Service. • AWS Firewall Manager – To troubleshoot issues related to the AWS Firewall Manager. • AWS Glue – To debug issues related to the AWS Glue. • Amazon GuardDuty – To debug issues related to the Amazon GuardDuty. • EC2 Image Builder – To troubleshoot issues related to the EC2 Image Builder. • AWS IoT – To troubleshoot issues related to the AWS IoT. • AWS IoT FleetWise – To debug issues related to the AWS IoT FleetWise. • Amazon CloudWatch Logs – To debug issues related	

Change	Description	Date
	to the Amazon CloudWatch Logs. • AWS Elemental MediaLive – To debug issues related to the AWS Elemental MediaLive. • Network Flow Monitor – To troubleshoot issues related to the Network Flow Monitor. • AWS Network Manager – To troubleshoot issues related to AWS Network Manager. • Amazon CloudWatch Observability Admin Service – To troublesh oot issues related to the Amazon CloudWatc h Observability Admin Service. • AWS Parallel Computing Service – To debug issues related to the AWS Parallel Computing Service. • Amazon Redshift Serverles s – To troubleshoot issues related to the Amazon Redshift Serverless. • Amazon Redshift – To troubleshoot issues related to the Amazon Redshift.	

Change	Description	Date
	• AWS Resilience Hub – To debug issues related to the AWS Resilience Hub. • AWS Identity and Access Management Roles Anywhere– To debug issues related to the AWS Identity and Access Management Roles Anywhere. • Amazon S3 on Outposts – To troubleshoot issues related to the Amazon S3 on Outposts. • Amazon S3 – To troublesh oot issues related to the Amazon S3. • Amazon S3 Tables– To troubleshoot issues related to the Amazon S3 Tables. • Amazon SageMaker AI – To debug issues related to the Amazon SageMaker AI. • AWS Security Hub CSPM – To debug issues related to the AWS Security Hub CSPM. • Amazon SQS – To debug issues related to the Amazon SQS. • AWS Systems Manager Incident Manager – To troubleshoot issues related	

Change	Description	Date
	to the AWS Systems Manager Incident Manager. • AWS Systems Manager Quick Setup – To debug issues related to the AWS Systems Manager Quick Setup. • AWS Systems Manager – To debug issues related to the AWS Systems Manager. • Amazon WorkSpaces Thin Client – To troublesh oot issues related to the Amazon WorkSpaces Thin Client. • Amazon Timestream – To debug issues related to the Amazon Timestream. • AWS Telco Network Builder – To troubleshoot issues related to the AWS Telco Network Builder. • AWS Transfer Family – To debug issues related to the AWS Transfer Family. • Amazon VPC Lattice – To troubleshoot issues related to the Amazon VPC Lattice.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 88 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • Amazon Bedrock – To troubleshoot issues related to Amazon Bedrock. • Amazon Connect – To debug issues related to Amazon Connect. • Amazon DataZone – To debug issues related to Amazon DataZone. • Amazon EC2 – To troublesh oot issues related to the Amazon EC2. • Amazon EKS – To debug issues related to the Amazon EKS. • AWS Glue – To troubleshoot issues related to AWS Glue. • Amazon Managed Service for Apache Flink – To troubleshoot issues related to the Amazon Managed Service for Apache Flink. • AWS Lambda – To debug issues related to the AWS Lambda.	Nov 25, 2024

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 79 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • Amazon OpenSearch Serverless – To troublesh oot issues related to Amazon OpenSearch Serverless. • AWS AppConfig – To debug issues related to AWS AppConfig. • Application Signals– To debug issues related to Application Signals. • Amazon Athena – To troubleshoot issues related to the Amazon Athena. • Amazon CloudWatch – To debug issues related to the Amazon CloudWatch. • Amazon DynamoDB – To troubleshoot issues related to Amazon DynamoDB. • Amazon EC2 – To troublesh oot issues related to the Amazon EC2. • AWS IoT – To debug issues related to the AWS IoT.	Oct 8, 2024

Change	Description	Date
	• AWS Lambda – To troubleshoot issues related to the AWS Lambda. • AWS Launch Wizard – To troubleshoot issues related to the AWS Launch Wizard. • AWS Security Hub CSPM – To debug issues related to AWS Security Hub CSPM. • Amazon WorkSpaces – To debug issues related to the Amazon WorkSpaces.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 79 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administrative, and technical support: • AWS account – To troubleshoot issues related to the AWS account. • AWS Auto Scaling – To debug issues related to AWS Auto Scaling. • Amazon Bedrock – To debug issues related to Amazon Bedrock. • AWS CodeConnections – To troubleshoot issues related to the AWS CodeConnections. • AWS Deadline Cloud – To debug issues related to the AWS Deadline Cloud. • Amazon Elastic Kubernetes Service – To troubleshoot issues related to Amazon Elastic Kubernetes Service. • Elastic Load Balancing – To troubleshoot issues related to the Elastic Load Balancing.	Aug 5, 2024

Change	Description	Date
	• AWS Free Tier – To debug issues related to the AWS Free Tier. • Amazon Inspector – To troubleshoot issues related to the Amazon Inspector. • Amazon OpenSearch Ingestion – To troublesh oot issues related to the Amazon OpenSearch Ingestion. • Amazon WorkSpaces – To debug issues related to Amazon WorkSpaces. • AWS X-Ray – To debug issues related to the AWS X-Ray.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 17 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • Amazon CloudWatch Network Monitor – To troubleshoot issues related to the Network Monitor service. • Amazon CloudWatch Logs – To debug issues related to Amazon CloudWatch Logs. • Amazon Managed Streaming for Apache Kafka – To debug issues related to Amazon Managed Streaming for Apache Kafka. • Amazon Managed Service for Prometheus – To troubleshoot issues related to the Amazon Managed Service for Prometheus.	Mar 22, 2024

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 63 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • AWS Clean Rooms – To troubleshoot issues related to the AWS Clean Rooms. • CodeConnections – To troubleshoot issues related to CodeConnections. • Amazon EKS – To debug issues related to Amazon EKS. • Image Builder – To debug issues related to the Image Builder. • Amazon Inspector2 – To troubleshoot issues related to Amazon Inspector2. • Amazon Inspector Scan – To debug issues related to the Amazon Inspector Scan. • Amazon CloudWatch Logs – To troubleshoot issues related to Amazon CloudWatch Logs. • AWS Outposts – To troubleshoot issues related to the AWS Outposts.	Jan 17, 2024

Change	Description	Date
	• Amazon RDS – To debug issues related to Amazon RDS. • AWS IAM Identity Center – To troubleshoot issues related to AWS IAM Identity Center. • Amazon S3 Express – To debug issues related to Amazon S3 Express. • AWS Trusted Advisor – To troubleshoot issues related to AWS Trusted Advisor.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 126 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administrative, and technical support: • AWS Direct Connect – To troubleshoot issues related to the AWS Direct Connect service. • Amazon SageMaker AI – To troubleshoot issues related to Amazon SageMaker AI service. • Amazon AppStream – To debug issues related to Amazon AppStream. • AWS Resource Explorer – To debug issues related to the AWS Resource Explorer. • Amazon Redshift serverless – To troubleshoot issues related to Amazon Redshift serverless. • Amazon ElastiCache – To debug issues related to the Amazon ElastiCache. • Amazon Comprehend – To troubleshoot issues related to Amazon Comprehend.	Dec 6, 2023

Change	Description	Date
	• Amazon EC2 – To troublesh oot issues related to the Amazon EC2. • Amazon Elastic Kubernete s Service – To debug issues related to Amazon Elastic Kubernetes Service. • AWS Elastic Disaster Recovery – To troublesh oot issues related to AWS Elastic Disaster Recovery. • AWS AppSync – To debug issues related to AWS AppSync. • Amazon CloudWatch Logs – To troubleshoot issues related to Amazon CloudWatch Logs. • AWS Health – To debug issues related to the AWS Health Service. • Amazon Connect – To debug issues related to the Amazon Connect. • AWS Snowball Edge – To troubleshoot issues related to AWS Snowball Edge. • AWS HealthImaging – To troubleshoot issues related to AWS HealthImaging.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 163 new permissio ns to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • Amazon CloudFront – To troubleshoot issues related to the CloudFront service. • Amazon EC2 – To troublesh oot issues related to Amazon EC2 service. • Amazon AppStream – To debug issues related to Amazon AppStream. • AWS WAF – To debug issues related to the AWS Web Application Firewall. • Amazon Connect – To troubleshoot issues related to Amazon Connect. • AWS IoT – To debug issues related to the AWS IoT. • Amazon Route 53 – To troubleshoot issues related to Amazon Route 53. • AWS Verified Access – To troubleshoot issues related to the AWS Verified Access service. • Amazon Simple Email Service – To debug issues	Oct 27, 2023

Change	Description	Date
	related to Amazon Simple Email Service. • AWS Elastic Beanstalk – To troubleshoot issues related to AWS Elastic Beanstalk. • Amazon DynamoDB – To debug issues related to Amazon DynamoDB. • AWS EC2 Image Builder – To troubleshoot issues related to AWS EC2 Image Builder. • AWS Outposts – To debug issues related to the AWS Outposts Service. • AWS Glue – To debug issues related to the AWS Glue. • AWS Directory Service – To troubleshoot issues related to AWS Directory Service. • AWS Elastic Disaster Recovery – To troublesh oot issues related to AWS Elastic Disaster Recovery. • AWS Step Functions – To debug issues related to AWS Step Functions. • Amazon EMR – To troublesh oot issues related to Amazon EMR. • Amazon Relational Database Service – To troubleshoot issues related	

Change	Description	Date
	to Amazon Relational Database Service. • Amazon EC2 Systems Manager – To debug issues related to Amazon EC2 Systems Manager.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 176 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administrative, and technical support: • AWS Glue – To troubleshoot issues related to the AWS Glue service • Amazon EMR – To troubleshoot issues related to Amazon EMR service. • Amazon Security Lake – To debug issues related to Amazon Security Lake. • AWS Systems Manager – To debug issues related to the Systems Manager service. • Amazon Verified Permissions – To troubleshoot issues related to Amazon Verified Permissions. • AWS IAM Access Analyzer – To debug issues related to the IAM Access Analyzer service. • AWS Backup – To troubleshoot issues related to AWS Backup. • AWS Database Migration Service – To troubleshoot	Aug 28, 2023

Change	Description	Date
	issues related to the DMS service. • Amazon DynamoDB – To debug issues related to Dynamo DB. • Amazon Elastic Container Registry (Amazon ECR) – To troubleshoot issues related to Amazon Elastic Container Registry (Amazon ECR). • Amazon Elastic Container Service – To debug issues related to Amazon Elastic Container Service. • Amazon Elastic Kubernetes Service – To troubleshoot issues related to Amazon Elastic Kubernetes Service. • Amazon EMR Serverless – To debug issues related to the Amazon EMR Serverless Service. • AWS Identity and Access Management – To troubleshoot issues related to AWS Identity and Access Management. • AWS Network Firewall – To troubleshoot issues related to AWS Network Firewall.	

Change	Description	Date
	• AWS HealthOmics – To debug issues related to AWS HealthOmics. • Amazon Quick Suite – To debug issues related to Amazon Quick Suite. • Amazon Relational Database Service – To troubleshoot issues related to Amazon Relational Database Service. • Amazon Redshift – To troubleshoot issues related to Amazon Redshift. • Amazon Redshift Serverless – To debug issues related to Amazon Redshift Serverless. • Amazon SageMaker AI – To debug issues related to Amazon SageMaker AI.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 141 new permissio ns to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • Lambda – To troubleshoot issues related to Lambda service. • Amazon Lex – To troublesh oot issues related to Amazon Lex service. • AWS Transfer – To debug issues related to Transfer service. • AWS Amplify – To debug issues related to Amplify service. • Amazon EventBridge Pipes – To troubleshoot permissio ns and billing issues related to Pipes. • Amazon EventBridge – To debug issues related to Amazon EventBridge • Amazon CloudWatch Logs – To troubleshoot issues related to Amazon CloudWatch Logs. • AWS Systems Manager – To troubleshoot issues related to Systems Manager.	June 26, 2023

Change	Description	Date
	• Amazon CloudWatch – To debug issues related to CloudWatch. • Amazon ElastiCache – To troubleshoot issues related to Amazon ElastiCache. • Amazon Athena – To debug issues related to Athena. • AWS Elastic Disaster Recovery – To troublesh oot issues related to Elastic Disaster Recovery. • Amazon CloudWatch – To troubleshoot configurations of Amazon CloudWatch. • Amazon EC2 – To debug issues related to the EC2 service. • AWS Certificate Manager – To troubleshoot issues related to Certificate Manager. • Amazon EventBridge Scheduler – To troublesh oot issues related to EventBridge Scheduler. • Amazon OpenSearch Service – To troubleshoot issues related to OpenSearc h. • Amazon EventBridge Schemas – To debug issues	

Change	Description	Date
	related to EventBridge Schemas. • AWS User Notifications – To troubleshoot issues related to User Notifications. • Amazon CloudWatch Application Insights – To troubleshoot issues related to CloudWatch Application Insights. • Amazon DynamoDB – To troubleshoot issues related to DynamoDB. • Amazon DocumentD B Elastic Clusters – To troubleshoot issues related to DocumentDB Elastic Clusters.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 53 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • Auto Scaling – To troublesh oot issues related to Auto Scaling service. • Amazon CloudWatch – To troubleshoot issues related to Amazon CloudWatch. • AWS Compute Optimizer – To troubleshoot issues related to Compute Optimizer. • Amazon CloudWatch Evidently – To troubleshoot issues related to Evidently. • EC2 Image Builder – To troubleshoot issues related to Image Builder service. • AWS IoT TwinMaker – To troubleshoot issues related to AWS IoT TwinMaker. • Amazon CloudWatch Logs – To troubleshoot issues related to Amazon CloudWatch Logs. • Amazon Pinpoint – To troubleshoot issues related to Amazon Pinpoint.	May 02, 2023

Change	Description	Date
	• AWS OAM Link – To debug issues related to OAM resources. • AWS Outposts – To troubleshoot issues related to AWS Outposts. • Amazon RDS – To debug issues related to Amazon RDS. • AWS Resource Explorer – To troubleshoot issues related to Resource Explorer. • Amazon CloudWatch RUM – To troubleshoot configurations of RUM service resources. • Amazon SNS – To troublesh oot issues related to Amazon SNS. • Amazon CloudWatch Synthetics – To troublesh oot issues related to CloudWatch Synthetics.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 52 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • AWS Backup gateway – To troubleshoot issues related to Backup gateway. • Amazon S3 – To debug issues related to Amazon S3. • AWS Application Migration Service – To troubleshoot issues related to Applicati on Migration Service. • AWS Clean Rooms – To debug issues related to AWS Clean Rooms; • AWS Systems Manager for SAP – To troublesh oot issues related to AWS Systems Manager for SAP. • Amazon VPC Lattice – To debug issues related to Amazon VPC Lattice.	March 16, 2023

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 220 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administrative, and technical support: • Amazon Athena – To enable AWS Support to develop tools that can be used to help customers with their queries related to Athena. • Amazon Chime – To troubleshoot issues related to Amazon Chime. • Amazon CloudWatch Internet Monitor – To debug issues related to Internet Monitor. • Amazon Comprehend – To troubleshoot issues related to Amazon Comprehend. • Amazon Elastic Compute Cloud – To debug issues related to Transit Gateway Connect and multicast features. • Amazon EventBridge Pipes – To troubleshoot issues related to EventBridge Pipes. • Amazon Interactive Video Service – To enable AWS Support to query Amazon	January 10, 2023

Change	Description	Date
	IVS resources to troublesh oot customer issues. • Amazon FSx – To enable AWS Support to develop tools to support importing and exporting for an Amazon FSx data repositor y. • Amazon GameLift Servers – To troubleshoot issues related to Amazon GameLift Servers. • AWS Glue– To troubleshoot issues related to AWS Glue Data Quality. • Amazon Kinesis Video Streams– To troubleshoot issues related to Kinesis Video Streams. • Amazon Managed Service for Prometheus – To troubleshoot issues related to Amazon Managed Service for Prometheus. • Amazon Managed Streaming for Apache Kafka – To troubleshoot issues related to Amazon MSK Connect. • AWS Network Manager – To troubleshoot issues related to Network Manager.	

Change	Description	Date
	• Amazon Nimble Studio – To debug issues related to Nimble Studio. • Amazon Personalize – To debug issues related to Amazon Personalize. • Amazon Pinpoint – To troubleshoot issues related to Amazon Pinpoint. • AWS HealthOmics – To troubleshoot issues related to HealthOmics. • Amazon Transcribe – To debug issues related to Amazon Transcribe.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 47 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • AWS Application Migration Service – To troubleshoot replication and launch issues. • AWS CloudFormation hooks – To enable AWS Support to develop automation tools that can help resolve issues. • Amazon Elastic Kubernete s Service – To troubleshoot issues related to Amazon EKS. • AWS IoT FleetWise – To troubleshoot issues related to AWS IoT FleetWise. • AWS Mainframe Moderniza tion – To debug issues related to AWS Mainframe Modernization. • AWS Outposts – To help AWS Support get a list of dedicated hosts and assets. • AWS Private 5G – To troubleshoot issues related to Private 5G.	October 4, 2022

Change	Description	Date
	• AWS Tiros – To debug issues related to Tiros.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 46 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • Amazon Managed Streaming for Apache Kafka – To troubleshoot issues related to Amazon MSK. • AWS DataSync – To troubleshoot issues related to DataSync. • AWS Elastic Disaster Recovery – To troublesh oot replication and launch issues. • Amazon GameSparks – To troubleshoot issues related to GameSparks. • AWS IoT TwinMaker – To debug issues related to AWS IoT TwinMaker. • AWS Lambda – To view the configuration of a function URL to troubleshooting issues. • Amazon Lookout for Equipment – To troublesh oot issues related to Lookout for Equipment.	August 17, 2022

Change	Description	Date
	• Amazon Route 53 and Amazon Route 53 Resolver – To get resolver configurations so that AWS Support can check the DNS resolution behavior of a VPC.	
AWSSupportServiceRolePolicy – Update to an existing policy	Added new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administrative, and technical support: • Amazon CloudWatch Logs – To help troubleshoot CloudWatch Logs related issues. • Amazon Interactive Video Service – To help Support check existing Amazon IVS resources for support cases regarding fraud or compromised accounts. • Amazon Inspector – To troubleshoot Amazon Inspector related issues. Removed permissions for services, such as Amazon WorkLink. Amazon WorkLink was deprecated on April 19, 2022.	June 23, 2022

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 25 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • AWS Amplify UI Builder – To troubleshoot issues related to component and theme generation. • Amazon AppStream – To troubleshoot issues by retrieving resources for features that launched recently. • AWS Backup – To troublesh oot issues related to backup jobs. • AWS CloudFormation – To perform diagnostics on issues related to IAM, extension, and versioning. • Amazon Kinesis – To troubleshoot issues related to Kinesis. • AWS Transfer Family – To troubleshoot issues related to Transfer Family.	April 27, 2022

Change	Description	Date

AWSSupportServiceRolePolicy – Update to an existing policy	Added 54 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administrative, and technical support:
	• Amazon Elastic Compute Cloud
	  • To troubleshoot issues related to customer and AWS-managed prefixed lists.
	  • To troubleshoot issues related to Amazon VPC IP Address Manager (IPAM).
	• AWS Network Manager – To troubleshoot issues related to Network Manager.
	• Savings Plans – To get metadata about outstanding Savings Plans commitments.
	• AWS Serverless Application Repository – To improve and support response actions as part of researching and resolving support cases.
	• Amazon WorkSpaces Web – To debug and troubleshoot issues with WorkSpaces Web services.
		March 14, 2022

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 74 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administr ative, and technical support: • AWS Application Migration Service – To support agentless replication in the Application Migration Service. • AWS CloudFormation – To perform diagnostics on IAM, extension, and versioning related issues. • Amazon CloudWatch Logs – To validate resource policies. • Amazon EC2 Recycle Bin – To get metadata about Recycle Bin retention rules. • AWS Elastic Disaster Recovery – To troublesh oot replication and launch problems in customer accounts. • Amazon FSx – To view the description of Amazon FSx snapshots. • Amazon Lightsail – To view metadata and configura tions details for Lightsail buckets.	February 17, 2022

Change	Description	Date
	• Amazon Macie – To view Macie configurations, such as classification jobs, custom data identifiers, regular expressions and findings. • Amazon S3 – To gather metadata and configura tions for Amazon S3 buckets. • AWS Storage Gateway – To view metadata about customers' automatic tape creation policies. • Elastic Load Balancing – To view the description of resource limits when using the Service Quotas console. For more information, see Permission changes for AWSSupportServiceRolePolicy .	
Change log published	Change log for the AWS Support managed policies.	February 17, 2022

Permission changes for AWSSupportServiceRolePolicy

Most permissions added to `AWSSupportServiceRolePolicy` allow AWS Support to call an API operation with the same name. However, some API operations require permissions that have a different name.

The following table only lists the API operations that require permissions with a different name. This table describes these differences beginning on February 17, 2022.

Date	API operation name	Required policy permission
Added permissions on February 17, 2022	s3.GetBucketAnalyticsConfiguration s3.ListBucketAnalyticsConfiguration	s3:GetAnalyticsConfiguration
	s3.GetBucketNotificationConfiguration	s3:GetBucketNotification
	s3.GetBucketEncryption	s3:GetEncryptionConfiguration
	s3.GetBucketIntelligentTieringConfiguration s3.ListBucketIntelligentTieringConfiguration	s3:GetIntelligentTieringConfiguration
	s3.GetBucketInventoryConfiguration s3.ListBucketInventoryConfiguration	s3:GetInventoryConfiguration
	s3.GetBucketLifecycleConfiguration	s3:GetLifecycleConfiguration
	s3.GetBucketMetricsConfiguration s3.ListBucketMetricsConfiguration	s3:GetMetricsConfiguration

Date	API operation name	Required policy permission
	s3.GetBucketReplication	s3:GetReplicationConfiguration
	s3.HeadBucket s3.ListObjects	s3:ListBucket
	s3.ListBuckets	s3:ListAllMyBuckets
	s3.ListMultipartUploads	s3:ListBucketMultipartUploads
	s3.ListObjectVersions	s3:ListBucketVersions
	s3.ListParts	s3:ListMultipartUploadParts
Added permissions on July 15, 2025	cloudcontrolapi:GetResource	cloudformation:GetResource
	cloudcontrolapi:ListResources	cloudformation:ListResources

AWS managed policies for AWS Support App in Slack

> **Note**
>
> To access and view support cases in the AWS Support Center Console, see Manage access to AWS Support Center.

AWS Support App has the following managed policies.

Contents

- AWS managed policy: AWSSupportAppReadOnlyAccess
- AWS Support App updates to AWS managed policies

AWS managed policy: AWSSupportAppFullAccess

You can use the AWSSupportAppFullAccess managed policy to grant the IAM role the permissions to your Slack channel configurations. You can also attach the AWSSupportAppFullAccess policy to your IAM entities.

For more information, see AWS Support App in Slack.

To view the permissions for this policy, see AWSSupportAppFullAccess in the *AWS Managed Policy Reference*.

Permissions details

This policy includes the following permissions:

- `servicequotas` – Describes your existing service quotas and requests, and creates service quota increases for your account.
- `support` – Creates, updates, and resolves your support cases. Updates and describes information about your cases, such as file attachments, correspondences, and severity levels. Initiates live chat sessions with a support agent.
- `iam` – Creates a service-linked role for Service Quotas.

For more information, see Managing access to the AWS Support App.

AWS managed policy: AWSSupportAppReadOnlyAccess

The AWSSupportAppReadOnlyAccess policy grants permissions that allow the entity to perform read-only AWS Support App actions. For more information, see AWS Support App in Slack.

To view the permissions for this policy, see AWSSupportAppReadOnlyAccess in the *AWS Managed Policy Reference*.

Permissions details

This policy includes the following permissions:

- `support` – Describes support case details and communications added to the support cases.

AWS Support App updates to AWS managed policies

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

The following table describes important updates to the AWS Support App managed policies since August 17, 2022.

AWS Support App

Change	Description	Date
AWSSupportAppFullAccess and AWSSupportAppReadOnlyAccess New AWS managed policies for the AWS Support App	You can use these policies for the IAM role that you configure for your Slack channel configuration. For more information, see Managing access to the AWS Support App.	August 19, 2022
Change log published	Change log for the AWS Support App managed policies.	August 19, 2022

AWS managed policies for AWS Trusted Advisor

Trusted Advisor has the following AWS managed policies.

Contents

- Trusted Advisor updates to AWS managed policies

AWS managed policy: AWSTrustedAdvisorPriorityFullAccess

The AWSTrustedAdvisorPriorityFullAccess policy grants full access to Trusted Advisor Priority. This policy also allows the user to add Trusted Advisor as a trusted service with AWS Organizations and to specify the delegated administrator accounts for Trusted Advisor Priority.

Permissions details

In the first statement, the policy includes the following permissions for `trustedadvisor`:

- Describes your account and organization.
- Describes identified risks from Trusted Advisor Priority. The permissions allow you to download and update the risk status.
- Describes your configurations for Trusted Advisor Priority email notifications. The permissions allow you to configure the email notifications and disable them for your delegated administrators.
- Sets up Trusted Advisor so that your account can enable AWS Organizations.

In the second statement, the policy includes the following permissions for `organizations`:

- Describes your Trusted Advisor account and organization.
- Lists the AWS services that you enabled to use Organizations.

In the third statement, the policy includes the following permissions for `organizations`:

- Lists the delegated administrators for Trusted Advisor Priority.
- Enables and disables trusted access with Organizations.

In the fourth statement, the policy includes the following permissions for `iam`:

- Creates the `AWSServiceRoleForTrustedAdvisorReporting` service-linked role.

In the fifth statement, the policy includes the following permissions for `organizations`:

- Allows you to register and deregister delegated administrators for Trusted Advisor Priority.

JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Sid": "AWSTrustedAdvisorPriorityFullAccess",
   "Effect": "Allow",
   "Action": [
    "trustedadvisor:DescribeAccount*",
    "trustedadvisor:DescribeOrganization",
    "trustedadvisor:DescribeRisk*",
    "trustedadvisor:DownloadRisk",
    "trustedadvisor:UpdateRiskStatus",
    "trustedadvisor:DescribeNotificationConfigurations",
    "trustedadvisor:UpdateNotificationConfigurations",
    "trustedadvisor:DeleteNotificationConfigurationForDelegatedAdmin",
    "trustedadvisor:SetOrganizationAccess"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowAccessForOrganization",
   "Effect": "Allow",
   "Action": [
    "organizations:DescribeAccount",
    "organizations:DescribeOrganization",
    "organizations:ListAWSServiceAccessForOrganization"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowListDelegatedAdministrators",
   "Effect": "Allow",
   "Action": [
    "organizations:ListDelegatedAdministrators",
    "organizations:EnableAWSServiceAccess",
    "organizations:DisableAWSServiceAccess"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "organizations:ServicePrincipal": [
```

```
      "reporting.trustedadvisor.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "AllowCreateServiceLinkedRole",
   "Effect": "Allow",
   "Action": "iam:CreateServiceLinkedRole",
   "Resource": "arn:aws:iam::*:role/aws-service-role/
reporting.trustedadvisor.amazonaws.com/AWSServiceRoleForTrustedAdvisorReporting",
   "Condition": {
    "StringLike": {
     "iam:AWSServiceName": "reporting.trustedadvisor.amazonaws.com"
    }
   }
  },
  {
   "Sid": "AllowRegisterDelegatedAdministrators",
   "Effect": "Allow",
   "Action": [
    "organizations:RegisterDelegatedAdministrator",
    "organizations:DeregisterDelegatedAdministrator"
   ],
   "Resource": "arn:aws:organizations::*:*",
   "Condition": {
    "StringEquals": {
     "organizations:ServicePrincipal": [
      "reporting.trustedadvisor.amazonaws.com"
     ]
    }
   }
  }
 ]
}
```

AWS managed policy: AWSTrustedAdvisorPriorityReadOnlyAccess

The AWSTrustedAdvisorPriorityReadOnlyAccess policy grants read-only permissions to Trusted Advisor Priority, including permission to view the delegated administrator accounts.

Permissions details

In the first statement, the policy includes the following permissions for `trustedadvisor`:

- Describes your Trusted Advisor account and organization.
- Describes the identified risks from Trusted Advisor Priority and allows you to download them.
- Describes the configurations for Trusted Advisor Priority email notifications.

In the second and third statement, the policy includes the following permissions for `organizations`:

- Describes your organization with Organizations.
- Lists the AWS services that you enabled to use Organizations.
- Lists the delegated administrators for Trusted Advisor Priority

JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Sid": "AWSTrustedAdvisorPriorityReadOnlyAccess",
   "Effect": "Allow",
   "Action": [
    "trustedadvisor:DescribeAccount*",
    "trustedadvisor:DescribeOrganization",
    "trustedadvisor:DescribeRisk*",
    "trustedadvisor:DownloadRisk",
    "trustedadvisor:DescribeNotificationConfigurations"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowAccessForOrganization",
   "Effect": "Allow",
   "Action": [
    "organizations:DescribeOrganization",
    "organizations:ListAWSServiceAccessForOrganization"
   ],
   "Resource": "*"
  },
```

```
  {
   "Sid": "AllowListDelegatedAdministrators",
   "Effect": "Allow",
   "Action": [
    "organizations:ListDelegatedAdministrators"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "organizations:ServicePrincipal": [
      "reporting.trustedadvisor.amazonaws.com"
     ]
    }
   }
  }
 ]
}
```

AWS managed policy: AWSTrustedAdvisorServiceRolePolicy

This policy is attached to the `AWSServiceRoleForTrustedAdvisor` service-linked role. It allows the service-linked role to perform actions for you. You can't attach the AWSTrustedAdvisorServiceRolePolicy to your AWS Identity and Access Management (IAM) entities. For more information, see Using service-linked roles for Trusted Advisor.

This policy grants administrative permissions that allow the service-linked role to access AWS services. These permissions allow the checks for Trusted Advisor to evaluate your account.

Permissions details

This policy includes the following permissions.

- `accessanalyzer` – Describes AWS Identity and Access Management Access Analyzer resources
- `Auto Scaling` – Describes Amazon EC2 Auto Scaling account quotas and resources
- `cloudformation` – Describes AWS CloudFormation (CloudFormation) account quotas and stacks
- `cloudfront` – Describes Amazon CloudFront distributions

- `cloudtrail` – Describes AWS CloudTrail (CloudTrail) trails
- `dynamodb` – Describes Amazon DynamoDB account quotas and resources
- `dynamodbaccelerator` – Describes DynamoDB Accelerator resources
- `ec2` – Describes Amazon Elastic Compute Cloud (Amazon EC2) account quotas and resources
- `elasticloadbalancing` – Describes ELB (ELB) account quotas and resources
- `iam` – Gets IAM resources, such as credentials, password policy, and certificates
- `networkfirewall` – Describes AWS Network Firewall resources
- `kinesis` – Describes Amazon Kinesis (Kinesis) account quotas
- `rds` – Describes Amazon Relational Database Service (Amazon RDS) resources
- `redshift` – Describes Amazon Redshift resources
- `route53` – Describes Amazon Route 53 account quotas and resources
- `s3` – Describes Amazon Simple Storage Service (Amazon S3) resources
- `ses` – Gets Amazon Simple Email Service (Amazon SES) send quotas
- `sqs` – Lists Amazon Simple Queue Service (Amazon SQS) queues
- `cloudwatch` – Gets Amazon CloudWatch Events (CloudWatch Events) metric statistics
- `ce` – Gets Cost Explorer Service (Cost Explorer) recommendations
- `route53resolver` – Gets Amazon Route 53 Resolver Resolver Endpoints and resources
- `kafka` – Gets Amazon Managed Streaming for Apache Kafka resources
- `ecs` – Gets Amazon ECS resources
- `outposts` – Gets AWS Outposts resources

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "TrustedAdvisorServiceRolePermissions",
            "Effect": "Allow",
            "Action": [
                "access-analyzer:ListAnalyzers",
                "autoscaling:DescribeAccountLimits",
                "autoscaling:DescribeAutoScalingGroups",
```

```
                "autoscaling:DescribeLaunchConfigurations",
                "ce:GetReservationPurchaseRecommendation",
                "ce:GetSavingsPlansPurchaseRecommendation",
                "cloudformation:DescribeAccountLimits",
                "cloudformation:DescribeStacks",
                "cloudformation:ListStacks",
                "cloudfront:ListDistributions",
                "cloudtrail:DescribeTrails",
                "cloudtrail:GetTrailStatus",
                "cloudtrail:GetTrail",
                "cloudtrail:ListTrails",
                "cloudtrail:GetEventSelectors",
                "cloudwatch:GetMetricStatistics",
                "cloudwatch:ListMetrics",
                "dax:DescribeClusters",
                "dynamodb:DescribeLimits",
                "dynamodb:DescribeTable",
                "dynamodb:ListTables",
                "ec2:DescribeAddresses",
                "ec2:DescribeReservedInstances",
                "ec2:DescribeInstances",
                "ec2:DescribeVpcs",
                "ec2:DescribeInternetGateways",
                "ec2:DescribeImages",
                "ec2:DescribeNatGateways",
                "ec2:DescribeVolumes",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeRegions",
                "ec2:DescribeReservedInstancesOfferings",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSnapshots",
                "ec2:DescribeVpcEndpoints",
                "ec2:DescribeVpnConnections",
                "ec2:DescribeVpnGateways",
                "ec2:DescribeLaunchTemplateVersions",
                "ec2:GetManagedPrefixListEntries",
                "ecs:DescribeTaskDefinition",
                "ecs:ListTaskDefinitions",
                "elasticloadbalancing:DescribeAccountLimits",
                "elasticloadbalancing:DescribeInstanceHealth",
                "elasticloadbalancing:DescribeLoadBalancerAttributes",
                "elasticloadbalancing:DescribeLoadBalancerPolicies",
                "elasticloadbalancing:DescribeLoadBalancerPolicyTypes",
```

```
                "elasticloadbalancing:DescribeLoadBalancers",
                "elasticloadbalancing:DescribeListeners",
                "elasticloadbalancing:DescribeRules",
                "elasticloadbalancing:DescribeTargetGroups",
                "elasticloadbalancing:DescribeTargetHealth",
                "iam:GenerateCredentialReport",
                "iam:GetAccountPasswordPolicy",
                "iam:GetAccountSummary",
                "iam:GetCredentialReport",
                "iam:GetServerCertificate",
                "iam:ListServerCertificates",
                "iam:ListSAMLProviders",
                "kinesis:DescribeLimits",
                "kafka:DescribeClusterV2",
                "kafka:ListClustersV2",
                "kafka:ListNodes",
                "network-firewall:ListFirewalls",
                "network-firewall:DescribeFirewall",
                "outposts:GetOutpost",
                "outposts:ListAssets",
                "outposts:ListOutposts",
                "rds:DescribeAccountAttributes",
                "rds:DescribeDBClusters",
                "rds:DescribeDBEngineVersions",
                "rds:DescribeDBInstances",
                "rds:DescribeDBParameterGroups",
                "rds:DescribeDBParameters",
                "rds:DescribeDBSecurityGroups",
                "rds:DescribeDBSnapshots",
                "rds:DescribeDBSubnetGroups",
                "rds:DescribeEngineDefaultParameters",
                "rds:DescribeEvents",
                "rds:DescribeOptionGroupOptions",
                "rds:DescribeOptionGroups",
                "rds:DescribeOrderableDBInstanceOptions",
                "rds:DescribeReservedDBInstances",
                "rds:DescribeReservedDBInstancesOfferings",
                "rds:ListTagsForResource",
                "redshift:DescribeClusters",
                "redshift:DescribeReservedNodeOfferings",
                "redshift:DescribeReservedNodes",
                "route53:GetAccountLimit",
                "route53:GetHealthCheck",
                "route53:GetHostedZone",
```

```
                "route53:ListHealthChecks",
                "route53:ListHostedZones",
                "route53:ListHostedZonesByName",
                "route53:ListResourceRecordSets",
                "route53resolver:ListResolverEndpoints",
                "route53resolver:ListResolverEndpointIpAddresses",
                "s3:GetAccountPublicAccessBlock",
                "s3:GetBucketAcl",
                "s3:GetBucketPolicy",
                "s3:GetBucketPolicyStatus",
                "s3:GetBucketLocation",
                "s3:GetBucketLogging",
                "s3:GetBucketVersioning",
                "s3:GetBucketPublicAccessBlock",
                "s3:GetLifecycleConfiguration",
                "s3:ListBucket",
                "s3:ListAllMyBuckets",
                "ses:GetSendQuota",
                "sqs:GetQueueAttributes",
                "sqs:ListQueues"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS managed policy: AWSTrustedAdvisorReportingServiceRolePolicy

This policy is attached to the `AWSServiceRoleForTrustedAdvisorReporting` service-linked role that allows Trusted Advisor to perform actions for the organizational view feature. You can't attach the AWSTrustedAdvisorReportingServiceRolePolicy to your IAM entities. For more information, see Using service-linked roles for Trusted Advisor.

This policy grants administrative permissions that allow the service-linked role to perform AWS Organizations actions.

Permissions details

This policy includes the following permissions.

- `organizations` – Describes your organization and lists the service access, accounts, parents, children, and organizational units

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "organizations:DescribeOrganization",
                "organizations:ListAWSServiceAccessForOrganization",
                "organizations:ListAccounts",
                "organizations:ListAccountsForParent",
                "organizations:ListDelegatedAdministrators",
                "organizations:ListOrganizationalUnitsForParent",
                "organizations:ListChildren",
                "organizations:ListParents",
                "organizations:DescribeOrganizationalUnit",
                "organizations:DescribeAccount"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Trusted Advisor updates to AWS managed policies

View details about updates to AWS managed policies for AWS Support and Trusted Advisor since these services began tracking these changes. For automatic alerts about changes to this page, subscribe to the RSS feed on the Document history page.

The following table describes important updates to the Trusted Advisor managed policies since August 10, 2021.

Trusted Advisor

Change	Description	Date
AWSTrustedAdvisorServiceRolePolicy Update to an existing policy.	Trusted Advisor added new actions to grant the `elasticloadbalancing:DescribeListeners`, and `elasticloadbalancing:DescribeRules` permissions.	October 30, 2024
AWSTrustedAdvisorServiceRolePolicy Update to an existing policy.	Trusted Advisor added new actions to grant the `access-analyzer:ListAnalyzers`, `cloudwatch:ListMetrics`, `dax:DescribeClusters`, `ec2:DescribeNatGateways`, `ec2:DescribeRouteTables`, `ec2:DescribeVpcEndpoints`, `ec2:GetManagedPrefixListEntries`, `elasticloadbalancing:DescribeTargetHealth`, `iam:ListSAMLProviders`, `kafka:DescribeClusterV2` `network-firewall:ListFirewalls` `network-firewall:DescribeFirewall` and `sqs:GetQueueAttributes` permissions.	June 11, 2024

Change	Description	Date
AWSTrustedAdvisorServiceRolePolicy Update to an existing policy.	Trusted Advisor added new actions to grant the `cloudtrail:GetTrail` `cloudtrail:ListTrails` `cloudtrail:GetEventSelectors` `outposts:GetOutpost` , `outposts:ListAssets` and `outposts:ListOutposts` permissions.	January 18, 2024
AWSTrustedAdvisorPriorityFullAccess Update to an existing policy.	Trusted Advisor updated the `AWSTrustedAdvisorPriorityFullAccess` AWS managed policy to include statement IDs.	December 6, 2023
AWSTrustedAdvisorPriorityReadOnlyAccess Update to an existing policy.	Trusted Advisor updated the `AWSTrustedAdvisorPriorityReadOnlyAccess` AWS managed policy to include statement IDs.	December 6, 2023
AWSTrustedAdvisorServiceRolePolicy – Update to an existing policy	Trusted Advisor added new actions to grant the `ec2:DescribeRegions` `s3:GetLifecycleConfiguration` `ecs:DescribeTaskDefinition` and `ecs:ListTaskDefinitions` permissions.	November 9, 2023

Change	Description	Date
AWSTrustedAdvisorServiceRolePolicy – Update to an existing policy	Trusted Advisor added new IAM actions `route53resolver:ListResolverEndpoints` , `route53resolver:ListResolverEndpointIpAddresses` , `ec2:DescribeSubnets` , `kafka:ListClustersV2` and `kafka:ListNodes` to onboard new resilience checks.	September 14, 2023
AWSTrustedAdvisorReportingServiceRolePolicy V2 of managed policy attached on Trusted Advisor `AWSServiceRoleForTrustedAdvisorReporting` service-linked role	Upgrade AWS managed policy to V2 for the Trusted Advisor `AWSServiceRoleForTrustedAdvisorReporting` service-linked role. The V2 will add one more IAM action `organizations:ListDelegatedAdministrators`	Feb 28, 2023
AWSTrustedAdvisorPriorityFullAccess and AWSTrustedAdvisorPriorityReadOnlyAccess New AWS managed policies for the Trusted Advisor	Trusted Advisor added two new managed policies that you can use to control access to Trusted Advisor Priority.	August 17, 2022

Change	Description	Date
AWSTrustedAdvisorServiceRolePolicy – Update to an existing policy	Trusted Advisor added new actions to grant the `DescribeTargetGroups` and `GetAccountPublicAccessBlock` permissions. The `DescribeTargetGroup` permission is required for the **Auto Scaling Group Health Check** to retrieve non-Classic Load Balancers that are attached to an Auto Scaling group. The `GetAccountPublicAccessBlock` permission is required for the **Amazon S3 Bucket Permissions** check to retrieve the block public access settings for an AWS account.	August 10, 2021
Change log published	Trusted Advisor started tracking changes for its AWS managed policies.	August 10, 2021

AWS managed policies for AWS Support Plans

AWS Support Plans has the following managed policies.

Contents

AWS managed policy: AWSSupportPlansFullAccess

AWS Support Plans uses the AWSSupportPlansFullAccess AWS managed policy. The IAM entity uses this policy to complete the following Support Plans actions for you:

- View your support plan for your AWS account
- View details about the status for a request to change your support plan
- Change the support plan for your AWS account
- Create support plan schedules for your AWS account
- View a list of all support plan modifiers for your AWS account

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "supportplans:GetSupportPlan",
                "supportplans:GetSupportPlanUpdateStatus",
                "supportplans:StartSupportPlanUpdate",
                "supportplans:CreateSupportPlanSchedule",
                "supportplans:ListSupportPlanModifiers"
            ],
            "Resource": "*"
        }
    ]
}
```

For a list of changes to the policies, see AWS Support Plans updates to AWS managed policies.

AWS managed policy: AWSSupportPlansReadOnlyAccess

AWS Support Plans uses the AWSSupportPlansReadOnlyAccess AWS managed policy. The IAM entity uses this policy to complete the following read-only Support Plans actions for you:

- View your support plan for your AWS account

- View details about the status for a request to change your support plan
- View a list of all support plan modifiers for your AWS account

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "supportplans:GetSupportPlan",
                "supportplans:GetSupportPlanUpdateStatus",
                "supportplans:ListSupportPlanModifiers"
            ],
            "Resource": "*"
        }
    ]
}
```

For a list of changes to the policies, see AWS Support Plans updates to AWS managed policies.

AWS Support Plans updates to AWS managed policies

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

The following table describes important updates to the Support Plans managed policies since September 29, 2022.

AWS Support

Change	Description	Date
AWSSupportPlansReadOnlyAccess - Update to an existing policy	Add `ListSupportPlanModifiers` action to	September 9, 2024

Change	Description	Date
AWSSupportPlansFullAccess - Update to an existing policy	`AWSSupportPlansFullAccess` and `AWSSupportPlansReadOnlyAccess` managed policies.	
AWSSupportPlansFullAccess - Update to an existing policy	Add `CreateSupportPlanSchedule` action to `AWSSupportPlansFullAccess` managed policy.	May 8, 2023
Change log published	Change log for the Support Plans managed policies.	September 29, 2022

AWS managed policies for AWS Partner-Led Support

An AWS managed policy is a standalone policy that is created and administered by AWS. AWS managed policies are designed to provide permissions for many common use cases so that you can start assigning permissions to users, groups, and roles.

Keep in mind that AWS managed policies might not grant least-privilege permissions for your specific use cases because they're available for all AWS customers to use. We recommend that you reduce permissions further by defining customer managed policies that are specific to your use cases.

You cannot change the permissions defined in AWS managed policies. If AWS updates the permissions defined in an AWS managed policy, the update affects all principal identities (users, groups, and roles) that the policy is attached to. AWS is most likely to update an AWS managed policy when a new AWS service is launched or new API operations become available for existing services.

For more information, see AWS managed policies in the *IAM User Guide*.

AWS managed policy: AWSPartnerLedSupportReadOnlyAccess

You can attach `AWSPartnerLedSupportReadOnlyAccess` to your users, groups, and roles.

This policy can be used to grant read-only access to APIs that can read service metadata for services in your AWS account. You can use this policy to provide your partners in the AWS Partner-Led Support Program with access to the services specified in the following permissions details section.

> **Important**
>
> Although AWSPartnerLedSupportReadOnlyAccess is a managed policy provided by AWS, you're responsible for reviewing the services and permissions that are included in the policy to verify that they meet your specific support requirements. Don't assume that this managed policy automatically includes all existing or new AWS services. You might need to create and maintain additional custom policies to cover services outside the scope of this managed policy.

Permissions details

This policy includes the following permissions.

- `acm` – Allow principals to troubleshoot technical support cases related to AWS Certificate Manager.
- `acm-pca` – Allow principals to troubleshoot technical support cases related to AWS Private Certificate Authority.
- `apigateway` – Allow principals to troubleshoot technical support cases related to Amazon API Gateway.
- `athena` – Allow principals to troubleshoot technical support cases related to Amazon Athena.
- `backup` – Allow principals to troubleshoot technical support cases related to AWS Backup.
- `backup-gateway` – Allow principals to troubleshoot technical support cases related to AWS Backup Gateway.
- `cloudformation` – Allow principals to troubleshoot technical support cases related to AWS CloudFormation.
- `cloudfront` – Allow principals to troubleshoot technical support cases related to Amazon CloudFront.

- `cloudtrail` – Allow principals to troubleshoot technical support cases related to AWS CloudTrail.
- `cloudwatch` – Allow principals to troubleshoot technical support cases related to Amazon CloudWatch.
- `codepipeline` – Allow principals to troubleshoot technical support cases related to AWS CodePipeline.
- `cognito-identity` – Allow principals to troubleshoot technical support cases related to Amazon Cognito Identity.
- `cognito-idp` – Allow principals to troubleshoot technical support cases related to Amazon Cognito user pools.
- `cognito-sync` – Allow principals to troubleshoot technical support cases related to Amazon Cognito Sync.
- `connect` – Allow principals to troubleshoot technical support cases related to Amazon Connect.
- `directconnect` – Allow principals to troubleshoot technical support cases related to AWS Direct Connect.
- `dms` – Allow principals to troubleshoot technical support cases related to AWS Database Migration Service.
- `ds` – Allow principals to troubleshoot technical support cases related to AWS Directory Service.
- `ec2` – Allow principals to troubleshoot technical support cases related to Amazon Elastic Compute Cloud. This include technical support categories in EC2 (Windows and Linux), Virtual Private Cloud (VPC) and VPC.
- `ecs` – Allow principals to troubleshoot technical support cases related to Amazon Elastic Container Service.
- `eks` – Allow principals to troubleshoot technical support cases related to Amazon Elastic Kubernetes Service.
- `elasticache` – Allow principals to troubleshoot technical support cases related to Amazon ElastiCache.
- `elasticbeanstalk` – Allow principals to troubleshoot technical support cases related to AWS Elastic Beanstalk.
- `elasticfilesystem` – Allow principals to troubleshoot technical support cases related to Amazon Elastic File System.
- `elasticloadbalancing` – Allow principals to troubleshoot technical support cases related to Elastic Load Balancing.

- `emr-containers` – Allow principals to troubleshoot technical support cases related to Amazon EMR on EKS.
- `emr-serverless` – Allow principals to troubleshoot technical support cases related to Amazon EMR Serverless.
- `es` – Allow principals to troubleshoot technical support cases related to Amazon OpenSearch Service. This includes technical support categories such as OpenSearch Service Managed Cluster.
- `events` – Allow principals to troubleshoot technical support cases related to Amazon EventBridge.
- `fsx` – Allow principals to troubleshoot technical support cases related to Amazon FSx. This includes technical support categories such as FSX for Windows File Server.
- `glue` – Allow principals to troubleshoot technical support cases related to AWS Glue.
- `guardduty` – Allow principals to troubleshoot technical support cases related to Amazon GuardDuty.
- `iam` – Allow principals to troubleshoot technical support cases related to AWS Identity and Access Management.
- `kafka` – Allow principals to troubleshoot technical support cases related to Amazon Managed Streaming for Apache Kafka.
- `kafkaconnect` – Allow principals to troubleshoot technical support cases related to Amazon Managed Streaming for Apache Kafka Connect.
- `lambda` – Allow principals to troubleshoot technical support cases related to AWS Lambda.
- `logs` – Allow principals to troubleshoot technical support cases related to Amazon CloudWatch Logs.
- `medialive` – Allow principals to troubleshoot technical support cases related to AWS Elemental MediaLive.
- `mobiletargeting` – Allow principals to troubleshoot technical support cases related to Amazon Pinpoint.
- `pipes` – Allow principals to troubleshoot technical support cases related to Amazon EventBridge Pipes.
- `polly` – Allow principals to troubleshoot technical support cases related to Amazon Polly.
- `quicksight` – Allow principals to troubleshoot technical support cases related to Amazon Quick Suite.
- `rds` – Allow principals to troubleshoot technical support cases related to Amazon Relational Database Service. This includes technical support categories such as: Relational Database Service

(Aurora - MySQL-Compat), Relational Database Service (Aurora - PostgreSQL-c), Relational Database Service (PostgreSQL), Relational Database Service (SQL Server), Relational Database Service (MySQL) and Relational Database Service (Oracle).

- `redshift` – Allow principals to troubleshoot technical support cases related to Amazon Redshift.
- `redshift-data` – Allow principals to troubleshoot technical support cases related to Amazon Redshift Data API.
- `redshift-serverless` – Allow principals to troubleshoot technical support cases related to Amazon Redshift Serverless.
- `route53` – Allow principals to troubleshoot technical support cases related to Amazon Route 53.
- `route53domains` – Allow principals to troubleshoot technical support cases related to Amazon Route 53 Domains.
- `route53-recovery-cluster` – Allow principals to troubleshoot technical support cases related to Amazon Route 53 Recovery Cluster.
- `route53-recovery-control-config` – Allow principals to troubleshoot technical support cases related to Amazon Route 53 Recovery Controls.
- `route53-recovery-readiness` – Allow principals to troubleshoot technical support cases related to Amazon Route 53 Recovery Readiness.
- `route53resolver` – Allow principals to troubleshoot technical support cases related to Amazon Route 53 Resolver.
- `s3` – Allow principals to troubleshoot technical support cases related to Amazon Simple Storage Service.
- `s3express` – Allow principals to troubleshoot technical support cases related to Amazon S3 Express.
- `sagemaker` – Allow principals to troubleshoot technical support cases related to Amazon SageMaker AI.
- `scheduler` – Allow principals to troubleshoot technical support cases related to Amazon EventBridge Scheduler.
- `servicequotas` – Allow principals to troubleshoot technical support cases related to Service Quotas.
- `ses` – Allow principals to troubleshoot technical support cases related to Amazon Simple Email Service.

- `sns` – Allow principals to troubleshoot technical support cases related to Amazon Simple Notification Service.
- `ssm` – Allow principals to troubleshoot technical support cases related to AWS Systems Manager.
- `ssm-contacts` – Allow principals to troubleshoot technical support cases related to AWS Systems Manager Incident Manager Contacts.
- `ssm-incidents` – Allow principals to troubleshoot technical support cases related to AWS Systems Manager Incident Manager.
- `ssm-sap` – Allow principals to troubleshoot technical support cases related to AWS Systems Manager for SAP.
- `swf` – Allow principals to troubleshoot technical support cases related to Amazon Simple Workflow Service.
- `vpc-lattice` – Allow principals to troubleshoot technical support cases related to Amazon VPC Lattice. This includes technical support categories such as VPC - Transit Gateway.
- `waf` – Allow principals to troubleshoot technical support cases related to AWS WAF.
- `waf-regional` – Allow principals to troubleshoot technical support cases related to AWS WAF Regional.
- `wafv2` – Allow principals to troubleshoot technical support cases related to AWS WAF V2.
- `workspaces` – Allow principals to troubleshoot technical support cases related to Amazon WorkSpaces. This includes technical support categories such as Workspaces (Windows).
- `workspaces-web` – Allow principals to troubleshoot technical support cases related to Amazon WorkSpaces Secure Browser. This includes technical support categories such as Workspaces (Windows).

To view the permissions for this policy, see AWSPartnerLedSupportReadOnlyAccess in the *AWS Managed Policy Reference*.

AWS Partner-Led Support updates to AWS managed policies

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

Change	Description	Date
AWSPartnerLedSupportReadOnlyAccess – New policy	Added a new AWS managed policy that contains permissions that can read service metadata for services in your AWS account.	November 22, 2024
AWS Partner-Led Support started tracking changes	AWS Partner-Led Support started tracking changes for its AWS managed policies.	November 22, 2024

Manage access to AWS Support Center

You must have permissions to access Support Center and to create a support case.

You can use one of the following options to access Support Center:

- Use AWS Identity and Access Management (IAM).
- Use the email address and password associated with your AWS account. This identity is called the AWS account *root user* (not recommended).

If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, you can also use the Support API to access Support and Trusted Advisor operations programmatically. For more information, see the AWS Support API Reference.

> **Note**
>
> If you can't sign in to Support Center, you can use the Contact Us page instead. You can use this page to get help with billing and account issues.

AWS account (not recommended)

You can sign in to the AWS Management Console and access the Support Center by using your AWS account email address and password. This identity is called the AWS account *root user*. However, we strongly recommend that you don't use the root user for your everyday tasks, even

the administrative ones. Instead, we recommend that you use IAM, which lets you control who can perform certain tasks in your account.

AWS support actions

You can perform the following Support actions in the console. You can also specify these Support actions in an IAM policy to allow or deny specific actions.

> **Note**
>
> Denying any of the following actions in your IAM policies, might result in unintended behavior in Support Center when creating or interacting with a support case.

Action	Description
`AddAttachmentsToSet`	Grants permission to add one or more attachments to an attachment set. An attachment set is a temporary container for attachments that you add to a case or case communication. The set is available for 1 hour after it's created. The expiryTime returned in the response is when the set expires.
`AddCommunicationToCase`	Grants permission to add additional customer communication to an Support case, including a set of email addresses to copy on the communication.
`CreateCase`	Grants permission to create a case.
`DescribeAttachment`	Grants permission to retrieve an attachment on a case.
`DescribeCaseAttributes`	Grants permission to allow secondary services to read Support case attributes. ***This is used internally by Support Center to get attributes tagged on your case.**

Action	Description
DescribeCases	Grants permission to return a list of Support cases that matches a case ID or case IDs.
DescribeCommunication	Grants permission to get a single communica tion and attachments for a single AWS Support case.
DescribeCommunications	Grants permission to return communications and attachments for one or more Support cases.
DescribeCreateCaseOptions	Grants permission to return a list of CreateCas eOption types along with the corresponding supported hours and language availability.
DescribeIssueTypes	Grants permission to return issue types for Support cases. This is used internally by Support Center to get available issue types for your account.
DescribeServices	Grants permission to return the current list of AWS services and a list of service categories for each service. You then use service names and categories to create a case. Each AWS service has its own set of categories.
DescribeSeverityLevels	Grants permission to return the list of severity levels that you can assign to a Support case.
DescribeSupportedLanguages	Grants permission to return a list of supported languages for a specified categoryCode, issueType and serviceCode.
DescribeSupportLevel	Grants permission to return the support level for an AWS account identifier. This is used internally by Support Center to identify your support level.

Action	Description
`DescribeTrustedAdvisorCheckRefreshStatuses`	Grants permission to return the refresh status of the AWS Trusted Advisor checks that have the specified check IDs.
`DescribeTrustedAdvisorCheckResult`	Grants permission to return the results of the AWS Trusted Advisor check that has the specified check ID.
`DescribeTrustedAdvisorChecks`	Grants permission to return information about all available AWS Trusted Advisor checks, including the name, ID, category, description, and metadata.
`DescribeTrustedAdvisorCheckSummaries`	Grants permission to return the results for the AWS Trusted Advisor check summaries for the check IDs that you specified.
`GetInteraction`	Grants permission to retrieve details about a specific interaction by its unique identifier. This is used internally by Support Center to retrieve personalized recommendations.
`InitiateCallForCase`	Grants permission to initiate a call on Support Center. This is used internally by Support Center to start a call on your behalf.
`ListInteractionEntries`	Grants permission to retrieve a list of entries within a specific interaction, including messages, status updates, or other relevant data points. This is used internally by Support Center to track the detailed trail of an interacti on.

Action	Description
ListInteractions	Grants permission to retrieve a list of interactions, potentially with filters or pagination. This is used internally by Support Center to manage and overview multiple interactions.
InitiateChatForCase	Grants permission to initiate a chat on Support Center. This is used internally by Support Center to start a chat on your behalf.
PutCaseAttributes	Grants permission to allow secondary services to attach attributes to Support cases. This is used internally by Support Center to add operational tags to your Support cases.
RateCaseCommunication	Grants permission to rate a Support case communication.
RefreshTrustedAdvisorCheck	Grants permission to refresh the AWS Trusted Advisor check that you specify using the check ID.
ResolveCase	Grants permission to resolve a Support case.
ResolveInteraction	Grants permission to mark an interaction as resolved using its unique identifier, indicating that the issue has been fully addressed and requires no further action. Once resolved, the interaction's status is set to CLOSED and becomes accessible to all users within the same account.
SearchForCases	Grants permission to return a list of Support cases that matches the given inputs. This is used internally by Support Center to find searched cases.

Action	Description
`StartInteraction`	Grants permission to initiate a new interacti on to receive personalized troubleshooting assistance for account and technical issues. This is used internally by Support Center to initiate the troubleshooting process.
`UpdateInteraction`	Grants permission to update a specific interaction by its unique identifier with another message. This is used internally by Support Center to update the troubleshooting process.

IAM

By default, IAM users can't access the Support Center. You can use IAM to create individual users or groups. Then, you attach IAM policies to these entities, so that they have permission to perform actions and access resources, such as to open Support Center cases and use the Support API.

After you create IAM users, you can give those users individual passwords and an account-specific sign-in page. They can then sign in to your AWS account and work in the Support Center. IAM users who have AWS Support access can see all cases that are created for the account.

For more information, see Sign in to the AWS Management Console as an IAM user in the *IAM User Guide*.

The easiest way to grant permissions is to attach the AWS managed policy AWSSupportAccess to the user, group, or role. AWS Support allows action-level permissions to control access to specific AWS Support operations. AWS Support doesn't provide resource-level access, so the `Resource` element is always set to `*`. You can't allow or deny access to specific support cases.

Example : Allow access to all Support actions

The AWS managed policy AWSSupportAccess grants an IAM user access to Support. An IAM user with this policy can access all AWS Support operations and resources.

JSON

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

For more information about how to attach the `AWSSupportAccess` policy to your entities, see Adding IAM identity permissions (console) in the *IAM User Guide*.

Example : Allow access to all actions except the ResolveCase action

You can also create *customer managed policies* in IAM to specify what actions to allow or deny. The following policy statement allows an IAM user to perform all actions in Support except resolve a case.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
   {
      "Effect": "Allow",
      "Action": "support:*",
      "Resource": "*"
   },
   {
       "Effect": "Deny",
       "Action": "support:ResolveCase",
       "Resource": "*"
    }]
}
```

For more information about how to create a customer managed IAM policy, see Creating IAM policies (console) in the *IAM User Guide*.

If the user or group already has a policy, you can add the AWS Support-specific policy statement to that policy.

Important

- If you can't view cases in the Support Center, make sure that you have the required permissions. You might need to contact your IAM administrator. For more information, see Identity and access management for AWS Support.

Access to AWS Trusted Advisor

In the AWS Management Console, a separate `trustedadvisor` IAM namespace controls access to Trusted Advisor. In the Support API, the `support` IAM namespace controls access to Trusted Advisor. For more information, see Manage access to AWS Trusted Advisor.

Manage access to AWS Support Plans

Topics

Permissions for the Support Plans console

To access the Support Plans console, a user must have a minimum set of permissions. These permissions must allow the user to list and view details about the Support Plans resources in your AWS account.

You can create an AWS Identity and Access Management (IAM) policy with the `supportplans` namespace. You can use this policy to specify permissions for actions and resources.

When you create a policy, you can specify the namespace of the service to allow or deny an action. The namespace for Support Plans is `supportplans`.

You can use AWS managed policies and attach them to your IAM entities. For more information, see AWS managed policies for AWS Support Plans.

Support Plans actions

You can perform the following Support Plans actions in the console. You can also specify these Support Plans actions in an IAM policy to allow or deny specific actions.

Action	Description
`GetSupportPlan`	Grants permission to view details about the current support plan for this AWS account.
`GetSupportPlanUpdateStatus`	Grants permission to view details about the status for a request to update a support plan.
`StartSupportPlanUpdate`	Grants permission to start the request to update the support plan for this AWS account.
`CreateSupportPlanSchedule`	Grants permission to create support plan schedules for this AWS account.
`ListSupportPlanModifiers`	Grants permission to view a list of all support plan modifiers for this AWS account.

Example IAM policies for Support Plans

You can use the following example policies to manage access to Support Plans.

Full access to Support Plans

The following policy allows users full access to Support Plans.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Action": "supportplans:*",
            "Resource": "*"
        }
    ]
}
```

Read-only access to Support Plans

The following policy allows read-only access to Support Plans.

JSON

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

Deny access to Support Plans

The following policy doesn't allow users access to Support Plans.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Deny",
            "Action": "supportplans:*",
            "Resource": "*"
        }
    ]
}
```

Troubleshooting

See the following topics to manage access to Support Plans.

When I try to view or change my support plan, the Support Plans console says that I'm missing the `GetSupportPlan` permission

IAM users must have the required permissions to access the Support Plans console. You can update your IAM policy to include the missing permission or use an AWS managed policy, such as AWSSupportPlansFullAccess or AWSSupportPlansReadOnlyAccess. For more information, see AWS managed policies for AWS Support Plans.

If you don't have access to update your IAM policies, contact your AWS account administrator.

Related information

For more information, see the following topics in the *IAM User Guide*:

- Testing IAM policies with the IAM policy simulator
- Troubleshooting access denied error messages

I have the correct Support Plans permissions, but I still get the same error

If your AWS account is a member account that's part of AWS Organizations, the service control policy (SCP) might need to be updated. SCPs are a type of policy that manages permissions in an organization.

Because Support Plans is a *global* service, policies that restrict AWS Regions might prevent member accounts from viewing or changing their support plan. To allow global services for your organization, such as IAM and Support Plans, you must add the service to the exclusion list in any applicable SCP. This means that accounts in the organization can access these services, even if the SCP denies a specified AWS Region.

To add Support Plans as an exception, enter `"supportplans:*"` to the `"NotAction"` list in the SCP.

```
"supportplans:*",
```

Your SCP might appear as the following policy snippet.

Example : SCP that allows Support Plans access in an organization

```
{ "Version": "2012-10-17",
   "Statement": [
     { "Sid":  "GRREGIONDENY",
       "Effect":  "Deny",
       "NotAction": [
         "aws-portal:*",
         "budgets:*",
         "chime:*"
         "iam:*",
         "supportplans:*",
         ....
```

If you have a member account and can't update the SCP, contact your AWS account administrator. The management account might need to update the SCP so that all member accounts can access Support Plans.

Notes for AWS Control Tower

- If your organization uses an SCP with AWS Control Tower, you can update the **Deny access to AWS based on the requested AWS Region** control (commonly referred to as the Region deny control).
- If you update the SCP for AWS Control Tower to allow `supportplans`, repairing the drift will remove your update to the SCP. For more information, see Detect and resolve drift in AWS Control Tower.

Related information

For more information, see the following topics:

- Service control policies (SCPs) in the *AWS Organizations User Guide.*

- Configure the Region deny control in the *AWS Control Tower User Guide*
- Deny access to AWS based on the requested AWS Region in the *AWS Control Tower User Guide*

Manage access to AWS Trusted Advisor

You can access AWS Trusted Advisor from the AWS Management Console. All AWS accounts have access to a select core Trusted Advisor checks. If you have a AWS Business Support+, AWS Enterprise Support, or AWS Unified Operations plan, you can access all checks. for more information, see AWS Trusted Advisor check reference.

You can use AWS Identity and Access Management (IAM) to control access to Trusted Advisor.

Topics

Permissions for the Trusted Advisor console

To access the Trusted Advisor console, a user must have a minimum set of permissions. These permissions must allow the user to list and view details about the Trusted Advisor resources in your AWS account.

You can use the following options to control access to Trusted Advisor:

- Use the tag filter feature of the Trusted Advisor console. The user or role must have permissions associated with the tags.

  You can use AWS managed policies or custom policies to assign permissions by tags. For more information, see Controlling access to and for IAM users and roles using tags.
- Create an IAM policy with the `trustedadvisor` namespace. You can use this policy to specify permissions for actions and resources.

When you create a policy, you can specify the namespace of the service to allow or deny an action. The namespace for Trusted Advisor is `trustedadvisor`. However, you can't use the

`trustedadvisor` namespace to allow or deny Trusted Advisor API operations in the Support API. You must use the `support` namespace for Support instead.

> **Note**
>
> If you have permissions to the AWS Support API, the Trusted Advisor widget in the AWS Management Console shows a summary view of your Trusted Advisor results. To view your results in the Trusted Advisor console, you must have permission to the `trustedadvisor` namespace.

Trusted Advisor actions

You can perform the following Trusted Advisor actions in the console. You can also specify these Trusted Advisor actions in an IAM policy to allow or deny specific actions.

Action	Description
`DescribeAccount`	Grants permission to view the Support plan and various Trusted Advisor preferences.
`DescribeAccountAccess`	Grants permission to view if the AWS account has enabled or disabled Trusted Advisor.
`DescribeCheckItems`	Grants permission to view details for the check items.
`DescribeCheckRefreshStatuses`	Grants permission to view the refresh statuses for Trusted Advisor checks.
`DescribeCheckSummaries`	Grants permission to view Trusted Advisor check summaries.
`DescribeChecks`	Grants permission to view details for Trusted Advisor checks.
`DescribeNotificationPreferences`	Grants permission to view the notification preferences for the AWS account.

Action	Description
ExcludeCheckItems	Grants permission to exclude recommend ations for Trusted Advisor checks.
IncludeCheckItems	Grants permission to include recommend ations for Trusted Advisor checks.
RefreshCheck	Grants permission to refresh a Trusted Advisor check.
SetAccountAccess	Grants permission to enable or disable Trusted Advisor for the account.
UpdateNotificationPreferences	Grants permission to update notification preferences for Trusted Advisor.
DescribeCheckStatusHistoryC hanges	Grants permission to view the results and changed statuses for checks in the last 30 days.

Trusted Advisor actions for organizational view

The following Trusted Advisor actions are for the organizational view feature. For more information, see Organizational view for AWS Trusted Advisor.

Action	Description
DescribeOrganization	Grants permission to view if the AWS account meets the requirements to enable the organizational view feature.
DescribeOrganizationAccounts	Grants permission to view the linked AWS accounts that are in the organization.
DescribeReports	Grants permission to view details for organizat ional view reports, such as the report name, runtime, date created, status, and format.

Action	Description
`DescribeServiceMetadata`	Grants permission to view information about organizational view reports, such as the AWS Regions, check categories, check names, and resource statuses.
`GenerateReport`	Grants permission to create a report for Trusted Advisor checks in your organization.
`ListAccountsForParent`	Grants permission to view, in the Trusted Advisor console, all of the accounts in an AWS organization that are contained by a root or organizational unit (OU).
`ListOrganizationalUnitsForParent`	Grants permission to view, in the Trusted Advisor console, all of the organizational units (OUs) in a parent organizational unit or root.
`ListRoots`	Grants permission to view, in the Trusted Advisor console, all of the roots that are defined in an AWS organization.
`SetOrganizationAccess`	Grants permission to enable the organizat ional view feature for Trusted Advisor.

Trusted Advisor Priority actions

If you have Trusted Advisor Priority enabled for your account, you can perform the following Trusted Advisor actions in the console. You can also add these Trusted Advisor actions in an IAM policy to allow or deny specific actions. For more information, see Example IAM policies for Trusted Advisor Priority.

> **Note**
>
> The risks that appear in Trusted Advisor Priority are recommendations that your technical account manager (TAM) has identified for your account. Recommendations from a service, such as a Trusted Advisor check, are created for you automatically. Recommendations from

your TAM are created for you manually. Next, your TAM sends these recommendations so that they appear in Trusted Advisor Priority for your account.

For more information, see Get started with AWS Trusted Advisor Priority.

Action	Description
`DescribeRisks`	Grants permission to view risks in Trusted Advisor Priority.
`DescribeRisk`	Grants permission to view risk details in Trusted Advisor Priority.
`DescribeRiskResources`	Grants permission to view affected resources for a risk in Trusted Advisor Priority.
`DownloadRisk`	Grants permission to download a file that contains details about the risk in Trusted Advisor Priority.
`UpdateRiskStatus`	Grants permission to update the risk status in Trusted Advisor Priority.
`DescribeNotificationConfigurations`	Grants permission to get your email notification preferences for Trusted Advisor Priority.
`UpdateNotificationConfigurations`	Grants permission to create or update your email notification preferences for Trusted Advisor Priority.
`DeleteNotificationConfigurationForDelegatedAdmin`	Grants permission to the organization management account to delete email notification preferences from a delegated administrator account for Trusted Advisor Priority.

IAM policy examples

The following policies show you how to allow and deny access to Trusted Advisor. You can use one of the following policies to create a *customer managed policy* in the IAM console. For example, you can copy an example policy, and then paste it into the JSON tab of the IAM console. Then, you attach the policy to your IAM user, group, or role.

For more information about how to create an IAM policy, see Creating IAM policies (console) in the *IAM User Guide*.

Examples

- Full access to Trusted Advisor
- Read-only access to Trusted Advisor
- Deny access to Trusted Advisor
- Allow and deny specific actions
- Control access to the Support API operations for Trusted Advisor
- Example IAM policies for Trusted Advisor Priority

Full access to Trusted Advisor

The following policy allows users to view and take all actions on all Trusted Advisor checks in the Trusted Advisor console.

JSON

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

Read-only access to Trusted Advisor

The following policy allows users read-only access to the Trusted Advisor console. Users can't make changes, such as refresh checks or change notification preferences.

JSON

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

Deny access to Trusted Advisor

The following policy doesn't allow users to view or take actions for Trusted Advisor checks in the Trusted Advisor console.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "trustedadvisor:*",
            "Resource": "*"
        }
    ]
}
```

Allow and deny specific actions

The following policy allows users to view all Trusted Advisor checks in the Trusted Advisor console, but doesn't allow them to refresh any checks.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "trustedadvisor:*",
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": "trustedadvisor:RefreshCheck",
            "Resource": "*"
        }
    ]
}
```

Control access to the Support API operations for Trusted Advisor

In the AWS Management Console, a separate `trustedadvisor` IAM namespace controls access to Trusted Advisor. You can't use the `trustedadvisor` namespace to allow or deny Trusted Advisor API operations in the Support API. Instead, you use the `support` namespace. You must have permissions to the Support API to call Trusted Advisor programmatically.

For example, if you want to call the RefreshTrustedAdvisorCheck operation, you must have permissions to this action in the policy.

Example : Allow Trusted Advisor API operations only

The following policy allows users access to the Support API operations for Trusted Advisor, but not the rest of the Support API operations. For example, users can use the API to view and refresh checks. They can't create, view, update, or resolve AWS Support cases.

You can use this policy to call the Trusted Advisor API operations programmatically, but you can't use this policy to view or refresh checks in the Trusted Advisor console.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "support:DescribeTrustedAdvisorCheckRefreshStatuses",
                "support:DescribeTrustedAdvisorCheckResult",
                "support:DescribeTrustedAdvisorChecks",
                "support:DescribeTrustedAdvisorCheckSummaries",
                "support:RefreshTrustedAdvisorCheck",
                "trustedadvisor:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "support:AddAttachmentsToSet",
                "support:AddCommunicationToCase",
                "support:CreateCase",
                "support:DescribeAttachment",
                "support:DescribeCases",
                "support:DescribeCommunications",
                "support:DescribeServices",
                "support:DescribeSeverityLevels",
                "support:ResolveCase"
            ],
            "Resource": "*"
        }
    ]
}
```

For more information about how IAM works with Support and Trusted Advisor, see Actions.

Example IAM policies for Trusted Advisor Priority

You can use the following AWS managed policies to control access to Trusted Advisor Priority. For more information, see AWS managed policies for AWS Trusted Advisor and Get started with AWS Trusted Advisor Priority.

See also

For more information about Trusted Advisor permissions, see the following resources:

- Actions defined by AWS Trusted Advisor in the *IAM User Guide*.
- Controlling Access to the Trusted Advisor Console

Example Service Control Policies for AWS Trusted Advisor

AWS Trusted Advisor supports service control policies (SCPs). SCPs are policies that you attach to elements in an organization to manage permissions within that organization. An SCP applies to all AWS accounts under the element to which you attach the SCP. SCPs offer central control over the maximum available permissions for all accounts in your organization. They can help you to ensure your AWS accounts stay within your organization's access control guidelines. For more information, see Service control policies in the *AWS Organizations User Guide*.

Topics

- Prerequisites
- Example Service Control Policies

Prerequisites

To use SCPs, you must first do the following:

- Enable all features in your organization. For more information, see Enabling all features in your organization in the *AWS Organizations User Guide*.
- Enable SCPs for use within your organization. For more information, see Enabling and disabling policy types in the *AWS Organizations User Guide*.
- Create the SCPs that you need. For more information about creating SCPs, see Creating, updating, and deleting service control policies in the *AWS Organizations User Guide*.

Example Service Control Policies

The following examples show how you can control various aspects of resource sharing in an organization.

Example : Prevent users from creating or editing engagements in Trusted Advisor Engage

The following SCP prevents users from creating new engagements or editing existing engagements.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "trustedadvisor:CreateEngagement",
        "trustedadvisor:UpdateEngagement*"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

Example : Deny Trusted Advisor Engage and Trusted Advisor Priority Access

The following SCP prevents users from accessing or performing any actions within Trusted Advisor Engage and Trusted Advisor Priority.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
```

```
      "Action": [
        "trustedadvisor:ListEngagement*",
        "trustedadvisor:GetEngagement*",
        "trustedadvisor:CreateEngagement*",
        "trustedadvisor:UpdateEngagement*",
        "trustedadvisor:DescribeRisk*",
        "trustedadvisor:UpdateRisk*",
        "trustedadvisor:DownloadRisk"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

Troubleshooting AWS Support identity and access

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

Topics

I'm not authorized to perform iam:PassRole

If you receive an error that you're not authorized to perform the `iam:PassRole` action, your policies must be updated to allow you to pass a role to Support.

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 Support. However, the action requires the service to have permissions that are 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's policies must be updated to allow her to perform the `iam:PassRole` action.

If you need help, contact your AWS administrator. Your administrator is the person who provided you with your sign-in credentials.

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 AWS 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 Support

To allow others to access Support, you must grant permission to the people or applications that need access. If you are using AWS IAM Identity Center to manage people and applications, you assign permission sets to users or groups to define their level of access. Permission sets automatically create and assign IAM policies to IAM roles that are associated with the person or application. For more information, see Permission sets in the *AWS IAM Identity Center User Guide*.

If you are not using IAM Identity Center, you must create IAM entities (users or roles) for the people or applications that need access. You must then attach a policy to the entity that grants them the correct permissions in Support. After the permissions are granted, provide the credentials to the user or application developer. They will use those credentials to access AWS. To learn more about creating IAM users, groups, policies, and permissions, see IAM Identities and Policies and permissions in IAM in the *IAM User Guide*.

I want to allow people outside of my AWS account to access my Support 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 Support supports these features, see How AWS Support works with IAM.
- 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 Cross account resource access in IAM in the *IAM User Guide*.

Incident response

Incident response for Support is an AWS responsibility. AWS has a formal, documented policy and program that governs incident response. For more information, see the AWS Security Incident Response Technical Guide.

Use the following options to inform yourself about operational issues:

- View AWS operational issues with broad impact on the AWS Service Health Dashboard. For example, events that affect a service or Region that isn't specific to your account.

- View operational issues for individual accounts in the AWS Health Dashboard. For example, events that affect services or resources in your account. For more information, see Getting started with the AWS Health Dashboard in the *AWS Health User Guide*.

Logging and monitoring in AWS Support and AWS Trusted Advisor

Monitoring is an important part of maintaining the reliability, availability, and performance of AWS Support and AWS Trusted Advisor and your other AWS solutions. AWS provides the following monitoring tools to watch AWS Support and AWS Trusted Advisor, report when something is wrong, and take actions when appropriate:

- *Amazon CloudWatch* monitors your AWS resources and the applications that you run on AWS in real time. You can collect and track metrics, create customized dashboards, and set alarms that notify you or take actions when a specified metric reaches a threshold that you specify. For example, you can have CloudWatch track CPU usage or other metrics of your Amazon Elastic Compute Cloud (Amazon EC2) instances and automatically launch new instances when needed. For more information, see the Amazon CloudWatch User Guide.
- *Amazon EventBridge* delivers a near real-time stream of system events that describe changes in AWS resources. EventBridge enables automated event-driven computing, as you can write rules that watch for certain events and trigger automated actions in other AWS services when these events happen. For more information, see the Amazon EventBridge User Guide.
- *AWS CloudTrail* captures API calls and related events made by or on behalf of your AWS account and delivers the log files to an Amazon Simple Storage Service (Amazon S3) bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see the AWS CloudTrail User Guide.

For more information, see Monitoring and logging for AWS Support and Monitoring and logging for AWS Trusted Advisor.

Compliance validation for AWS Support

To learn whether an AWS service is within the scope of specific compliance programs, see and choose the compliance program that you are interested in. For general information, see .

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. For more information about your compliance responsibility when using AWS services, see AWS Security Documentation.

Resilience in AWS Support

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.

Infrastructure security in AWS Support

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

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

Configuration and vulnerability analysis in Support

For AWS Trusted Advisor, AWS handles basic security tasks such as guest operating system (OS) and database patching, firewall configuration, and disaster recovery.

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

Code examples for Support using AWS SDKs

The following code examples show how to use Support with an AWS software development kit (SDK).

Basics are code examples that show you how to perform the essential operations within a service.

Actions are code excerpts from larger programs and must be run in context. While actions show you how to call individual service functions, you can see actions in context in their related scenarios.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Get started

Hello Support

The following code examples show how to get started using Support.

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
using Amazon.AWSSupport;
using Microsoft.Extensions.DependencyInjection;
using Microsoft.Extensions.Hosting;

public static class HelloSupport
{
    static async Task Main(string[] args)
    {
        // Use the AWS .NET Core Setup package to set up dependency injection for
 the AWS Support service.
```

```
        // Use your AWS profile name, or leave it blank to use the default
 profile.
        // You must have one of the following AWS Support plans: Business,
 Enterprise On-Ramp, or Enterprise. Otherwise, an exception will be thrown.
        using var host = Host.CreateDefaultBuilder(args)
            .ConfigureServices((_, services) =>
                services.AddAWSService<IAmazonAWSSupport>()
            ).Build();

        // Now the client is available for injection.
        var supportClient =
 host.Services.GetRequiredService<IAmazonAWSSupport>();

        // You can use await and any of the async methods to get a response.
        var response = await supportClient.DescribeServicesAsync();
        Console.WriteLine($"\tHello AWS Support! There are
 {response.Services.Count} services available.");
    }
}
```

- For API details, see DescribeServices in *AWS SDK for .NET API Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.support.SupportClient;
import software.amazon.awssdk.services.support.model.Category;
import software.amazon.awssdk.services.support.model.DescribeServicesRequest;
import software.amazon.awssdk.services.support.model.DescribeServicesResponse;
import software.amazon.awssdk.services.support.model.Service;
import software.amazon.awssdk.services.support.model.SupportException;
import java.util.ArrayList;
import java.util.List;
```

```
/**
 * Before running this Java (v2) code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 *
 * In addition, you must have the AWS Business Support Plan to use the AWS
 * Support Java API. For more information, see:
 *
 * https://aws.amazon.com/premiumsupport/plans/
 *
 * This Java example performs the following task:
 *
 * 1. Gets and displays available services.
 *
 *
 * NOTE: To see multiple operations, see SupportScenario.
 */

public class HelloSupport {
    public static void main(String[] args) {
        Region region = Region.US_WEST_2;
        SupportClient supportClient = SupportClient.builder()
                .region(region)
                .build();

        System.out.println("***** Step 1. Get and display available services.");
        displayServices(supportClient);
    }

    // Return a List that contains a Service name and Category name.
    public static void displayServices(SupportClient supportClient) {
        try {
            DescribeServicesRequest servicesRequest =
 DescribeServicesRequest.builder()
                    .language("en")
                    .build();

            DescribeServicesResponse response =
 supportClient.describeServices(servicesRequest);
```

```
            List<Service> services = response.services();

            System.out.println("Get the first 10 services");
            int index = 1;
            for (Service service : services) {
                if (index == 11)
                    break;

                System.out.println("The Service name is: " + service.name());

                // Display the Categories for this service.
                List<Category> categories = service.categories();
                for (Category cat : categories) {
                    System.out.println("The category name is: " + cat.name());
                }
                index++;
            }

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
}
```

- For API details, see DescribeServices in *AWS SDK for Java 2.x API Reference*.

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

Invoke `main()` to run the example.

```
import {
  DescribeServicesCommand,
```

```
  SupportClient,
} from "@aws-sdk/client-support";

// Change the value of 'region' to your preferred AWS Region.
const client = new SupportClient({ region: "us-east-1" });

const getServiceCount = async () => {
  try {
    const { services } = await client.send(new DescribeServicesCommand({}));
    return services.length;
  } catch (err) {
    if (err.name === "SubscriptionRequiredException") {
      throw new Error(
        "You must be subscribed to the AWS Support plan to use this feature.",
      );
    }
    throw err;
  }
};

export const main = async () => {
  try {
    const count = await getServiceCount();
    console.log(`Hello, AWS Support! There are ${count} services available.`);
  } catch (err) {
    console.error("Failed to get service count: ", err.message);
  }
};
```

- For API details, see DescribeServices in *AWS SDK for JavaScript API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/**
Before running this Kotlin code example, set up your development environment,
including your credentials.

For more information, see the following documentation topic:
https://docs.aws.amazon.com/sdk-for-kotlin/latest/developer-guide/setup.html

In addition, you must have the AWS Business Support Plan to use the AWS Support
 Java API. For more information, see:

https://aws.amazon.com/premiumsupport/plans/

This Kotlin example performs the following task:

1. Gets and displays available services.
 */

suspend fun main() {
    displaySomeServices()
}

// Return a List that contains a Service name and Category name.
suspend fun displaySomeServices() {
    val servicesRequest =
        DescribeServicesRequest {
            language = "en"
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.describeServices(servicesRequest)
        println("Get the first 10 services")
        var index = 1

        response.services?.forEach { service ->
            if (index == 11) {
                return@forEach
            }

            println("The Service name is: " + service.name)

            // Get the categories for this service.
            service.categories?.forEach { cat ->
```

```
                println("The category name is ${cat.name}")
                index++
            }
        }
    }
}
```

- For API details, see DescribeServices in *AWS SDK for Kotlin API reference*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import logging
import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)


def hello_support(support_client):
    """
    Use the AWS SDK for Python (Boto3) to create an AWS Support client and count
    the available services in your account.
    This example uses the default settings specified in your shared credentials
    and config files.

    :param support_client: A Boto3 Support Client object.
    """
    try:
        print("Hello, AWS Support! Let's count the available Support services:")
        response = support_client.describe_services()
        print(f"There are {len(response['services'])} services available.")
    except ClientError as err:
```

```
        if err.response["Error"]["Code"] == "SubscriptionRequiredException":
            logger.info(
                "You must have a Business, Enterprise On-Ramp, or Enterprise
 Support "
                "plan to use the AWS Support API. \n\tPlease upgrade your
 subscription to run these "
                "examples."
            )
        else:
            logger.error(
                "Couldn't count services. Here's why: %s: %s",
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise


if __name__ == "__main__":
    hello_support(boto3.client("support"))
```

- For API details, see DescribeServices in *AWS SDK for Python (Boto3) API Reference*.

Code examples

- Basic examples for Support using AWS SDKs
  - Hello Support
  - Learn the basics of Support with an AWS SDK
  - Actions for Support using AWS SDKs
    - Use AddAttachmentsToSet with an AWS SDK or CLI
    - Use AddCommunicationToCase with an AWS SDK or CLI
    - Use CreateCase with an AWS SDK or CLI
    - Use DescribeAttachment with an AWS SDK or CLI
    - Use DescribeCases with an AWS SDK or CLI
    - Use DescribeCommunications with an AWS SDK or CLI
    - Use DescribeServices with an AWS SDK or CLI
    - Use DescribeSeverityLevels with an AWS SDK or CLI
    - Use DescribeTrustedAdvisorCheckRefreshStatuses with a CLI

- Use DescribeTrustedAdvisorCheckResult with a CLI
- Use DescribeTrustedAdvisorCheckSummaries with a CLI
- Use DescribeTrustedAdvisorChecks with a CLI
- Use RefreshTrustedAdvisorCheck with a CLI
- Use ResolveCase with an AWS SDK or CLI

Basic examples for Support using AWS SDKs

The following code examples show how to use the basics of AWS Support with AWS SDKs.

Examples

- Hello Support
- Learn the basics of Support with an AWS SDK
- Actions for Support using AWS SDKs
  - Use AddAttachmentsToSet with an AWS SDK or CLI
  - Use AddCommunicationToCase with an AWS SDK or CLI
  - Use CreateCase with an AWS SDK or CLI
  - Use DescribeAttachment with an AWS SDK or CLI
  - Use DescribeCases with an AWS SDK or CLI
  - Use DescribeCommunications with an AWS SDK or CLI
  - Use DescribeServices with an AWS SDK or CLI
  - Use DescribeSeverityLevels with an AWS SDK or CLI
  - Use DescribeTrustedAdvisorCheckRefreshStatuses with a CLI
  - Use DescribeTrustedAdvisorCheckResult with a CLI
  - Use DescribeTrustedAdvisorCheckSummaries with a CLI
  - Use DescribeTrustedAdvisorChecks with a CLI
  - Use RefreshTrustedAdvisorCheck with a CLI
  - Use ResolveCase with an AWS SDK or CLI

Hello Support

The following code examples show how to get started using Support.

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
using Amazon.AWSSupport;
using Microsoft.Extensions.DependencyInjection;
using Microsoft.Extensions.Hosting;

public static class HelloSupport
{
    static async Task Main(string[] args)
    {
        // Use the AWS .NET Core Setup package to set up dependency injection for
 the AWS Support service.
        // Use your AWS profile name, or leave it blank to use the default
 profile.
        // You must have one of the following AWS Support plans: Business,
 Enterprise On-Ramp, or Enterprise. Otherwise, an exception will be thrown.
        using var host = Host.CreateDefaultBuilder(args)
            .ConfigureServices((_, services) =>
                services.AddAWSService<IAmazonAWSSupport>()
            ).Build();

        // Now the client is available for injection.
        var supportClient =
 host.Services.GetRequiredService<IAmazonAWSSupport>();

        // You can use await and any of the async methods to get a response.
        var response = await supportClient.DescribeServicesAsync();
        Console.WriteLine($"\tHello AWS Support! There are
 {response.Services.Count} services available.");
    }
}
```

- For API details, see DescribeServices in *AWS SDK for .NET API Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.support.SupportClient;
import software.amazon.awssdk.services.support.model.Category;
import software.amazon.awssdk.services.support.model.DescribeServicesRequest;
import software.amazon.awssdk.services.support.model.DescribeServicesResponse;
import software.amazon.awssdk.services.support.model.Service;
import software.amazon.awssdk.services.support.model.SupportException;
import java.util.ArrayList;
import java.util.List;

/**
 * Before running this Java (v2) code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 *
 * In addition, you must have the AWS Business Support Plan to use the AWS
 * Support Java API. For more information, see:
 *
 * https://aws.amazon.com/premiumsupport/plans/
 *
 * This Java example performs the following task:
 *
 * 1. Gets and displays available services.
 *
 *
 * NOTE: To see multiple operations, see SupportScenario.
```

```
 */

public class HelloSupport {
    public static void main(String[] args) {
        Region region = Region.US_WEST_2;
        SupportClient supportClient = SupportClient.builder()
                .region(region)
                .build();

        System.out.println("***** Step 1. Get and display available services.");
        displayServices(supportClient);
    }

    // Return a List that contains a Service name and Category name.
    public static void displayServices(SupportClient supportClient) {
        try {
            DescribeServicesRequest servicesRequest =
 DescribeServicesRequest.builder()
                    .language("en")
                    .build();

            DescribeServicesResponse response =
 supportClient.describeServices(servicesRequest);
            List<Service> services = response.services();

            System.out.println("Get the first 10 services");
            int index = 1;
            for (Service service : services) {
                if (index == 11)
                    break;

                System.out.println("The Service name is: " + service.name());

                // Display the Categories for this service.
                List<Category> categories = service.categories();
                for (Category cat : categories) {
                    System.out.println("The category name is: " + cat.name());
                }
                index++;
            }

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
```

```
        }
    }
}
```

- For API details, see DescribeServices in *AWS SDK for Java 2.x API Reference*.

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

Invoke `main()` to run the example.

```
import {
  DescribeServicesCommand,
  SupportClient,
} from "@aws-sdk/client-support";

// Change the value of 'region' to your preferred AWS Region.
const client = new SupportClient({ region: "us-east-1" });

const getServiceCount = async () => {
  try {
    const { services } = await client.send(new DescribeServicesCommand({}));
    return services.length;
  } catch (err) {
    if (err.name === "SubscriptionRequiredException") {
      throw new Error(
        "You must be subscribed to the AWS Support plan to use this feature.",
      );
    }
    throw err;
  }
};

export const main = async () => {
```

```
  try {
    const count = await getServiceCount();
    console.log(`Hello, AWS Support! There are ${count} services available.`);
  } catch (err) {
    console.error("Failed to get service count: ", err.message);
  }
};
```

- For API details, see DescribeServices in *AWS SDK for JavaScript API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/**
Before running this Kotlin code example, set up your development environment,
including your credentials.

For more information, see the following documentation topic:
https://docs.aws.amazon.com/sdk-for-kotlin/latest/developer-guide/setup.html

In addition, you must have the AWS Business Support Plan to use the AWS Support
 Java API. For more information, see:

https://aws.amazon.com/premiumsupport/plans/

This Kotlin example performs the following task:

1. Gets and displays available services.
 */

suspend fun main() {
    displaySomeServices()
}
```

```
// Return a List that contains a Service name and Category name.
suspend fun displaySomeServices() {
    val servicesRequest =
        DescribeServicesRequest {
            language = "en"
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.describeServices(servicesRequest)
        println("Get the first 10 services")
        var index = 1

        response.services?.forEach { service ->
            if (index == 11) {
                return@forEach
            }

            println("The Service name is: " + service.name)

            // Get the categories for this service.
            service.categories?.forEach { cat ->
                println("The category name is ${cat.name}")
                index++
            }
        }
    }
}
```

- For API details, see DescribeServices in *AWS SDK for Kotlin API reference*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import logging
import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)


def hello_support(support_client):
    """
    Use the AWS SDK for Python (Boto3) to create an AWS Support client and count
    the available services in your account.
    This example uses the default settings specified in your shared credentials
    and config files.

    :param support_client: A Boto3 Support Client object.
    """
    try:
        print("Hello, AWS Support! Let's count the available Support services:")
        response = support_client.describe_services()
        print(f"There are {len(response['services'])} services available.")
    except ClientError as err:
        if err.response["Error"]["Code"] == "SubscriptionRequiredException":
            logger.info(
                "You must have a Business, Enterprise On-Ramp, or Enterprise
 Support "
                "plan to use the AWS Support API. \n\tPlease upgrade your
 subscription to run these "
                "examples."
            )
        else:
            logger.error(
                "Couldn't count services. Here's why: %s: %s",
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise


if __name__ == "__main__":
    hello_support(boto3.client("support"))
```

- For API details, see DescribeServices in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Learn the basics of Support with an AWS SDK

The following code examples show how to:

- Get and display available services and severity levels for cases.
- Create a support case using a selected service, category, and severity level.
- Get and display a list of open cases for the current day.
- Add an attachment set and a communication to the new case.
- Describe the new attachment and communication for the case.
- Resolve the case.
- Get and display a list of resolved cases for the current day.

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

Run an interactive scenario at a command prompt.

```
/// <summary>
/// Hello AWS Support example.
/// </summary>
public static class SupportCaseScenario
{
    /*
    Before running this .NET code example, set up your development environment,
 including your credentials.
    To use the AWS Support API, you must have one of the following AWS Support
 plans: Business, Enterprise On-Ramp, or Enterprise.
```

```
   This .NET example performs the following tasks:
   1.  Get and display services. Select a service from the list.
   2.  Select a category from the selected service.
   3.  Get and display severity levels and select a severity level from the
list.
   4.  Create a support case using the selected service, category, and severity
level.
   5.  Get and display a list of open support cases for the current day.
   6.  Create an attachment set with a sample text file to add to the case.
   7.  Add a communication with the attachment to the support case.
   8.  List the communications of the support case.
   9.  Describe the attachment set.
   10. Resolve the support case.
   11. Get a list of resolved cases for the current day.
  */

   private static SupportWrapper _supportWrapper = null!;

   static async Task Main(string[] args)
   {
       // Set up dependency injection for the AWS Support service.
       // Use your AWS profile name, or leave it blank to use the default
profile.
       using var host = Host.CreateDefaultBuilder(args)
           .ConfigureLogging(logging =>
               logging.AddFilter("System", LogLevel.Debug)
                   .AddFilter<DebugLoggerProvider>("Microsoft",
LogLevel.Information)
                   .AddFilter<ConsoleLoggerProvider>("Microsoft",
LogLevel.Trace))
           .ConfigureServices((_, services) =>
               services.AddAWSService<IAmazonAWSSupport>(new AWSOptions()
{ Profile = "default" })
                   .AddTransient<SupportWrapper>()
           )
           .Build();

       var logger = LoggerFactory.Create(builder =>
       {
           builder.AddConsole();
       }).CreateLogger(typeof(SupportCaseScenario));

       _supportWrapper = host.Services.GetRequiredService<SupportWrapper>();
```

```
        Console.WriteLine(new string('-', 80));
        Console.WriteLine("Welcome to the AWS Support case example scenario.");
        Console.WriteLine(new string('-', 80));

        try
        {
            var apiSupported = await _supportWrapper.VerifySubscription();
            if (!apiSupported)
            {
                logger.LogError("You must have a Business, Enterprise On-Ramp, or
Enterprise Support " +
                                "plan to use the AWS Support API. \n\tPlease
upgrade your subscription to run these examples.");
                return;
            }

            var service = await DisplayAndSelectServices();

            var category = DisplayAndSelectCategories(service);

            var severityLevel = await DisplayAndSelectSeverity();

            var caseId = await CreateSupportCase(service, category,
severityLevel);

            await DescribeTodayOpenCases();

            var attachmentSetId = await CreateAttachmentSet();

            await AddCommunicationToCase(attachmentSetId, caseId);

            var attachmentId = await ListCommunicationsForCase(caseId);

            await DescribeCaseAttachment(attachmentId);

            await ResolveCase(caseId);

            await DescribeTodayResolvedCases();

            Console.WriteLine(new string('-', 80));
            Console.WriteLine("AWS Support case example scenario complete.");
            Console.WriteLine(new string('-', 80));
        }
```

```
        catch (Exception ex)
        {
            logger.LogError(ex, "There was a problem executing the scenario.");
        }
    }

    /// <summary>
    /// List some available services from AWS Support, and select a service for
the example.
    /// </summary>
    /// <returns>The selected service.</returns>
    private static async Task<Service> DisplayAndSelectServices()
    {
        Console.WriteLine(new string('-', 80));
        var services = await _supportWrapper.DescribeServices();
        Console.WriteLine($"AWS Support client returned {services.Count}
services.");

        Console.WriteLine($"1. Displaying first 10 services:");
        for (int i = 0; i < 10 && i < services.Count; i++)
        {
            Console.WriteLine($"\t{i + 1}. {services[i].Name}");
        }

        var choiceNumber = 0;
        while (choiceNumber < 1 || choiceNumber > services.Count)
        {
            Console.WriteLine(
                "Select an example support service by entering a number from the
preceding list:");
            var choice = Console.ReadLine();
            Int32.TryParse(choice, out choiceNumber);
        }
        Console.WriteLine(new string('-', 80));

        return services[choiceNumber - 1];
    }

    /// <summary>
    /// List the available categories for a service and select a category for the
example.
    /// </summary>
    /// <param name="service">Service to use for displaying categories.</param>
    /// <returns>The selected category.</returns>
```

```
    private static Category DisplayAndSelectCategories(Service service)
    {
        Console.WriteLine(new string('-', 80));

        Console.WriteLine($"2. Available support categories for Service
\"{service.Name}\":");
        for (int i = 0; i < service.Categories.Count; i++)
        {
            Console.WriteLine($"\t{i + 1}. {service.Categories[i].Name}");
        }

        var choiceNumber = 0;
        while (choiceNumber < 1 || choiceNumber > service.Categories.Count)
        {
            Console.WriteLine(
                "Select an example support category by entering a number from the
preceding list:");
            var choice = Console.ReadLine();
            Int32.TryParse(choice, out choiceNumber);
        }

        Console.WriteLine(new string('-', 80));

        return service.Categories[choiceNumber - 1];
    }

    /// <summary>
    /// List available severity levels from AWS Support, and select a level for
the example.
    /// </summary>
    /// <returns>The selected severity level.</returns>
    private static async Task<SeverityLevel> DisplayAndSelectSeverity()
    {
        Console.WriteLine(new string('-', 80));
        var severityLevels = await _supportWrapper.DescribeSeverityLevels();

        Console.WriteLine($"3. Get and display available severity levels:");
        for (int i = 0; i < 10 && i < severityLevels.Count; i++)
        {
            Console.WriteLine($"\t{i + 1}. {severityLevels[i].Name}");
        }

        var choiceNumber = 0;
        while (choiceNumber < 1 || choiceNumber > severityLevels.Count)
```

```
        {
            Console.WriteLine(
                "Select an example severity level by entering a number from the
preceding list:");
            var choice = Console.ReadLine();
            Int32.TryParse(choice, out choiceNumber);
        }
        Console.WriteLine(new string('-', 80));

        return severityLevels[choiceNumber - 1];
    }

    /// <summary>
    /// Create an example support case.
    /// </summary>
    /// <param name="service">Service to use for the new case.</param>
    /// <param name="category">Category to use for the new case.</param>
    /// <param name="severity">Severity to use for the new case.</param>
    /// <returns>The caseId of the new support case.</returns>
    private static async Task<string> CreateSupportCase(Service service,
        Category category, SeverityLevel severity)
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"4. Create an example support case" +
                          $" with the following settings:" +
                          $" \n\tService: {service.Name}, Category:
{category.Name} " +
                          $"and Severity Level: {severity.Name}.");
        var caseId = await _supportWrapper.CreateCase(service.Code,
category.Code, severity.Code,
            "Example case for testing, ignore.", "This is my example support
case.");

        Console.WriteLine($"\tNew case created with ID {caseId}");

        Console.WriteLine(new string('-', 80));

        return caseId;
    }

    /// <summary>
    /// List open cases for the current day.
    /// </summary>
    /// <returns>Async task.</returns>
```

```
    private static async Task DescribeTodayOpenCases()
    {
        Console.WriteLine($"5. List the open support cases for the current
day.");
        // Describe the cases. If it is empty, try again and allow time for the
new case to appear.
        List<CaseDetails> currentOpenCases = null!;
        while (currentOpenCases == null || currentOpenCases.Count == 0)
        {
            Thread.Sleep(1000);
            currentOpenCases = await _supportWrapper.DescribeCases(
                new List<string>(),
                null,
                false,
                false,
                DateTime.UtcNow.Date,
                DateTime.UtcNow);
        }

        foreach (var openCase in currentOpenCases)
        {
            Console.WriteLine($"\tCase: {openCase.CaseId} created
{openCase.TimeCreated}");
        }

        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
    /// Create an attachment set for a support case.
    /// </summary>
    /// <returns>The attachment set id.</returns>
    private static async Task<string> CreateAttachmentSet()
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"6. Create an attachment set for a support case.");
        var fileName = "example_attachment.txt";

        // Create the file if it does not already exist.
        if (!File.Exists(fileName))
        {
            await using StreamWriter sw = File.CreateText(fileName);
            await sw.WriteLineAsync(
                "This is a sample file for attachment to a support case.");
```

```
        }

        await using var ms = new MemoryStream(await
 File.ReadAllBytesAsync(fileName));

        var attachmentSetId = await _supportWrapper.AddAttachmentToSet(
            ms,
            fileName);

        Console.WriteLine($"\tNew attachment set created with id: \n
\t{attachmentSetId.Substring(0, 65)}...");

        Console.WriteLine(new string('-', 80));

        return attachmentSetId;
    }

    /// <summary>
    /// Add an attachment set and communication to a case.
    /// </summary>
    /// <param name="attachmentSetId">Id of the attachment set.</param>
    /// <param name="caseId">Id of the case to receive the attachment set.</
param>
    /// <returns>Async task.</returns>
    private static async Task AddCommunicationToCase(string attachmentSetId,
 string caseId)
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"7. Add attachment set and communication to
 {caseId}.");

        await _supportWrapper.AddCommunicationToCase(
            caseId,
            "This is an example communication added to a support case.",
            attachmentSetId);

        Console.WriteLine($"\tNew attachment set and communication added to
 {caseId}");

        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
    /// List the communications for a case.
```

```
    /// </summary>
    /// <param name="caseId">Id of the case to describe.</param>
    /// <returns>An attachment id.</returns>
    private static async Task<string> ListCommunicationsForCase(string caseId)
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"8. List communications for case {caseId}.");

        var communications = await
_supportWrapper.DescribeCommunications(caseId);
        var attachmentId = "";
        foreach (var communication in communications)
        {
            Console.WriteLine(
                $"\tCommunication created on: {communication.TimeCreated} has
{communication.AttachmentSet.Count} attachments.");
            if (communication.AttachmentSet.Any())
            {
                attachmentId = communication.AttachmentSet.First().AttachmentId;
            }
        }

        Console.WriteLine(new string('-', 80));
        return attachmentId;
    }

    /// <summary>
    /// Describe an attachment by id.
    /// </summary>
    /// <param name="attachmentId">Id of the attachment to describe.</param>
    /// <returns>Async task.</returns>
    private static async Task DescribeCaseAttachment(string attachmentId)
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"9. Describe the attachment set.");

        var attachment = await _supportWrapper.DescribeAttachment(attachmentId);
        var data = Encoding.ASCII.GetString(attachment.Data.ToArray());
        Console.WriteLine($"\tAttachment includes {attachment.FileName} with
data: \n\t{data}");

        Console.WriteLine(new string('-', 80));
    }
```

```
    /// <summary>
    /// Resolve the support case.
    /// </summary>
    /// <param name="caseId">Id of the case to resolve.</param>
    /// <returns>Async task.</returns>
    private static async Task ResolveCase(string caseId)
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"10. Resolve case {caseId}.");

        var status = await _supportWrapper.ResolveCase(caseId);
        Console.WriteLine($"\tCase {caseId} has final status {status}");

        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
    /// List resolved cases for the current day.
    /// </summary>
    /// <returns>Async Task.</returns>
    private static async Task DescribeTodayResolvedCases()
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"11. List the resolved support cases for the current
day.");
        var currentCases = await _supportWrapper.DescribeCases(
            new List<string>(),
            null,
            false,
            true,
            DateTime.UtcNow.Date,
            DateTime.UtcNow);

        foreach (var currentCase in currentCases)
        {
            if (currentCase.Status == "resolved")
            {
                Console.WriteLine(
                    $"\tCase: {currentCase.CaseId}: status
{currentCase.Status}");
            }
        }

        Console.WriteLine(new string('-', 80));
```

```
    }
}
```

Wrapper methods used by the scenario for Support actions.

```
/// <summary>
/// Wrapper methods to use AWS Support for working with support cases.
/// </summary>
public class SupportWrapper
{
    private readonly IAmazonAWSSupport _amazonSupport;
    public SupportWrapper(IAmazonAWSSupport amazonSupport)
    {
        _amazonSupport = amazonSupport;
    }


    /// <summary>
    /// Get the descriptions of AWS services.
    /// </summary>
    /// <param name="name">Optional language for services.
    /// Currently Chinese (“zh”), English ("en"), Japanese ("ja") and Korean
 (“ko”) are supported.</param>
    /// <returns>The list of AWS service descriptions.</returns>
    public async Task<List<Service>> DescribeServices(string language = "en")
    {
        var response = await _amazonSupport.DescribeServicesAsync(
            new DescribeServicesRequest()
            {
                Language = language
            });
        return response.Services;
    }



    /// <summary>
    /// Get the descriptions of support severity levels.
    /// </summary>
    /// <param name="name">Optional language for severity levels.
```

```
   /// Currently Chinese (“zh”), English ("en"), Japanese ("ja") and Korean
(“ko”) are supported.</param>
   /// <returns>The list of support severity levels.</returns>
   public async Task<List<SeverityLevel>> DescribeSeverityLevels(string language
= "en")
   {
       var response = await _amazonSupport.DescribeSeverityLevelsAsync(
           new DescribeSeverityLevelsRequest()
           {
               Language = language
           });
       return response.SeverityLevels;
   }



   /// <summary>
   /// Create a new support case.
   /// </summary>
   /// <param name="serviceCode">Service code for the new case.</param>
   /// <param name="categoryCode">Category for the new case.</param>
   /// <param name="severityCode">Severity code for the new case.</param>
   /// <param name="subject">Subject of the new case.</param>
   /// <param name="body">Body text of the new case.</param>
   /// <param name="language">Optional language support for your case.
   /// Currently Chinese (“zh”), English ("en"), Japanese ("ja") and Korean
(“ko”) are supported.</param>
   /// <param name="attachmentSetId">Optional Id for an attachment set for the
new case.</param>
   /// <param name="issueType">Optional issue type for the new case. Options are
"customer-service" or "technical".</param>
   /// <returns>The caseId of the new support case.</returns>
   public async Task<string> CreateCase(string serviceCode, string categoryCode,
string severityCode, string subject,
       string body, string language = "en", string? attachmentSetId = null,
string issueType = "customer-service")
   {
       var response = await _amazonSupport.CreateCaseAsync(
           new CreateCaseRequest()
           {
               ServiceCode = serviceCode,
               CategoryCode = categoryCode,
               SeverityCode = severityCode,
               Subject = subject,
```

```
                Language = language,
                AttachmentSetId = attachmentSetId,
                IssueType = issueType,
                CommunicationBody = body
            });
        return response.CaseId;
    }



    /// <summary>
    /// Add an attachment to a set, or create a new attachment set if one does
not exist.
    /// </summary>
    /// <param name="data">The data for the attachment.</param>
    /// <param name="fileName">The file name for the attachment.</param>
    /// <param name="attachmentSetId">Optional setId for the attachment. Creates
a new attachment set if empty.</param>
    /// <returns>The setId of the attachment.</returns>
    public async Task<string> AddAttachmentToSet(MemoryStream data, string
fileName, string? attachmentSetId = null)
    {
        var response = await _amazonSupport.AddAttachmentsToSetAsync(
            new AddAttachmentsToSetRequest
            {
                AttachmentSetId = attachmentSetId,
                Attachments = new List<Attachment>
                {
                    new Attachment
                    {
                        Data = data,
                        FileName = fileName
                    }
                }
            });
        return response.AttachmentSetId;
    }



    /// <summary>
    /// Get description of a specific attachment.
    /// </summary>
```

```
    /// <param name="attachmentId">Id of the attachment, usually fetched by
 describing the communications of a case.</param>
    /// <returns>The attachment object.</returns>
    public async Task<Attachment> DescribeAttachment(string attachmentId)
    {
        var response = await _amazonSupport.DescribeAttachmentAsync(
            new DescribeAttachmentRequest()
            {
                AttachmentId = attachmentId
            });
        return response.Attachment;
    }



    /// <summary>
    /// Add communication to a case, including optional attachment set ID and CC
 email addresses.
    /// </summary>
    /// <param name="caseId">Id for the support case.</param>
    /// <param name="body">Body text of the communication.</param>
    /// <param name="attachmentSetId">Optional Id for an attachment set.</param>
    /// <param name="ccEmailAddresses">Optional list of CC email addresses.</
param>
    /// <returns>True if successful.</returns>
    public async Task<bool> AddCommunicationToCase(string caseId, string body,
        string? attachmentSetId = null, List<string>? ccEmailAddresses = null)
    {
        var response = await _amazonSupport.AddCommunicationToCaseAsync(
            new AddCommunicationToCaseRequest()
            {
                CaseId = caseId,
                CommunicationBody = body,
                AttachmentSetId = attachmentSetId,
                CcEmailAddresses = ccEmailAddresses
            });
        return response.Result;
    }



    /// <summary>
    /// Describe the communications for a case, optionally with a date filter.
    /// </summary>
```

```
    /// <param name="caseId">The ID of the support case.</param>
    /// <param name="afterTime">The optional start date for a filtered search.</
param>
    /// <param name="beforeTime">The optional end date for a filtered search.</
param>
    /// <returns>The list of communications for the case.</returns>
    public async Task<List<Communication>> DescribeCommunications(string caseId,
 DateTime? afterTime = null, DateTime? beforeTime = null)
    {
        var results = new List<Communication>();
        var paginateCommunications =
 _amazonSupport.Paginators.DescribeCommunications(
            new DescribeCommunicationsRequest()
            {
                CaseId = caseId,
                AfterTime = afterTime?.ToString("s"),
                BeforeTime = beforeTime?.ToString("s")
            });
        // Get the entire list using the paginator.
        await foreach (var communications in
 paginateCommunications.Communications)
        {
            results.Add(communications);
        }
        return results;
    }



    /// <summary>
    /// Get case details for a list of case ids, optionally with date filters.
    /// </summary>
    /// <param name="caseIds">The list of case IDs.</param>
    /// <param name="displayId">Optional display ID.</param>
    /// <param name="includeCommunication">True to include communication.
 Defaults to true.</param>
    /// <param name="includeResolvedCases">True to include resolved cases.
 Defaults to false.</param>
    /// <param name="afterTime">The optional start date for a filtered search.</
param>
    /// <param name="beforeTime">The optional end date for a filtered search.</
param>
    /// <param name="language">Optional language support for your case.
```

```
    /// Currently Chinese (“zh”), English ("en"), Japanese ("ja") and Korean
(“ko”) are supported.</param>
    /// <returns>A list of CaseDetails.</returns>
    public async Task<List<CaseDetails>> DescribeCases(List<string> caseIds,
string? displayId = null, bool includeCommunication = true,
        bool includeResolvedCases = false, DateTime? afterTime = null, DateTime?
beforeTime = null,
        string language = "en")
    {
        var results = new List<CaseDetails>();
        var paginateCases = _amazonSupport.Paginators.DescribeCases(
            new DescribeCasesRequest()
            {
                CaseIdList = caseIds,
                DisplayId = displayId,
                IncludeCommunications = includeCommunication,
                IncludeResolvedCases = includeResolvedCases,
                AfterTime = afterTime?.ToString("s"),
                BeforeTime = beforeTime?.ToString("s"),
                Language = language
            });
        // Get the entire list using the paginator.
        await foreach (var cases in paginateCases.Cases)
        {
            results.Add(cases);
        }
        return results;
    }



    /// <summary>
    /// Resolve a support case by caseId.
    /// </summary>
    /// <param name="caseId">Id for the support case.</param>
    /// <returns>The final status of the case after resolving.</returns>
    public async Task<string> ResolveCase(string caseId)
    {
        var response = await _amazonSupport.ResolveCaseAsync(
            new ResolveCaseRequest()
            {
                CaseId = caseId
            });
        return response.FinalCaseStatus;
```

```
    }


    /// <summary>
    /// Verify the support level for AWS Support API access.
    /// </summary>
    /// <returns>True if the subscription level supports API access.</returns>
    public async Task<bool> VerifySubscription()
    {
        try
        {
            var response = await _amazonSupport.DescribeServicesAsync(
                new DescribeServicesRequest()
                {
                    Language = "en"
                });
            return response.HttpStatusCode == HttpStatusCode.OK;
        }
        catch (Amazon.AWSSupport.AmazonAWSSupportException ex)
        {
            if (ex.ErrorCode == "SubscriptionRequiredException")
            {
                return false;
            }
            else throw;
        }
    }
}
```

- For API details, see the following topics in *AWS SDK for .NET API Reference*.
  - AddAttachmentsToSet
  - AddCommunicationToCase
  - CreateCase
  - DescribeAttachment
  - DescribeCases
  - DescribeCommunications
  - DescribeServices
  - DescribeSeverityLevels
  - ResolveCase

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

Run various Support operations.

```
import software.amazon.awssdk.core.SdkBytes;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.support.SupportClient;
import software.amazon.awssdk.services.support.model.AddAttachmentsToSetResponse;
import
 software.amazon.awssdk.services.support.model.AddCommunicationToCaseRequest;
import
 software.amazon.awssdk.services.support.model.AddCommunicationToCaseResponse;
import software.amazon.awssdk.services.support.model.Attachment;
import software.amazon.awssdk.services.support.model.AttachmentDetails;
import software.amazon.awssdk.services.support.model.CaseDetails;
import software.amazon.awssdk.services.support.model.Category;
import software.amazon.awssdk.services.support.model.Communication;
import software.amazon.awssdk.services.support.model.CreateCaseRequest;
import software.amazon.awssdk.services.support.model.CreateCaseResponse;
import software.amazon.awssdk.services.support.model.DescribeAttachmentRequest;
import software.amazon.awssdk.services.support.model.DescribeAttachmentResponse;
import software.amazon.awssdk.services.support.model.DescribeCasesRequest;
import software.amazon.awssdk.services.support.model.DescribeCasesResponse;
import
 software.amazon.awssdk.services.support.model.DescribeCommunicationsRequest;
import
 software.amazon.awssdk.services.support.model.DescribeCommunicationsResponse;
import software.amazon.awssdk.services.support.model.DescribeServicesRequest;
import software.amazon.awssdk.services.support.model.DescribeServicesResponse;
import
 software.amazon.awssdk.services.support.model.DescribeSeverityLevelsRequest;
import
 software.amazon.awssdk.services.support.model.DescribeSeverityLevelsResponse;
import software.amazon.awssdk.services.support.model.ResolveCaseRequest;
import software.amazon.awssdk.services.support.model.ResolveCaseResponse;
```

```
import software.amazon.awssdk.services.support.model.Service;
import software.amazon.awssdk.services.support.model.SeverityLevel;
import software.amazon.awssdk.services.support.model.SupportException;
import software.amazon.awssdk.services.support.model.AddAttachmentsToSetRequest;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.InputStream;
import java.time.Instant;
import java.time.temporal.ChronoUnit;
import java.util.ArrayList;
import java.util.List;

/**
 * Before running this Java (v2) code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 *
 * In addition, you must have the AWS Business Support Plan to use the AWS
 * Support Java API. For more information, see:
 *
 * https://aws.amazon.com/premiumsupport/plans/
 *
 * This Java example performs the following tasks:
 *
 * 1. Gets and displays available services.
 * 2. Gets and displays severity levels.
 * 3. Creates a support case by using the selected service, category, and
 * severity level.
 * 4. Gets a list of open cases for the current day.
 * 5. Creates an attachment set with a generated file.
 * 6. Adds a communication with the attachment to the support case.
 * 7. Lists the communications of the support case.
 * 8. Describes the attachment set included with the communication.
 * 9. Resolves the support case.
 * 10. Gets a list of resolved cases for the current day.
 */
public class SupportScenario {
```

```
   public static final String DASHES = new String(new char[80]).replace("\0",
"-");

   public static void main(String[] args) {
       final String usage = """

               Usage:
                   <fileAttachment>Where:
                   fileAttachment - The file can be a simple saved .txt file to
use as an email attachment.\s
               """;

     //  if (args.length != 1) {
     //      System.out.println(usage);
     //      System.exit(1);
     //  }

       String fileAttachment = "C:\\AWS\\test.txt" ; //args[0];
       Region region = Region.US_WEST_2;
       SupportClient supportClient = SupportClient.builder()
               .region(region)
               .build();

       System.out.println(DASHES);
       System.out.println("***** Welcome to the AWS Support case example
scenario.");
       System.out.println(DASHES);

       System.out.println(DASHES);
       System.out.println("1. Get and display available services.");
       List<String> sevCatList = displayServices(supportClient);
       System.out.println(DASHES);

       System.out.println(DASHES);
       System.out.println("2. Get and display Support severity levels.");
       String sevLevel = displaySevLevels(supportClient);
       System.out.println(DASHES);

       System.out.println(DASHES);
       System.out.println("3. Create a support case using the selected service,
category, and severity level.");
       String caseId = createSupportCase(supportClient, sevCatList, sevLevel);
       if (caseId.compareTo("") == 0) {
           System.out.println("A support case was not successfully created!");
```

```
            System.exit(1);
        } else
            System.out.println("Support case " + caseId + " was successfully
created!");
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("4. Get open support cases.");
        getOpenCase(supportClient);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("5. Create an attachment set with a generated file to
add to the case.");
        String attachmentSetId = addAttachment(supportClient, fileAttachment);
        System.out.println("The Attachment Set id value is" + attachmentSetId);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("6. Add communication with the attachment to the
support case.");
        addAttachSupportCase(supportClient, caseId, attachmentSetId);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("7. List the communications of the support case.");
        String attachId = listCommunications(supportClient, caseId);
        System.out.println("The Attachment id value is" + attachId);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("8. Describe the attachment set included with the
communication.");
        describeAttachment(supportClient, attachId);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("9. Resolve the support case.");
        resolveSupportCase(supportClient, caseId);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("10. Get a list of resolved cases for the current
day.");
```

```
        getResolvedCase(supportClient);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("***** This Scenario has successfully completed");
        System.out.println(DASHES);
    }

    public static void getResolvedCase(SupportClient supportClient) {
        try {
            // Specify the start and end time.
            Instant now = Instant.now();
            java.time.LocalDate.now();
            Instant yesterday = now.minus(1, ChronoUnit.DAYS);

            DescribeCasesRequest describeCasesRequest =
DescribeCasesRequest.builder()
                    .maxResults(30)
                    .afterTime(yesterday.toString())
                    .beforeTime(now.toString())
                    .includeResolvedCases(true)
                    .build();

            DescribeCasesResponse response =
supportClient.describeCases(describeCasesRequest);
            List<CaseDetails> cases = response.cases();
            for (CaseDetails sinCase : cases) {
                if (sinCase.status().compareTo("resolved") == 0)
                    System.out.println("The case status is " + sinCase.status());
            }

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    public static void resolveSupportCase(SupportClient supportClient, String
caseId) {
        try {
            ResolveCaseRequest caseRequest = ResolveCaseRequest.builder()
                    .caseId(caseId)
                    .build();
```

```
            ResolveCaseResponse response =
supportClient.resolveCase(caseRequest);
            System.out.println("The status of case " + caseId + " is " +
response.finalCaseStatus());

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    public static void describeAttachment(SupportClient supportClient, String
attachId) {
        try {
            DescribeAttachmentRequest attachmentRequest =
DescribeAttachmentRequest.builder()
                    .attachmentId(attachId)
                    .build();

            DescribeAttachmentResponse response =
supportClient.describeAttachment(attachmentRequest);
            System.out.println("The name of the file is " +
response.attachment().fileName());

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    public static String listCommunications(SupportClient supportClient, String
caseId) {
        try {
            String attachId = null;
            DescribeCommunicationsRequest communicationsRequest =
DescribeCommunicationsRequest.builder()
                    .caseId(caseId)
                    .maxResults(10)
                    .build();

            DescribeCommunicationsResponse response =
supportClient.describeCommunications(communicationsRequest);
            List<Communication> communications = response.communications();
            for (Communication comm : communications) {
```

```
                System.out.println("the body is: " + comm.body());

                // Get the attachment id value.
                List<AttachmentDetails> attachments = comm.attachmentSet();
                for (AttachmentDetails detail : attachments) {
                    attachId = detail.attachmentId();
                }
            }
            return attachId;

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
        return "";
    }

    public static void addAttachSupportCase(SupportClient supportClient, String
caseId, String attachmentSetId) {
        try {
            AddCommunicationToCaseRequest caseRequest =
AddCommunicationToCaseRequest.builder()
                    .caseId(caseId)
                    .attachmentSetId(attachmentSetId)
                    .communicationBody("Please refer to attachment for details.")
                    .build();

            AddCommunicationToCaseResponse response =
supportClient.addCommunicationToCase(caseRequest);
            if (response.result())
                System.out.println("You have successfully added a communication
to an AWS Support case");
            else
                System.out.println("There was an error adding the communication
to an AWS Support case");

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    public static String addAttachment(SupportClient supportClient, String
fileAttachment) {
```

```
        try {
            File myFile = new File(fileAttachment);
            InputStream sourceStream = new FileInputStream(myFile);
            SdkBytes sourceBytes = SdkBytes.fromInputStream(sourceStream);

            Attachment attachment = Attachment.builder()
                    .fileName(myFile.getName())
                    .data(sourceBytes)
                    .build();

            AddAttachmentsToSetRequest setRequest =
AddAttachmentsToSetRequest.builder()
                    .attachments(attachment)
                    .build();

            AddAttachmentsToSetResponse response =
supportClient.addAttachmentsToSet(setRequest);
            return response.attachmentSetId();

        } catch (SupportException | FileNotFoundException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
        return "";
    }

    public static void getOpenCase(SupportClient supportClient) {
        try {
            // Specify the start and end time.
            Instant now = Instant.now();
            java.time.LocalDate.now();
            Instant yesterday = now.minus(1, ChronoUnit.DAYS);

            DescribeCasesRequest describeCasesRequest =
DescribeCasesRequest.builder()
                    .maxResults(20)
                    .afterTime(yesterday.toString())
                    .beforeTime(now.toString())
                    .build();

            DescribeCasesResponse response =
supportClient.describeCases(describeCasesRequest);
            List<CaseDetails> cases = response.cases();
            for (CaseDetails sinCase : cases) {
```

```
                System.out.println("The case status is " + sinCase.status());
                System.out.println("The case Id is " + sinCase.caseId());
                System.out.println("The case subject is " + sinCase.subject());
            }

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    public static String createSupportCase(SupportClient supportClient,
List<String> sevCatList, String sevLevel) {
        try {
            String serviceCode = sevCatList.get(0);
            String caseCat = sevCatList.get(1);
            CreateCaseRequest caseRequest = CreateCaseRequest.builder()
                    .categoryCode(caseCat.toLowerCase())
                    .serviceCode(serviceCode.toLowerCase())
                    .severityCode(sevLevel.toLowerCase())
                    .communicationBody("Test issue with " +
serviceCode.toLowerCase())
                    .subject("Test case, please ignore")
                    .language("en")
                    .issueType("technical")
                    .build();

            CreateCaseResponse response = supportClient.createCase(caseRequest);
            return response.caseId();

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
        return "";
    }

    public static String displaySevLevels(SupportClient supportClient) {
        try {
            DescribeSeverityLevelsRequest severityLevelsRequest =
DescribeSeverityLevelsRequest.builder()
                    .language("en")
                    .build();
```

```
            DescribeSeverityLevelsResponse response =
supportClient.describeSeverityLevels(severityLevelsRequest);
            List<SeverityLevel> severityLevels = response.severityLevels();
            String levelName = null;
            for (SeverityLevel sevLevel : severityLevels) {
                System.out.println("The severity level name is: " +
sevLevel.name());
                if (sevLevel.name().compareTo("High") == 0)
                    levelName = sevLevel.name();
            }
            return levelName;

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
        return "";
    }

    // Return a List that contains a Service name and Category name.
    public static List<String> displayServices(SupportClient supportClient) {
        try {
            DescribeServicesRequest servicesRequest =
DescribeServicesRequest.builder()
                    .language("en")
                    .build();

            DescribeServicesResponse response =
supportClient.describeServices(servicesRequest);
            String serviceCode = null;
            String catName = null;
            List<String> sevCatList = new ArrayList<>();
            List<Service> services = response.services();

            System.out.println("Get the first 10 services");
            int index = 1;
            for (Service service : services) {
                if (index == 11)
                    break;

                System.out.println("The Service name is: " + service.name());
                if (service.name().compareTo("Account") == 0)
                    serviceCode = service.code();
```

```
                // Get the Categories for this service.
                List<Category> categories = service.categories();
                for (Category cat : categories) {
                    System.out.println("The category name is: " + cat.name());
                    if (cat.name().compareTo("Security") == 0)
                        catName = cat.name();
                }
                index++;
            }

            // Push the two values to the list.
            sevCatList.add(serviceCode);
            sevCatList.add(catName);
            return sevCatList;

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
        return null;
    }
}
```

- For API details, see the following topics in *AWS SDK for Java 2.x API Reference*.
  - AddAttachmentsToSet
  - AddCommunicationToCase
  - CreateCase
  - DescribeAttachment
  - DescribeCases
  - DescribeCommunications
  - DescribeServices
  - DescribeSeverityLevels
  - ResolveCase

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

Run an interactive scenario in the terminal.

```
import {
  AddAttachmentsToSetCommand,
  AddCommunicationToCaseCommand,
  CreateCaseCommand,
  DescribeAttachmentCommand,
  DescribeCasesCommand,
  DescribeCommunicationsCommand,
  DescribeServicesCommand,
  DescribeSeverityLevelsCommand,
  ResolveCaseCommand,
  SupportClient,
} from "@aws-sdk/client-support";
import * as inquirer from "@inquirer/prompts";
import { retry } from "@aws-doc-sdk-examples/lib/utils/util-timers.js";

const wrapText = (text, char = "=") => {
  const rule = char.repeat(80);
  return `${rule}\n    ${text}\n${rule}\n`;
};

const client = new SupportClient({ region: "us-east-1" });

// Verify that the account has a Support plan.
export const verifyAccount = async () => {
  const command = new DescribeServicesCommand({});

  try {
    await client.send(command);
  } catch (err) {
    if (err.name === "SubscriptionRequiredException") {
      throw new Error(
```

```
        "You must be subscribed to the AWS Support plan to use this feature.",
      );
    }
    throw err;
  }
};

/**
 * Select a service from the list returned from DescribeServices.
 */
export const getService = async () => {
  const { services } = await client.send(new DescribeServicesCommand({}));
  const selectedService = await inquirer.select({
    message:
      "Select a service. Your support case will be created for this service. The
 list of services is truncated for readability.",
    choices: services.slice(0, 10).map((s) => ({ name: s.name, value: s })),
  });
  return selectedService;
};

/**
 * @param {{ categories: import('@aws-sdk/client-support').Category[]}} service
 */
export const getCategory = async (service) => {
  const selectedCategory = await inquirer.select({
    message: "Select a category.",
    choices: service.categories.map((c) => ({ name: c.name, value: c })),
  });
  return selectedCategory;
};

// Get the available severity levels for the account.
export const getSeverityLevel = async () => {
  const command = new DescribeSeverityLevelsCommand({});
  const { severityLevels } = await client.send(command);
  const selectedSeverityLevel = await inquirer.select({
    message: "Select a severity level.",
    choices: severityLevels.map((s) => ({ name: s.name, value: s })),
  });
  return selectedSeverityLevel;
};

/**
```

```
 * Create a new support case
 * @param {{
 *  selectedService: import('@aws-sdk/client-support').Service
 *  selectedCategory: import('@aws-sdk/client-support').Category
 *  selectedSeverityLevel: import('@aws-sdk/client-support').SeverityLevel
 * }} selections
 * @returns
 */
export const createCase = async ({
  selectedService,
  selectedCategory,
  selectedSeverityLevel,
}) => {
  const command = new CreateCaseCommand({
    subject: "IGNORE: Test case",
    communicationBody: "This is a test. Please ignore.",
    serviceCode: selectedService.code,
    categoryCode: selectedCategory.code,
    severityCode: selectedSeverityLevel.code,
  });
  const { caseId } = await client.send(command);
  return caseId;
};

// Get a list of open support cases created today.
export const getTodaysOpenCases = async () => {
  const d = new Date();
  const startOfToday = new Date(d.getFullYear(), d.getMonth(), d.getDate());
  const command = new DescribeCasesCommand({
    includeCommunications: false,
    afterTime: startOfToday.toISOString(),
  });

  const { cases } = await client.send(command);

  if (cases.length === 0) {
    throw new Error(
      "Unexpected number of cases. Expected more than 0 open cases.",
    );
  }
  return cases;
};

// Create an attachment set.
```

```
export const createAttachmentSet = async () => {
  const command = new AddAttachmentsToSetCommand({
    attachments: [
      {
        fileName: "example.txt",
        data: new TextEncoder().encode("some example text"),
      },
    ],
  });
  const { attachmentSetId } = await client.send(command);
  return attachmentSetId;
};

export const linkAttachmentSetToCase = async (attachmentSetId, caseId) => {
  const command = new AddCommunicationToCaseCommand({
    attachmentSetId,
    caseId,
    communicationBody: "Adding attachment set to case.",
  });
  await client.send(command);
};

// Get all communications for a support case.
export const getCommunications = async (caseId) => {
  const command = new DescribeCommunicationsCommand({
    caseId,
  });
  const { communications } = await client.send(command);
  return communications;
};

/**
 * @param {import('@aws-sdk/client-support').Communication[]} communications
 */
export const getFirstAttachment = (communications) => {
  const firstCommWithAttachment = communications.find(
    (c) => c.attachmentSet.length > 0,
  );
  return firstCommWithAttachment?.attachmentSet[0].attachmentId;
};

// Get an attachment.
export const getAttachment = async (attachmentId) => {
  const command = new DescribeAttachmentCommand({
```

```
    attachmentId,
  });
  const { attachment } = await client.send(command);
  return attachment;
};

// Resolve the case matching the given case ID.
export const resolveCase = async (caseId) => {
  const shouldResolve = await inquirer.confirm({
    message: `Do you want to resolve ${caseId}?`,
  });

  if (shouldResolve) {
    const command = new ResolveCaseCommand({
      caseId: caseId,
    });

    await client.send(command);
    return true;
  }
  return false;
};

/**
 * Find a specific case in the list of provided cases by case ID.
 * If the case is not found, and the results are paginated, continue
 * paging through the results.
 * @param {{
 *   caseId: string,
 *   cases: import('@aws-sdk/client-support').CaseDetails[]
 *   nextToken: string
 * }} options
 * @returns
 */
export const findCase = async ({ caseId, cases, nextToken }) => {
  const foundCase = cases.find((c) => c.caseId === caseId);

  if (foundCase) {
    return foundCase;
  }

  if (nextToken) {
    const response = await client.send(
      new DescribeCasesCommand({
```

```
        nextToken,
        includeResolvedCases: true,
      }),
    );
    return findCase({
      caseId,
      cases: response.cases,
      nextToken: response.nextToken,
    });
  }

  throw new Error(`${caseId} not found.`);
};

// Get all cases created today.
export const getTodaysResolvedCases = async (caseIdToWaitFor) => {
  const d = new Date("2023-01-18");
  const startOfToday = new Date(d.getFullYear(), d.getMonth(), d.getDate());
  const command = new DescribeCasesCommand({
    includeCommunications: false,
    afterTime: startOfToday.toISOString(),
    includeResolvedCases: true,
  });
  const { cases, nextToken } = await client.send(command);
  await findCase({ cases, caseId: caseIdToWaitFor, nextToken });
  return cases.filter((c) => c.status === "resolved");
};

const main = async () => {
  let caseId;
  try {
    console.log(wrapText("Welcome to the AWS Support basic usage scenario."));

    // Verify that the account is subscribed to support.
    await verifyAccount();

    // Provided a truncated list of services and prompt the user to select one.
    const selectedService = await getService();

    // Provided the categories for the selected service and prompt the user to
 select one.
    const selectedCategory = await getCategory(selectedService);
```

```
   // Provide the severity available severity levels for the account and prompt
the user to select one.
   const selectedSeverityLevel = await getSeverityLevel();

   // Create a support case.
   console.log("\nCreating a support case.");
   caseId = await createCase({
     selectedService,
     selectedCategory,
     selectedSeverityLevel,
   });
   console.log(`Support case created: ${caseId}`);

   // Display a list of open support cases created today.
   const todaysOpenCases = await retry(
     { intervalInMs: 1000, maxRetries: 15 },
     getTodaysOpenCases,
   );
   console.log(
     `\nOpen support cases created today: ${todaysOpenCases.length}`,
   );
   console.log(todaysOpenCases.map((c) => `${c.caseId}`).join("\n"));

   // Create an attachment set.
   console.log("\nCreating an attachment set.");
   const attachmentSetId = await createAttachmentSet();
   console.log(`Attachment set created: ${attachmentSetId}`);

   // Add the attachment set to the support case.
   console.log(`\nAdding attachment set to ${caseId}`);
   await linkAttachmentSetToCase(attachmentSetId, caseId);
   console.log(`Attachment set added to ${caseId}`);

   // List the communications for a support case.
   console.log(`\nListing communications for ${caseId}`);
   const communications = await getCommunications(caseId);
   console.log(
     communications
       .map(
         (c) =>
           `Communication created on ${c.timeCreated}. Has
${c.attachmentSet.length} attachments.`,
       )
       .join("\n"),
```

```
    );

    // Describe the first attachment.
    console.log(`\nDescribing attachment ${attachmentSetId}`);
    const attachmentId = getFirstAttachment(communications);
    const attachment = await getAttachment(attachmentId);
    console.log(
      `Attachment is the file '${
        attachment.fileName
      }' with data: \n${new TextDecoder().decode(attachment.data)}`,
    );

    // Confirm that the support case should be resolved.
    const isResolved = await resolveCase(caseId);
    if (isResolved) {
      // List the resolved cases and include the one previously created.
      // Resolved cases can take a while to appear.
      console.log(
        "\nWaiting for case status to be marked as resolved. This can take some
 time.",
      );
      const resolvedCases = await retry(
        { intervalInMs: 20000, maxRetries: 15 },
        () => getTodaysResolvedCases(caseId),
      );
      console.log("Resolved cases:");
      console.log(resolvedCases.map((c) => c.caseId).join("\n"));
    }
  } catch (err) {
    console.error(err);
  }
};
```

- For API details, see the following topics in *AWS SDK for JavaScript API Reference*.
  - AddAttachmentsToSet
  - AddCommunicationToCase
  - CreateCase
  - DescribeAttachment
  - DescribeCases
  - DescribeCommunications

- DescribeServices
- DescribeSeverityLevels
- ResolveCase

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/**
Before running this Kotlin code example, set up your development environment,
including your credentials.

For more information, see the following documentation topic:

https://docs.aws.amazon.com/sdk-for-kotlin/latest/developer-guide/setup.html
In addition, you must have the AWS Business Support Plan to use the AWS Support
 Java API. For more information, see:

https://aws.amazon.com/premiumsupport/plans/

This Kotlin example performs the following tasks:
1. Gets and displays available services.
2. Gets and displays severity levels.
3. Creates a support case by using the selected service, category, and severity
 level.
4. Gets a list of open cases for the current day.
5. Creates an attachment set with a generated file.
6. Adds a communication with the attachment to the support case.
7. Lists the communications of the support case.
8. Describes the attachment set included with the communication.
9. Resolves the support case.
10. Gets a list of resolved cases for the current day.
*/
```

```
suspend fun main(args: Array<String>) {
    val usage = """
    Usage:
        <fileAttachment>
    Where:
         fileAttachment - The file can be a simple saved .txt file to use as an
 email attachment.
    """

    if (args.size != 1) {
        println(usage)
        exitProcess(0)
    }

    val fileAttachment = args[0]
    println("***** Welcome to the AWS Support case example scenario.")
    println("***** Step 1. Get and display available services.")
    val sevCatList = displayServices()

    println("***** Step 2. Get and display Support severity levels.")
    val sevLevel = displaySevLevels()

    println("***** Step 3. Create a support case using the selected service,
 category, and severity level.")
    val caseIdVal = createSupportCase(sevCatList, sevLevel)
    if (caseIdVal != null) {
        println("Support case $caseIdVal was successfully created!")
    } else {
        println("A support case was not successfully created!")
        exitProcess(1)
    }

    println("***** Step 4. Get open support cases.")
    getOpenCase()

    println("***** Step 5. Create an attachment set with a generated file to add
 to the case.")
    val attachmentSetId = addAttachment(fileAttachment)
    println("The Attachment Set id value is $attachmentSetId")

    println("***** Step 6. Add communication with the attachment to the support
 case.")
    addAttachSupportCase(caseIdVal, attachmentSetId)
```

```
    println("***** Step 7. List the communications of the support case.")
    val attachId = listCommunications(caseIdVal)
    println("The Attachment id value is $attachId")

    println("***** Step 8. Describe the attachment set included with the
 communication.")
    describeAttachment(attachId)

    println("***** Step 9. Resolve the support case.")
    resolveSupportCase(caseIdVal)

    println("***** Step 10. Get a list of resolved cases for the current day.")
    getResolvedCase()
    println("***** This Scenario has successfully completed")
}

suspend fun getResolvedCase() {
    // Specify the start and end time.
    val now = Instant.now()
    LocalDate.now()
    val yesterday = now.minus(1, ChronoUnit.DAYS)
    val describeCasesRequest =
        DescribeCasesRequest {
            maxResults = 30
            afterTime = yesterday.toString()
            beforeTime = now.toString()
            includeResolvedCases = true
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.describeCases(describeCasesRequest)
        response.cases?.forEach { sinCase ->
            println("The case status is ${sinCase.status}")
            println("The case Id is ${sinCase.caseId}")
            println("The case subject is ${sinCase.subject}")
        }
    }
}

suspend fun resolveSupportCase(caseIdVal: String) {
    val caseRequest =
        ResolveCaseRequest {
            caseId = caseIdVal
        }
```

```
    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.resolveCase(caseRequest)
        println("The status of case $caseIdVal is ${response.finalCaseStatus}")
    }
}

suspend fun describeAttachment(attachId: String?) {
    val attachmentRequest =
        DescribeAttachmentRequest {
            attachmentId = attachId
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.describeAttachment(attachmentRequest)
        println("The name of the file is ${response.attachment?.fileName}")
    }
}

suspend fun listCommunications(caseIdVal: String?): String? {
    val communicationsRequest =
        DescribeCommunicationsRequest {
            caseId = caseIdVal
            maxResults = 10
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response =
 supportClient.describeCommunications(communicationsRequest)
        response.communications?.forEach { comm ->
            println("the body is: " + comm.body)
            comm.attachmentSet?.forEach { detail ->
                return detail.attachmentId
            }
        }
    }
    return ""
}

suspend fun addAttachSupportCase(
    caseIdVal: String?,
    attachmentSetIdVal: String?,
) {
    val caseRequest =
        AddCommunicationToCaseRequest {
```

```
            caseId = caseIdVal
            attachmentSetId = attachmentSetIdVal
            communicationBody = "Please refer to attachment for details."
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.addCommunicationToCase(caseRequest)
        if (response.result) {
            println("You have successfully added a communication to an AWS
 Support case")
        } else {
            println("There was an error adding the communication to an AWS
 Support case")
        }
    }
}

suspend fun addAttachment(fileAttachment: String): String? {
    val myFile = File(fileAttachment)
    val sourceBytes = (File(fileAttachment).readBytes())
    val attachmentVal =
        Attachment {
            fileName = myFile.name
            data = sourceBytes
        }

    val setRequest =
        AddAttachmentsToSetRequest {
            attachments = listOf(attachmentVal)
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.addAttachmentsToSet(setRequest)
        return response.attachmentSetId
    }
}

suspend fun getOpenCase() {
    // Specify the start and end time.
    val now = Instant.now()
    LocalDate.now()
    val yesterday = now.minus(1, ChronoUnit.DAYS)
    val describeCasesRequest =
        DescribeCasesRequest {
```

```
            maxResults = 20
            afterTime = yesterday.toString()
            beforeTime = now.toString()
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.describeCases(describeCasesRequest)
        response.cases?.forEach { sinCase ->
            println("The case status is ${sinCase.status}")
            println("The case Id is ${sinCase.caseId}")
            println("The case subject is ${sinCase.subject}")
        }
    }
}

suspend fun createSupportCase(
    sevCatListVal: List<String>,
    sevLevelVal: String,
): String? {
    val serCode = sevCatListVal[0]
    val caseCategory = sevCatListVal[1]
    val caseRequest =
        CreateCaseRequest {
            categoryCode = caseCategory.lowercase(Locale.getDefault())
            serviceCode = serCode.lowercase(Locale.getDefault())
            severityCode = sevLevelVal.lowercase(Locale.getDefault())
            communicationBody = "Test issue with
 ${serCode.lowercase(Locale.getDefault())}"
            subject = "Test case, please ignore"
            language = "en"
            issueType = "technical"
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.createCase(caseRequest)
        return response.caseId
    }
}

suspend fun displaySevLevels(): String {
    var levelName = ""
    val severityLevelsRequest =
        DescribeSeverityLevelsRequest {
            language = "en"
```

```
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response =
 supportClient.describeSeverityLevels(severityLevelsRequest)
        response.severityLevels?.forEach { sevLevel ->
            println("The severity level name is: ${sevLevel.name}")
            if (sevLevel.name == "High") {
                levelName = sevLevel.name!!
            }
        }
        return levelName
    }
}

// Return a List that contains a Service name and Category name.
suspend fun displayServices(): List<String> {
    var serviceCode = ""
    var catName = ""
    val sevCatList = mutableListOf<String>()
    val servicesRequest =
        DescribeServicesRequest {
            language = "en"
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.describeServices(servicesRequest)
        println("Get the first 10 services")
        var index = 1

        response.services?.forEach { service ->
            if (index == 11) {
                return@forEach
            }

            println("The Service name is ${service.name}")
            if (service.name == "Account") {
                serviceCode = service.code.toString()
            }

            // Get the categories for this service.
            service.categories?.forEach { cat ->
                println("The category name is ${cat.name}")
                if (cat.name == "Security") {
```

```
                    catName = cat.name!!
                }
            }
            index++
        }
    }

    // Push the two values to the list.
    serviceCode.let { sevCatList.add(it) }
    catName.let { sevCatList.add(it) }
    return sevCatList
}
```

- For API details, see the following topics in *AWS SDK for Kotlin API reference*.
  - AddAttachmentsToSet
  - AddCommunicationToCase
  - CreateCase
  - DescribeAttachment
  - DescribeCases
  - DescribeCommunications
  - DescribeServices
  - DescribeSeverityLevels
  - ResolveCase

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

Run an interactive scenario at a command prompt.

```
class SupportCasesScenario:
```

```
    """Runs an interactive scenario that shows how to get started using AWS
Support."""

    def __init__(self, support_wrapper):
        """
        :param support_wrapper: An object that wraps AWS Support actions.
        """
        self.support_wrapper = support_wrapper

    def display_and_select_service(self):
        """
        Lists support services and prompts the user to select one.

        :return: The support service selected by the user.
        """
        print("-" * 88)
        services_list = self.support_wrapper.describe_services("en")
        print(f"AWS Support client returned {len(services_list)} services.")
        print("Displaying first 10 services:")

        service_choices = [svc["name"] for svc in services_list[:10]]
        selected_index = q.choose(
            "Select an example support service by entering a number from the
preceding list:",
            service_choices,
        )
        selected_service = services_list[selected_index]
        print("-" * 88)
        return selected_service

    def display_and_select_category(self, service):
        """
        Lists categories for a support service and prompts the user to select
one.

        :param service: The service of the categories.
        :return: The selected category.
        """
        print("-" * 88)
        print(
            f"Available support categories for Service {service['name']}
{len(service['categories'])}:"
        )
```

```
        categories_choices = [category["name"] for category in
service["categories"]]
        selected_index = q.choose(
            "Select an example support category by entering a number from the
preceding list:",
            categories_choices,
        )
        selected_category = service["categories"][selected_index]
        print("-" * 88)
        return selected_category

    def display_and_select_severity(self):
        """
        Lists available severity levels and prompts the user to select one.

        :return: The selected severity level.
        """
        print("-" * 88)
        severity_levels_list =
self.support_wrapper.describe_severity_levels("en")
        print(f"Available severity levels:")
        severity_choices = [level["name"] for level in severity_levels_list]
        selected_index = q.choose(
            "Select an example severity level by entering a number from the
preceding list:",
            severity_choices,
        )
        selected_severity = severity_levels_list[selected_index]
        print("-" * 88)
        return selected_severity

    def create_example_case(self, service, category, severity_level):
        """
        Creates an example support case with the user's selections.

        :param service: The service for the new case.
        :param category: The category for the new case.
        :param severity_level: The severity level for the new case.
        :return: The caseId of the new support case.
        """
        print("-" * 88)
        print(f"Creating new case for service {service['name']}.")
        case_id = self.support_wrapper.create_case(service, category,
severity_level)
```

```
        print(f"\tNew case created with ID {case_id}.")
        print("-" * 88)
        return case_id

    def list_open_cases(self):
        """
        List the open cases for the current day.
        """
        print("-" * 88)
        print("Let's list the open cases for the current day.")
        start_time = str(datetime.utcnow().date())
        end_time = str(datetime.utcnow().date() + timedelta(days=1))
        open_cases = self.support_wrapper.describe_cases(start_time, end_time,
False)
        for case in open_cases:
            print(f"\tCase: {case['caseId']}: status {case['status']}.")
        print("-" * 88)

    def create_attachment_set(self):
        """
        Create an attachment set with a sample file.

        :return: The attachment set ID of the new attachment set.
        """
        print("-" * 88)
        print("Creating attachment set with a sample file.")
        attachment_set_id = self.support_wrapper.add_attachment_to_set()
        print(f"\tNew attachment set created with ID {attachment_set_id}.")
        print("-" * 88)
        return attachment_set_id

    def add_communication(self, case_id, attachment_set_id):
        """
        Add a communication with an attachment set to the case.

        :param case_id: The ID of the case for the communication.
        :param attachment_set_id: The ID of the attachment set to
        add to the communication.
        """
        print("-" * 88)
        print(f"Adding a communication and attachment set to the case.")
        self.support_wrapper.add_communication_to_case(attachment_set_id,
case_id)
        print(
```

```
            f"Added a communication and attachment set {attachment_set_id} to the
case {case_id}."
        )
        print("-" * 88)

    def list_communications(self, case_id):
        """
        List the communications associated with a case.

        :param case_id: The ID of the case.
        :return: The attachment ID of an attachment.
        """
        print("-" * 88)
        print("Let's list the communications for our case.")
        attachment_id = ""
        communications =
self.support_wrapper.describe_all_case_communications(case_id)
        for communication in communications:
            print(
                f"\tCommunication created on {communication['timeCreated']} "
                f"has {len(communication['attachmentSet'])} attachments."
            )
            if len(communication["attachmentSet"]) > 0:
                attachment_id = communication["attachmentSet"][0]["attachmentId"]
        print("-" * 88)
        return attachment_id

    def describe_case_attachment(self, attachment_id):
        """
        Describe an attachment associated with a case.

        :param attachment_id: The ID of the attachment.
        """
        print("-" * 88)
        print("Let's list the communications for our case.")
        attached_file = self.support_wrapper.describe_attachment(attachment_id)
        print(f"\tAttachment includes file {attached_file}.")
        print("-" * 88)

    def resolve_case(self, case_id):
        """
        Shows how to resolve an AWS Support case by its ID.

        :param case_id: The ID of the case to resolve.
```

```
        """
        print("-" * 88)
        print(f"Resolving case with ID {case_id}.")
        case_status = self.support_wrapper.resolve_case(case_id)
        print(f"\tFinal case status is {case_status}.")
        print("-" * 88)

    def list_resolved_cases(self):
        """
        List the resolved cases for the current day.
        """
        print("-" * 88)
        print("Let's list the resolved cases for the current day.")
        start_time = str(datetime.utcnow().date())
        end_time = str(datetime.utcnow().date() + timedelta(days=1))
        resolved_cases = self.support_wrapper.describe_cases(start_time,
end_time, True)
        for case in resolved_cases:
            print(f"\tCase: {case['caseId']}: status {case['status']}.")
        print("-" * 88)

    def run_scenario(self):
        logging.basicConfig(level=logging.INFO, format="%(levelname)s:
%(message)s")

        print("-" * 88)
        print("Welcome to the AWS Support get started with support cases demo.")
        print("-" * 88)

        selected_service = self.display_and_select_service()
        selected_category = self.display_and_select_category(selected_service)
        selected_severity = self.display_and_select_severity()
        new_case_id = self.create_example_case(
            selected_service, selected_category, selected_severity
        )
        wait(10)
        self.list_open_cases()
        new_attachment_set_id = self.create_attachment_set()
        self.add_communication(new_case_id, new_attachment_set_id)
        new_attachment_id = self.list_communications(new_case_id)
        self.describe_case_attachment(new_attachment_id)
        self.resolve_case(new_case_id)
        wait(10)
        self.list_resolved_cases()
```

```
        print("\nThanks for watching!")
        print("-" * 88)


if __name__ == "__main__":
    try:
        scenario = SupportCasesScenario(SupportWrapper.from_client())
        scenario.run_scenario()
    except Exception:
        logging.exception("Something went wrong with the demo.")
```

Define a class that wraps support client actions.

```
class SupportWrapper:
    """Encapsulates Support actions."""

    def __init__(self, support_client):
        """
        :param support_client: A Boto3 Support client.
        """
        self.support_client = support_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        support_client = boto3.client("support")
        return cls(support_client)


    def describe_services(self, language):
        """
        Get the descriptions of AWS services available for support for a
 language.

        :param language: The language for support services.
        Currently, only "en" (English) and "ja" (Japanese) are supported.
        :return: The list of AWS service descriptions.
        """
        try:
```

```
            response = self.support_client.describe_services(language=language)
            services = response["services"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't get Support services for language %s. Here's why:
%s: %s",
                    language,
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return services


    def describe_severity_levels(self, language):
        """
        Get the descriptions of available severity levels for support cases for a
language.

        :param language: The language for support severity levels.
        Currently, only "en" (English) and "ja" (Japanese) are supported.
        :return: The list of severity levels.
        """
        try:
            response =
self.support_client.describe_severity_levels(language=language)
            severity_levels = response["severityLevels"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
```

```
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't get severity levels for language %s. Here's why:
%s: %s",
                    language,
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return severity_levels


    def create_case(self, service, category, severity):
        """
        Create a new support case.

        :param service: The service to use for the new case.
        :param category: The category to use for the new case.
        :param severity: The severity to use for the new case.
        :return: The caseId of the new case.
        """
        try:
            response = self.support_client.create_case(
                subject="Example case for testing, ignore.",
                serviceCode=service["code"],
                severityCode=severity["code"],
                categoryCode=category["code"],
                communicationBody="Example support case body.",
                language="en",
                issueType="customer-service",
            )
            case_id = response["caseId"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
```

```
            else:
                logger.error(
                    "Couldn't create case. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return case_id


    def add_attachment_to_set(self):
        """
        Add an attachment to a set, or create a new attachment set if one does
not exist.

        :return: The attachment set ID.
        """
        try:
            response = self.support_client.add_attachments_to_set(
                attachments=[
                    {
                        "fileName": "attachment_file.txt",
                        "data": b"This is a sample file for attachment to a
support case.",
                    }
                ]
            )
            new_set_id = response["attachmentSetId"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't add attachment. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
```

```
                raise
        else:
            return new_set_id


    def add_communication_to_case(self, attachment_set_id, case_id):
        """
        Add a communication and an attachment set to a case.

        :param attachment_set_id: The ID of an existing attachment set.
        :param case_id: The ID of the case.
        """
        try:
            self.support_client.add_communication_to_case(
                caseId=case_id,
                communicationBody="This is an example communication added to a
support case.",
                attachmentSetId=attachment_set_id,
            )
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't add communication. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise


    def describe_all_case_communications(self, case_id):
        """
        Describe all the communications for a case using a paginator.

        :param case_id: The ID of the case.
        :return: The communications for the case.
        """
```

```
        try:
            communications = []
            paginator =
self.support_client.get_paginator("describe_communications")
            for page in paginator.paginate(caseId=case_id):
                communications += page["communications"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't describe communications. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return communications


    def describe_attachment(self, attachment_id):
        """
        Get information about an attachment by its attachmentID.

        :param attachment_id: The ID of the attachment.
        :return: The name of the attached file.
        """
        try:
            response = self.support_client.describe_attachment(
                attachmentId=attachment_id
            )
            attached_file = response["attachment"]["fileName"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
```

```
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't get attachment description. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return attached_file


    def resolve_case(self, case_id):
        """
        Resolve a support case by its caseId.

        :param case_id: The ID of the case to resolve.
        :return: The final status of the case.
        """
        try:
            response = self.support_client.resolve_case(caseId=case_id)
            final_status = response["finalCaseStatus"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't resolve case. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return final_status
```

```
    def describe_cases(self, after_time, before_time, resolved):
        """
        Describe support cases over a period of time, optionally filtering
        by status.

        :param after_time: The start time to include for cases.
        :param before_time: The end time to include for cases.
        :param resolved: True to include resolved cases in the results,
            otherwise results are open cases.
        :return: The final status of the case.
        """
        try:
            cases = []
            paginator = self.support_client.get_paginator("describe_cases")
            for page in paginator.paginate(
                afterTime=after_time,
                beforeTime=before_time,
                includeResolvedCases=resolved,
                language="en",
            ):
                cases += page["cases"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't describe cases. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            if resolved:
                cases = filter(lambda case: case["status"] == "resolved", cases)
            return cases
```

- For API details, see the following topics in *AWS SDK for Python (Boto3) API Reference*.
  - AddAttachmentsToSet
  - AddCommunicationToCase
  - CreateCase
  - DescribeAttachment
  - DescribeCases
  - DescribeCommunications
  - DescribeServices
  - DescribeSeverityLevels
  - ResolveCase

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Actions for Support using AWS SDKs

The following code examples demonstrate how to perform individual Support actions with AWS SDKs. Each example includes a link to GitHub, where you can find instructions for setting up and running the code.

The following examples include only the most commonly used actions. For a complete list, see the AWS Support API Reference.

Examples

- Use AddAttachmentsToSet with an AWS SDK or CLI
- Use AddCommunicationToCase with an AWS SDK or CLI
- Use CreateCase with an AWS SDK or CLI
- Use DescribeAttachment with an AWS SDK or CLI
- Use DescribeCases with an AWS SDK or CLI
- Use DescribeCommunications with an AWS SDK or CLI
- Use DescribeServices with an AWS SDK or CLI

- Use DescribeSeverityLevels with an AWS SDK or CLI
- Use DescribeTrustedAdvisorCheckRefreshStatuses with a CLI
- Use DescribeTrustedAdvisorCheckResult with a CLI
- Use DescribeTrustedAdvisorCheckSummaries with a CLI
- Use DescribeTrustedAdvisorChecks with a CLI
- Use RefreshTrustedAdvisorCheck with a CLI
- Use ResolveCase with an AWS SDK or CLI

Use `AddAttachmentsToSet` with an AWS SDK or CLI

The following code examples show how to use `AddAttachmentsToSet`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// Add an attachment to a set, or create a new attachment set if one does
not exist.
    /// </summary>
    /// <param name="data">The data for the attachment.</param>
    /// <param name="fileName">The file name for the attachment.</param>
    /// <param name="attachmentSetId">Optional setId for the attachment. Creates
a new attachment set if empty.</param>
    /// <returns>The setId of the attachment.</returns>
```

```
    public async Task<string> AddAttachmentToSet(MemoryStream data, string
 fileName, string? attachmentSetId = null)
    {
        var response = await _amazonSupport.AddAttachmentsToSetAsync(
            new AddAttachmentsToSetRequest
            {
                AttachmentSetId = attachmentSetId,
                Attachments = new List<Attachment>
                {
                    new Attachment
                    {
                        Data = data,
                        FileName = fileName
                    }
                }
            });
        return response.AttachmentSetId;
    }
```

- For API details, see AddAttachmentsToSet in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To add an attachment to a set

The following `add-attachments-to-set` example adds an image to a set that you can then specify for a support case in your AWS account.

```
aws support add-attachments-to-set \
    --attachment-set-id "as-2f5a6faa2a4a1e600-mu-nk5xQlBr70-
G1cUos5LZkd38KOAHZa9BMDVzNEXAMPLE" \
    --attachments fileName=troubleshoot-screenshot.png,data=base64-encoded-string
```

Output:

```
{
    "attachmentSetId": "as-2f5a6faa2a4a1e600-mu-nk5xQlBr70-
G1cUos5LZkd38KOAHZa9BMDVzNEXAMPLE",
```

```
    "expiryTime": "2020-05-14T17:04:40.790+0000"
}
```

For more information, see Case management in the *AWS Support User Guide*.

- For API details, see AddAttachmentsToSet in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static String addAttachment(SupportClient supportClient, String
fileAttachment) {
       try {
           File myFile = new File(fileAttachment);
           InputStream sourceStream = new FileInputStream(myFile);
           SdkBytes sourceBytes = SdkBytes.fromInputStream(sourceStream);

           Attachment attachment = Attachment.builder()
                   .fileName(myFile.getName())
                   .data(sourceBytes)
                   .build();

           AddAttachmentsToSetRequest setRequest =
AddAttachmentsToSetRequest.builder()
                   .attachments(attachment)
                   .build();

           AddAttachmentsToSetResponse response =
supportClient.addAttachmentsToSet(setRequest);
           return response.attachmentSetId();

       } catch (SupportException | FileNotFoundException e) {
           System.out.println(e.getLocalizedMessage());
           System.exit(1);
       }
```

```
        return "";
    }
```

- For API details, see AddAttachmentsToSet in *AWS SDK for Java 2.x API Reference*.

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import { AddAttachmentsToSetCommand } from "@aws-sdk/client-support";

import { client } from "../libs/client.js";

export const main = async () => {
  try {
    // Create a new attachment set or add attachments to an existing set.
    // Provide an 'attachmentSetId' value to add attachments to an existing set.
    // Use AddCommunicationToCase or CreateCase to associate an attachment set
 with a support case.
    const response = await client.send(
      new AddAttachmentsToSetCommand({
        // You can add up to three attachments per set. The size limit is 5 MB
 per attachment.
        attachments: [
          {
            fileName: "example.txt",
            data: new TextEncoder().encode("some example text"),
          },
        ],
      }),
    );
    // Use this ID in AddCommunicationToCase or CreateCase.
    console.log(response.attachmentSetId);
    return response;
  } catch (err) {
```

```
    console.error(err);
  }
};
```

- For API details, see AddAttachmentsToSet in *AWS SDK for JavaScript API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun addAttachment(fileAttachment: String): String? {
    val myFile = File(fileAttachment)
    val sourceBytes = (File(fileAttachment).readBytes())
    val attachmentVal =
        Attachment {
            fileName = myFile.name
            data = sourceBytes
        }

    val setRequest =
        AddAttachmentsToSetRequest {
            attachments = listOf(attachmentVal)
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.addAttachmentsToSet(setRequest)
        return response.attachmentSetId
    }
}
```

- For API details, see AddAttachmentsToSet in *AWS SDK for Kotlin API reference*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
class SupportWrapper:
    """Encapsulates Support actions."""

    def __init__(self, support_client):
        """
        :param support_client: A Boto3 Support client.
        """
        self.support_client = support_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        support_client = boto3.client("support")
        return cls(support_client)


    def add_attachment_to_set(self):
        """
        Add an attachment to a set, or create a new attachment set if one does
 not exist.

        :return: The attachment set ID.
        """
        try:
            response = self.support_client.add_attachments_to_set(
                attachments=[
                    {
                        "fileName": "attachment_file.txt",
                        "data": b"This is a sample file for attachment to a
 support case.",
```

```
                    }
                ]
            )
            new_set_id = response["attachmentSetId"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't add attachment. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return new_set_id
```

- For API details, see AddAttachmentsToSet in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `AddCommunicationToCase` with an AWS SDK or CLI

The following code examples show how to use `AddCommunicationToCase`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// Add communication to a case, including optional attachment set ID and CC
 email addresses.
    /// </summary>
    /// <param name="caseId">Id for the support case.</param>
    /// <param name="body">Body text of the communication.</param>
    /// <param name="attachmentSetId">Optional Id for an attachment set.</param>
    /// <param name="ccEmailAddresses">Optional list of CC email addresses.</
param>
    /// <returns>True if successful.</returns>
    public async Task<bool> AddCommunicationToCase(string caseId, string body,
        string? attachmentSetId = null, List<string>? ccEmailAddresses = null)
    {
        var response = await _amazonSupport.AddCommunicationToCaseAsync(
            new AddCommunicationToCaseRequest()
            {
                CaseId = caseId,
                CommunicationBody = body,
                AttachmentSetId = attachmentSetId,
                CcEmailAddresses = ccEmailAddresses
            });
        return response.Result;
    }
```

- For API details, see AddCommunicationToCase in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To add communication to a case

The following `add-communication-to-case` example adds communications to a support case in your AWS account.

```
aws support add-communication-to-case \
    --case-id "case-12345678910-2013-c4c1d2bf33c5cf47" \
    --communication-body "I'm attaching a set of images to this case." \
    --cc-email-addresses "myemail@example.com" \
    --attachment-set-id "as-2f5a6faa2a4a1e600-mu-nk5xQlBr70-
G1cUos5LZkd38KOAHZa9BMDVzNEXAMPLE"
```

Output:

```
{
    "result": true
}
```

For more information, see Case management in the *AWS Support User Guide*.

- For API details, see AddCommunicationToCase in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void addAttachSupportCase(SupportClient supportClient, String
caseId, String attachmentSetId) {
       try {
           AddCommunicationToCaseRequest caseRequest =
AddCommunicationToCaseRequest.builder()
```

```
                    .caseId(caseId)
                    .attachmentSetId(attachmentSetId)
                    .communicationBody("Please refer to attachment for details.")
                    .build();

            AddCommunicationToCaseResponse response =
 supportClient.addCommunicationToCase(caseRequest);
            if (response.result())
                System.out.println("You have successfully added a communication
 to an AWS Support case");
            else
                System.out.println("There was an error adding the communication
 to an AWS Support case");

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
```

- For API details, see AddCommunicationToCase in *AWS SDK for Java 2.x API Reference*.

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import { AddCommunicationToCaseCommand } from "@aws-sdk/client-support";

import { client } from "../libs/client.js";

export const main = async () => {
  let attachmentSetId;

  try {
    // Add a communication to a case.
```

```
    const response = await client.send(
      new AddCommunicationToCaseCommand({
        communicationBody: "Adding an attachment.",
        // Set value to an existing support case id.
        caseId: "CASE_ID",
        // Optional. Set value to an existing attachment set id to add
 attachments to the case.
        attachmentSetId,
      }),
    );
    console.log(response);
    return response;
  } catch (err) {
    console.error(err);
  }
};
```

- For API details, see AddCommunicationToCase in *AWS SDK for JavaScript API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun addAttachSupportCase(
    caseIdVal: String?,
    attachmentSetIdVal: String?,
) {
    val caseRequest =
        AddCommunicationToCaseRequest {
            caseId = caseIdVal
            attachmentSetId = attachmentSetIdVal
            communicationBody = "Please refer to attachment for details."
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
```

```
        val response = supportClient.addCommunicationToCase(caseRequest)
        if (response.result) {
            println("You have successfully added a communication to an AWS
 Support case")
        } else {
            println("There was an error adding the communication to an AWS
 Support case")
        }
    }
}
```

- For API details, see AddCommunicationToCase in *AWS SDK for Kotlin API reference*.

PowerShell

Tools for PowerShell V4

Example 1: Adds the body of an email communication to the specified case.

```
Add-ASACommunicationToCase -CaseId "case-12345678910-2013-c4c1d2bf33c5cf47" -
CommunicationBody "Some text about the case"
```

Example 2: Adds the body of an email communication to the specified case plus one or more email addresses contained in the CC line of the email.

```
Add-ASACommunicationToCase -CaseId "case-12345678910-2013-c4c1d2bf33c5cf47" -
CcEmailAddress @("email1@address.com", "email2@address.com") -CommunicationBody
 "Some text about the case"
```

- For API details, see AddCommunicationToCase in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Adds the body of an email communication to the specified case.

```
Add-ASACommunicationToCase -CaseId "case-12345678910-2013-c4c1d2bf33c5cf47" -
CommunicationBody "Some text about the case"
```

Example 2: Adds the body of an email communication to the specified case plus one or more email addresses contained in the CC line of the email.

```
Add-ASACommunicationToCase -CaseId "case-12345678910-2013-c4c1d2bf33c5cf47" -
CcEmailAddress @("email1@address.com", "email2@address.com") -CommunicationBody
 "Some text about the case"
```

- For API details, see AddCommunicationToCase in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
class SupportWrapper:
    """Encapsulates Support actions."""

    def __init__(self, support_client):
        """
        :param support_client: A Boto3 Support client.
        """
        self.support_client = support_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        support_client = boto3.client("support")
        return cls(support_client)


    def add_communication_to_case(self, attachment_set_id, case_id):
        """
        Add a communication and an attachment set to a case.

        :param attachment_set_id: The ID of an existing attachment set.
        :param case_id: The ID of the case.
```

```
        """
        try:
            self.support_client.add_communication_to_case(
                caseId=case_id,
                communicationBody="This is an example communication added to a
support case.",
                attachmentSetId=attachment_set_id,
            )
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't add communication. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
```

- For API details, see AddCommunicationToCase in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `CreateCase` with an AWS SDK or CLI

The following code examples show how to use `CreateCase`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   /// <summary>
   /// Create a new support case.
   /// </summary>
   /// <param name="serviceCode">Service code for the new case.</param>
   /// <param name="categoryCode">Category for the new case.</param>
   /// <param name="severityCode">Severity code for the new case.</param>
   /// <param name="subject">Subject of the new case.</param>
   /// <param name="body">Body text of the new case.</param>
   /// <param name="language">Optional language support for your case.
   /// Currently Chinese (“zh”), English ("en"), Japanese ("ja") and Korean
(“ko”) are supported.</param>
   /// <param name="attachmentSetId">Optional Id for an attachment set for the
new case.</param>
   /// <param name="issueType">Optional issue type for the new case. Options are
"customer-service" or "technical".</param>
   /// <returns>The caseId of the new support case.</returns>
   public async Task<string> CreateCase(string serviceCode, string categoryCode,
string severityCode, string subject,
       string body, string language = "en", string? attachmentSetId = null,
string issueType = "customer-service")
   {
       var response = await _amazonSupport.CreateCaseAsync(
           new CreateCaseRequest()
           {
               ServiceCode = serviceCode,
               CategoryCode = categoryCode,
               SeverityCode = severityCode,
               Subject = subject,
               Language = language,
               AttachmentSetId = attachmentSetId,
               IssueType = issueType,
```

```
                CommunicationBody = body
            });
        return response.CaseId;
    }
```

- For API details, see CreateCase in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To create a case

The following `create-case` example creates a support case for your AWS account.

```
aws support create-case \
    --category-code "using-aws" \
    --cc-email-addresses "myemail@example.com" \
    --communication-body "I want to learn more about an AWS service." \
    --issue-type "technical" \
    --language "en" \
    --service-code "general-info" \
    --severity-code "low" \
    --subject "Question about my account"
```

Output:

```
{
    "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47"
}
```

For more information, see Case management in the *AWS Support User Guide*.

- For API details, see CreateCase in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static String createSupportCase(SupportClient supportClient,
List<String> sevCatList, String sevLevel) {
       try {
           String serviceCode = sevCatList.get(0);
           String caseCat = sevCatList.get(1);
           CreateCaseRequest caseRequest = CreateCaseRequest.builder()
                   .categoryCode(caseCat.toLowerCase())
                   .serviceCode(serviceCode.toLowerCase())
                   .severityCode(sevLevel.toLowerCase())
                   .communicationBody("Test issue with " +
serviceCode.toLowerCase())
                   .subject("Test case, please ignore")
                   .language("en")
                   .issueType("technical")
                   .build();

           CreateCaseResponse response = supportClient.createCase(caseRequest);
           return response.caseId();

       } catch (SupportException e) {
           System.out.println(e.getLocalizedMessage());
           System.exit(1);
       }
       return "";
   }
```

- For API details, see CreateCase in *AWS SDK for Java 2.x API Reference*.

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import { CreateCaseCommand } from "@aws-sdk/client-support";

import { client } from "../libs/client.js";

export const main = async () => {
  try {
    // Create a new case and log the case id.
    // Important: This creates a real support case in your account.
    const response = await client.send(
      new CreateCaseCommand({
        // The subject line of the case.
        subject: "IGNORE: Test case",
        // Use DescribeServices to find available service codes for each service.
        serviceCode: "service-quicksight-end-user",
        // Use DescribeSecurityLevels to find available severity codes for your
 support plan.
        severityCode: "low",
        // Use DescribeServices to find available category codes for each
 service.
        categoryCode: "end-user-support",
        // The main description of the support case.
        communicationBody: "This is a test. Please ignore.",
      }),
    );
    console.log(response.caseId);
    return response;
  } catch (err) {
    console.error(err);
  }
};
```

- For API details, see CreateCase in *AWS SDK for JavaScript API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun createSupportCase(
    sevCatListVal: List<String>,
    sevLevelVal: String,
): String? {
    val serCode = sevCatListVal[0]
    val caseCategory = sevCatListVal[1]
    val caseRequest =
        CreateCaseRequest {
            categoryCode = caseCategory.lowercase(Locale.getDefault())
            serviceCode = serCode.lowercase(Locale.getDefault())
            severityCode = sevLevelVal.lowercase(Locale.getDefault())
            communicationBody = "Test issue with
 ${serCode.lowercase(Locale.getDefault())}"
            subject = "Test case, please ignore"
            language = "en"
            issueType = "technical"
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.createCase(caseRequest)
        return response.caseId
    }
}
```

- For API details, see CreateCase in *AWS SDK for Kotlin API reference*.

PowerShell

Tools for PowerShell V4

Example 1: Creates a new case in the AWS Support Center. Values for the -ServiceCode and -CategoryCode parameters can be obtained using the Get-ASAService cmdlet. The value for the -SeverityCode parameter can be obtained using the Get-ASASeverityLevel cmdlet. The -IssueType parameter value can be either "customer-service" or "technical". If successful the AWS Support case number is output. By default the case will be handled in English, to use Japanese add the -Language "ja" parameter. The -ServiceCode, -CategoryCode, -Subject and -CommunicationBody parameters are mandatory.

```
New-ASACase -ServiceCode "amazon-cloudfront" -CategoryCode "APIs" -SeverityCode
 "low" -Subject "subject text" -CommunicationBody "description of the case" -
CcEmailAddress @("email1@domain.com", "email2@domain.com") -IssueType "technical"
```

- For API details, see CreateCase in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Creates a new case in the AWS Support Center. Values for the -ServiceCode and -CategoryCode parameters can be obtained using the Get-ASAService cmdlet. The value for the -SeverityCode parameter can be obtained using the Get-ASASeverityLevel cmdlet. The -IssueType parameter value can be either "customer-service" or "technical". If successful the AWS Support case number is output. By default the case will be handled in English, to use Japanese add the -Language "ja" parameter. The -ServiceCode, -CategoryCode, -Subject and -CommunicationBody parameters are mandatory.

```
New-ASACase -ServiceCode "amazon-cloudfront" -CategoryCode "APIs" -SeverityCode
 "low" -Subject "subject text" -CommunicationBody "description of the case" -
CcEmailAddress @("email1@domain.com", "email2@domain.com") -IssueType "technical"
```

- For API details, see CreateCase in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
class SupportWrapper:
    """Encapsulates Support actions."""

    def __init__(self, support_client):
        """
        :param support_client: A Boto3 Support client.
        """
        self.support_client = support_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        support_client = boto3.client("support")
        return cls(support_client)


    def create_case(self, service, category, severity):
        """
        Create a new support case.

        :param service: The service to use for the new case.
        :param category: The category to use for the new case.
        :param severity: The severity to use for the new case.
        :return: The caseId of the new case.
        """
        try:
            response = self.support_client.create_case(
                subject="Example case for testing, ignore.",
                serviceCode=service["code"],
                severityCode=severity["code"],
```

```
                categoryCode=category["code"],
                communicationBody="Example support case body.",
                language="en",
                issueType="customer-service",
            )
            case_id = response["caseId"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't create case. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return case_id
```

- For API details, see CreateCase in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `DescribeAttachment` with an AWS SDK or CLI

The following code examples show how to use `DescribeAttachment`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// Get description of a specific attachment.
    /// </summary>
    /// <param name="attachmentId">Id of the attachment, usually fetched by
describing the communications of a case.</param>
    /// <returns>The attachment object.</returns>
    public async Task<Attachment> DescribeAttachment(string attachmentId)
    {
        var response = await _amazonSupport.DescribeAttachmentAsync(
            new DescribeAttachmentRequest()
            {
                AttachmentId = attachmentId
            });
        return response.Attachment;
    }
```

- For API details, see DescribeAttachment in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To describe an attachment

The following `describe-attachment` example returns information about the attachment with the specified ID.

```
aws support describe-attachment \
```

```
    --attachment-id "attachment-KBnjRNrePd9D6Jx0-Mm00xZuDEaL2JAj_0-
gJv9qqDooTipsz3V1Nb19rCfkZneeQeDPgp8X1iVJyHH7UuhZDdNeqGoduZsPrAhyMakqlc60-
iJjL5HqyYGiT1FG8EXAMPLE"
```

Output:

```
{
    "attachment": {
        "fileName": "troubleshoot-screenshot.png",
        "data": "base64-blob"
    }
}
```

For more information, see Case management in the *AWS Support User Guide*.

- For API details, see DescribeAttachment in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void describeAttachment(SupportClient supportClient, String
attachId) {
       try {
           DescribeAttachmentRequest attachmentRequest =
DescribeAttachmentRequest.builder()
                   .attachmentId(attachId)
                   .build();

           DescribeAttachmentResponse response =
supportClient.describeAttachment(attachmentRequest);
           System.out.println("The name of the file is " +
response.attachment().fileName());

       } catch (SupportException e) {
```

```
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
```

- For API details, see DescribeAttachment in *AWS SDK for Java 2.x API Reference*.

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import { DescribeAttachmentCommand } from "@aws-sdk/client-support";

import { client } from "../libs/client.js";

export const main = async () => {
  try {
    // Get the metadata and content of an attachment.
    const response = await client.send(
      new DescribeAttachmentCommand({
        // Set value to an existing attachment id.
        // Use DescribeCommunications or DescribeCases to find an attachment id.
        attachmentId: "ATTACHMENT_ID",
      }),
    );
    console.log(response.attachment?.fileName);
    return response;
  } catch (err) {
    console.error(err);
  }
};
```

- For API details, see DescribeAttachment in *AWS SDK for JavaScript API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun describeAttachment(attachId: String?) {
    val attachmentRequest =
        DescribeAttachmentRequest {
            attachmentId = attachId
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.describeAttachment(attachmentRequest)
        println("The name of the file is ${response.attachment?.fileName}")
    }
}
```

- For API details, see DescribeAttachment in *AWS SDK for Kotlin API reference*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
class SupportWrapper:
    """Encapsulates Support actions."""

    def __init__(self, support_client):
        """
```

```
        :param support_client: A Boto3 Support client.
        """
        self.support_client = support_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        support_client = boto3.client("support")
        return cls(support_client)


    def describe_attachment(self, attachment_id):
        """
        Get information about an attachment by its attachmentID.

        :param attachment_id: The ID of the attachment.
        :return: The name of the attached file.
        """
        try:
            response = self.support_client.describe_attachment(
                attachmentId=attachment_id
            )
            attached_file = response["attachment"]["fileName"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't get attachment description. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return attached_file
```

- For API details, see DescribeAttachment in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `DescribeCases` with an AWS SDK or CLI

The following code examples show how to use `DescribeCases`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// Get case details for a list of case ids, optionally with date filters.
    /// </summary>
    /// <param name="caseIds">The list of case IDs.</param>
    /// <param name="displayId">Optional display ID.</param>
    /// <param name="includeCommunication">True to include communication.
 Defaults to true.</param>
    /// <param name="includeResolvedCases">True to include resolved cases.
 Defaults to false.</param>
    /// <param name="afterTime">The optional start date for a filtered search.</
param>
```

```
    /// <param name="beforeTime">The optional end date for a filtered search.</
param>
    /// <param name="language">Optional language support for your case.
    /// Currently Chinese (“zh”), English ("en"), Japanese ("ja") and Korean
 (“ko”) are supported.</param>
    /// <returns>A list of CaseDetails.</returns>
    public async Task<List<CaseDetails>> DescribeCases(List<string> caseIds,
 string? displayId = null, bool includeCommunication = true,
        bool includeResolvedCases = false, DateTime? afterTime = null, DateTime?
 beforeTime = null,
        string language = "en")
    {
        var results = new List<CaseDetails>();
        var paginateCases = _amazonSupport.Paginators.DescribeCases(
            new DescribeCasesRequest()
            {
                CaseIdList = caseIds,
                DisplayId = displayId,
                IncludeCommunications = includeCommunication,
                IncludeResolvedCases = includeResolvedCases,
                AfterTime = afterTime?.ToString("s"),
                BeforeTime = beforeTime?.ToString("s"),
                Language = language
            });
        // Get the entire list using the paginator.
        await foreach (var cases in paginateCases.Cases)
        {
            results.Add(cases);
        }
        return results;
    }
```

- For API details, see DescribeCases in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To describe a case

The following `describe-cases` example returns information about the specified support case in your AWS account.

```
aws support describe-cases \
    --display-id "1234567890" \
    --after-time "2020-03-23T21:31:47.774Z" \
    --include-resolved-cases \
    --language "en" \
    --no-include-communications \
    --max-item 1
```

Output:

```
{
    "cases": [
        {
            "status": "resolved",
            "ccEmailAddresses": [],
            "timeCreated": "2020-03-23T21:31:47.774Z",
            "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47",
            "severityCode": "low",
            "language": "en",
            "categoryCode": "using-aws",
            "serviceCode": "general-info",
            "submittedBy": "myemail@example.com",
            "displayId": "1234567890",
            "subject": "Question about my account"
        }
    ]
}
```

For more information, see Case management in the *AWS Support User Guide*.

- For API details, see DescribeCases in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    public static void getOpenCase(SupportClient supportClient) {
        try {
            // Specify the start and end time.
            Instant now = Instant.now();
            java.time.LocalDate.now();
            Instant yesterday = now.minus(1, ChronoUnit.DAYS);

            DescribeCasesRequest describeCasesRequest =
DescribeCasesRequest.builder()
                    .maxResults(20)
                    .afterTime(yesterday.toString())
                    .beforeTime(now.toString())
                    .build();

            DescribeCasesResponse response =
supportClient.describeCases(describeCasesRequest);
            List<CaseDetails> cases = response.cases();
            for (CaseDetails sinCase : cases) {
                System.out.println("The case status is " + sinCase.status());
                System.out.println("The case Id is " + sinCase.caseId());
                System.out.println("The case subject is " + sinCase.subject());
            }

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
```

- For API details, see DescribeCases in *AWS SDK for Java 2.x API Reference*.

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import { DescribeCasesCommand } from "@aws-sdk/client-support";

import { client } from "../libs/client.js";

export const main = async () => {
  try {
    // Get all of the unresolved cases in your account.
    // Filter or expand results by providing parameters to the
 DescribeCasesCommand. Refer
    // to the TypeScript definition and the API doc for more information on
 possible parameters.
    // https://docs.aws.amazon.com/AWSJavaScriptSDK/v3/latest/clients/client-
support/interfaces/describecasescommandinput.html
    const response = await client.send(new DescribeCasesCommand({}));
    const caseIds = response.cases.map((supportCase) => supportCase.caseId);
    console.log(caseIds);
    return response;
  } catch (err) {
    console.error(err);
  }
};
```

- For API details, see DescribeCases in *AWS SDK for JavaScript API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun getOpenCase() {
    // Specify the start and end time.
    val now = Instant.now()
    LocalDate.now()
    val yesterday = now.minus(1, ChronoUnit.DAYS)
    val describeCasesRequest =
        DescribeCasesRequest {
            maxResults = 20
            afterTime = yesterday.toString()
            beforeTime = now.toString()
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.describeCases(describeCasesRequest)
        response.cases?.forEach { sinCase ->
            println("The case status is ${sinCase.status}")
            println("The case Id is ${sinCase.caseId}")
            println("The case subject is ${sinCase.subject}")
        }
    }
}
```

- For API details, see DescribeCases in *AWS SDK for Kotlin API reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns the details of all support cases.

```
Get-ASACase
```

Example 2: Returns the details of all support cases since the specified date and time.

```
Get-ASACase -AfterTime "2013-09-10T03:06Z"
```

Example 3: Returns the details of the first 10 support cases, including those that have been resolved.

```
Get-ASACase -MaxResult 10 -IncludeResolvedCases $true
```

Example 4: Returns the details of the single specified support case.

```
Get-ASACase -CaseIdList "case-12345678910-2013-c4c1d2bf33c5cf47"
```

Example 5: Returns the details of specified support cases.

```
Get-ASACase -CaseIdList @("case-12345678910-2013-c4c1d2bf33c5cf47",
 "case-18929034710-2011-c4fdeabf33c5cf47")
```

- For API details, see DescribeCases in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns the details of all support cases.

```
Get-ASACase
```

Example 2: Returns the details of all support cases since the specified date and time.

```
Get-ASACase -AfterTime "2013-09-10T03:06Z"
```

Example 3: Returns the details of the first 10 support cases, including those that have been resolved.

```
Get-ASACase -MaxResult 10 -IncludeResolvedCases $true
```

Example 4: Returns the details of the single specified support case.

```
Get-ASACase -CaseIdList "case-12345678910-2013-c4c1d2bf33c5cf47"
```

Example 5: Returns the details of specified support cases.

```
Get-ASACase -CaseIdList @("case-12345678910-2013-c4c1d2bf33c5cf47",
 "case-18929034710-2011-c4fdeabf33c5cf47")
```

- For API details, see DescribeCases in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
class SupportWrapper:
    """Encapsulates Support actions."""

    def __init__(self, support_client):
        """
        :param support_client: A Boto3 Support client.
        """
        self.support_client = support_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        support_client = boto3.client("support")
        return cls(support_client)


    def describe_cases(self, after_time, before_time, resolved):
```

```
        """
        Describe support cases over a period of time, optionally filtering
        by status.

        :param after_time: The start time to include for cases.
        :param before_time: The end time to include for cases.
        :param resolved: True to include resolved cases in the results,
            otherwise results are open cases.
        :return: The final status of the case.
        """
        try:
            cases = []
            paginator = self.support_client.get_paginator("describe_cases")
            for page in paginator.paginate(
                afterTime=after_time,
                beforeTime=before_time,
                includeResolvedCases=resolved,
                language="en",
            ):
                cases += page["cases"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't describe cases. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            if resolved:
                cases = filter(lambda case: case["status"] == "resolved", cases)
            return cases
```

- For API details, see DescribeCases in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `DescribeCommunications` with an AWS SDK or CLI

The following code examples show how to use `DescribeCommunications`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// Describe the communications for a case, optionally with a date filter.
    /// </summary>
    /// <param name="caseId">The ID of the support case.</param>
    /// <param name="afterTime">The optional start date for a filtered search.</
param>
    /// <param name="beforeTime">The optional end date for a filtered search.</
param>
    /// <returns>The list of communications for the case.</returns>
    public async Task<List<Communication>> DescribeCommunications(string caseId,
 DateTime? afterTime = null, DateTime? beforeTime = null)
    {
        var results = new List<Communication>();
        var paginateCommunications =
 _amazonSupport.Paginators.DescribeCommunications(
            new DescribeCommunicationsRequest()
            {
```

```
                CaseId = caseId,
                AfterTime = afterTime?.ToString("s"),
                BeforeTime = beforeTime?.ToString("s")
            });
        // Get the entire list using the paginator.
        await foreach (var communications in
 paginateCommunications.Communications)
        {
            results.Add(communications);
        }
        return results;
    }
```

- For API details, see DescribeCommunications in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To describe the latest communication for a case

The following `describe-communications` example returns the latest communication for the specified support case in your AWS account.

```
aws support describe-communications \
    --case-id "case-12345678910-2013-c4c1d2bf33c5cf47" \
    --after-time "2020-03-23T21:31:47.774Z" \
    --max-item 1
```

Output:

```
{
    "communications": [
        {
            "body": "I want to learn more about an AWS service.",
            "attachmentSet": [],
            "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47",
            "timeCreated": "2020-05-12T23:12:35.000Z",
            "submittedBy": "Amazon Web Services"
        }
```

```
    ],
    "NextToken":
 "eyJuZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQEXAMPLE=="
}
```

For more information, see Case management in the *AWS Support User Guide*.

- For API details, see DescribeCommunications in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static String listCommunications(SupportClient supportClient, String
caseId) {
       try {
           String attachId = null;
           DescribeCommunicationsRequest communicationsRequest =
DescribeCommunicationsRequest.builder()
                   .caseId(caseId)
                   .maxResults(10)
                   .build();

           DescribeCommunicationsResponse response =
supportClient.describeCommunications(communicationsRequest);
           List<Communication> communications = response.communications();
           for (Communication comm : communications) {
               System.out.println("the body is: " + comm.body());

               // Get the attachment id value.
               List<AttachmentDetails> attachments = comm.attachmentSet();
               for (AttachmentDetails detail : attachments) {
                   attachId = detail.attachmentId();
               }
           }
           return attachId;
```

```
        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
        return "";
    }
```

- For API details, see DescribeCommunications in *AWS SDK for Java 2.x API Reference*.

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import { DescribeCommunicationsCommand } from "@aws-sdk/client-support";

import { client } from "../libs/client.js";

export const main = async () => {
  try {
    // Get all communications for the support case.
    // Filter results by providing parameters to the
 DescribeCommunicationsCommand. Refer
    // to the TypeScript definition and the API doc for more information on
 possible parameters.
    // https://docs.aws.amazon.com/AWSJavaScriptSDK/v3/latest/clients/client-
support/interfaces/describecommunicationscommandinput.html
    const response = await client.send(
      new DescribeCommunicationsCommand({
        // Set value to an existing case id.
        caseId: "CASE_ID",
      }),
    );
    const text = response.communications.map((item) => item.body).join("\n");
    console.log(text);
```

```
    return response;
  } catch (err) {
    console.error(err);
  }
};
```

- For API details, see DescribeCommunications in *AWS SDK for JavaScript API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun listCommunications(caseIdVal: String?): String? {
    val communicationsRequest =
        DescribeCommunicationsRequest {
            caseId = caseIdVal
            maxResults = 10
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response =
 supportClient.describeCommunications(communicationsRequest)
        response.communications?.forEach { comm ->
            println("the body is: " + comm.body)
            comm.attachmentSet?.forEach { detail ->
                return detail.attachmentId
            }
        }
    }
    return ""
}
```

- For API details, see DescribeCommunications in *AWS SDK for Kotlin API reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns all communications for the specified case.

```
Get-ASACommunication -CaseId "case-12345678910-2013-c4c1d2bf33c5cf47"
```

Example 2: Returns all communications since midnight UTC on January 1st 2012 for the specified case.

```
Get-ASACommunication -CaseId "case-12345678910-2013-c4c1d2bf33c5cf47" -AfterTime
 "2012-01-10T00:00Z"
```

- For API details, see DescribeCommunications in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns all communications for the specified case.

```
Get-ASACommunication -CaseId "case-12345678910-2013-c4c1d2bf33c5cf47"
```

Example 2: Returns all communications since midnight UTC on January 1st 2012 for the specified case.

```
Get-ASACommunication -CaseId "case-12345678910-2013-c4c1d2bf33c5cf47" -AfterTime
 "2012-01-10T00:00Z"
```

- For API details, see DescribeCommunications in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
class SupportWrapper:
    """Encapsulates Support actions."""

    def __init__(self, support_client):
        """
        :param support_client: A Boto3 Support client.
        """
        self.support_client = support_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        support_client = boto3.client("support")
        return cls(support_client)


    def describe_all_case_communications(self, case_id):
        """
        Describe all the communications for a case using a paginator.

        :param case_id: The ID of the case.
        :return: The communications for the case.
        """
        try:
            communications = []
            paginator =
 self.support_client.get_paginator("describe_communications")
            for page in paginator.paginate(caseId=case_id):
                communications += page["communications"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
 Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
 subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't describe communications. Here's why: %s: %s",
```

```
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
    else:
        return communications
```

- For API details, see DescribeCommunications in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `DescribeServices` with an AWS SDK or CLI

The following code examples show how to use `DescribeServices`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/// <summary>
/// Get the descriptions of AWS services.
/// </summary>
/// <param name="name">Optional language for services.
```

```
    /// Currently Chinese (“zh”), English ("en"), Japanese ("ja") and Korean
 (“ko”) are supported.</param>
    /// <returns>The list of AWS service descriptions.</returns>
    public async Task<List<Service>> DescribeServices(string language = "en")
    {
        var response = await _amazonSupport.DescribeServicesAsync(
            new DescribeServicesRequest()
            {
                Language = language
            });
        return response.Services;
    }
```

- For API details, see DescribeServices in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To list AWS services and service categories

The following `describe-services` example lists the available service categories for requesting general information.

```
aws support describe-services \
    --service-code-list "general-info"
```

Output:

```
{
    "services": [
        {
            "code": "general-info",
            "name": "General Info and Getting Started",
            "categories": [
                {
                    "code": "charges",
                    "name": "How Will I Be Charged?"
                },
                {
```

```
                    "code": "gdpr-queries",
                    "name": "Data Privacy Query"
                },
                {
                    "code": "reserved-instances",
                    "name": "Reserved Instances"
                },
                {
                    "code": "resource",
                    "name": "Where is my Resource?"
                },
                {
                    "code": "using-aws",
                    "name": "Using AWS & Services"
                },
                {
                    "code": "free-tier",
                    "name": "Free Tier"
                },
                {
                    "code": "security-and-compliance",
                    "name": "Security & Compliance"
                },
                {
                    "code": "account-structure",
                    "name": "Account Structure"
                }
            ]
        }
    ]
}
```

For more information, see Case management in the *AWS Support User Guide*.

- For API details, see DescribeServices in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   // Return a List that contains a Service name and Category name.
   public static List<String> displayServices(SupportClient supportClient) {
       try {
           DescribeServicesRequest servicesRequest =
DescribeServicesRequest.builder()
                   .language("en")
                   .build();

           DescribeServicesResponse response =
supportClient.describeServices(servicesRequest);
           String serviceCode = null;
           String catName = null;
           List<String> sevCatList = new ArrayList<>();
           List<Service> services = response.services();

           System.out.println("Get the first 10 services");
           int index = 1;
           for (Service service : services) {
               if (index == 11)
                   break;

               System.out.println("The Service name is: " + service.name());
               if (service.name().compareTo("Account") == 0)
                   serviceCode = service.code();

               // Get the Categories for this service.
               List<Category> categories = service.categories();
               for (Category cat : categories) {
                   System.out.println("The category name is: " + cat.name());
                   if (cat.name().compareTo("Security") == 0)
                       catName = cat.name();
               }
```

```
                index++;
            }

            // Push the two values to the list.
            sevCatList.add(serviceCode);
            sevCatList.add(catName);
            return sevCatList;

        } catch (SupportException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
        return null;
    }
```

- For API details, see DescribeServices in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
// Return a List that contains a Service name and Category name.
suspend fun displayServices(): List<String> {
    var serviceCode = ""
    var catName = ""
    val sevCatList = mutableListOf<String>()
    val servicesRequest =
        DescribeServicesRequest {
            language = "en"
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.describeServices(servicesRequest)
        println("Get the first 10 services")
        var index = 1
```

```
        response.services?.forEach { service ->
            if (index == 11) {
                return@forEach
            }

            println("The Service name is ${service.name}")
            if (service.name == "Account") {
                serviceCode = service.code.toString()
            }

            // Get the categories for this service.
            service.categories?.forEach { cat ->
                println("The category name is ${cat.name}")
                if (cat.name == "Security") {
                    catName = cat.name!!
                }
            }
            index++
        }
    }

    // Push the two values to the list.
    serviceCode.let { sevCatList.add(it) }
    catName.let { sevCatList.add(it) }
    return sevCatList
}
```

- For API details, see DescribeServices in *AWS SDK for Kotlin API reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns all available service codes, names and categories.

```
Get-ASAService
```

Example 2: Returns the name and categories for the service with the specified code.

```
Get-ASAService -ServiceCodeList "amazon-cloudfront"
```

Example 3: Returns the name and categories for the specified service codes.

```
Get-ASAService -ServiceCodeList @("amazon-cloudfront", "amazon-cloudwatch")
```

Example 4: Returns the name and categories (in Japanese) for the specified service codes. Currently English ("en") and Japanese ("ja") language codes are supported.

```
Get-ASAService -ServiceCodeList @("amazon-cloudfront", "amazon-cloudwatch") -
Language "ja"
```

- For API details, see DescribeServices in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns all available service codes, names and categories.

```
Get-ASAService
```

Example 2: Returns the name and categories for the service with the specified code.

```
Get-ASAService -ServiceCodeList "amazon-cloudfront"
```

Example 3: Returns the name and categories for the specified service codes.

```
Get-ASAService -ServiceCodeList @("amazon-cloudfront", "amazon-cloudwatch")
```

Example 4: Returns the name and categories (in Japanese) for the specified service codes. Currently English ("en") and Japanese ("ja") language codes are supported.

```
Get-ASAService -ServiceCodeList @("amazon-cloudfront", "amazon-cloudwatch") -
Language "ja"
```

- For API details, see DescribeServices in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
class SupportWrapper:
    """Encapsulates Support actions."""

    def __init__(self, support_client):
        """
        :param support_client: A Boto3 Support client.
        """
        self.support_client = support_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        support_client = boto3.client("support")
        return cls(support_client)


    def describe_services(self, language):
        """
        Get the descriptions of AWS services available for support for a
 language.

        :param language: The language for support services.
        Currently, only "en" (English) and "ja" (Japanese) are supported.
        :return: The list of AWS service descriptions.
        """
        try:
            response = self.support_client.describe_services(language=language)
            services = response["services"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
```

```
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't get Support services for language %s. Here's why:
%s: %s",
                    language,
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return services
```

- For API details, see DescribeServices in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `DescribeSeverityLevels` with an AWS SDK or CLI

The following code examples show how to use `DescribeSeverityLevels`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// Get the descriptions of support severity levels.
    /// </summary>
    /// <param name="name">Optional language for severity levels.
    /// Currently Chinese (“zh”), English ("en"), Japanese ("ja") and Korean
(“ko”) are supported.</param>
    /// <returns>The list of support severity levels.</returns>
    public async Task<List<SeverityLevel>> DescribeSeverityLevels(string language
= "en")
    {
        var response = await _amazonSupport.DescribeSeverityLevelsAsync(
            new DescribeSeverityLevelsRequest()
            {
                Language = language
            });
        return response.SeverityLevels;
    }
```

- For API details, see DescribeSeverityLevels in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To list the available severity levels

The following `describe-severity-levels` example lists the available severity levels for a support case.

```
aws support describe-severity-levels
```

Output:

```
{
    "severityLevels": [
        {
            "code": "low",
            "name": "Low"
        },
        {
            "code": "normal",
            "name": "Normal"
        },
        {
            "code": "high",
            "name": "High"
        },
        {
            "code": "urgent",
            "name": "Urgent"
        },
        {
            "code": "critical",
            "name": "Critical"
        }
    ]
}
```

For more information, see Choosing a severity in the *AWS Support User Guide*.

- For API details, see DescribeSeverityLevels in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static String displaySevLevels(SupportClient supportClient) {
       try {
           DescribeSeverityLevelsRequest severityLevelsRequest =
DescribeSeverityLevelsRequest.builder()
                   .language("en")
                   .build();

           DescribeSeverityLevelsResponse response =
supportClient.describeSeverityLevels(severityLevelsRequest);
           List<SeverityLevel> severityLevels = response.severityLevels();
           String levelName = null;
           for (SeverityLevel sevLevel : severityLevels) {
               System.out.println("The severity level name is: " +
sevLevel.name());
               if (sevLevel.name().compareTo("High") == 0)
                   levelName = sevLevel.name();
           }
           return levelName;

       } catch (SupportException e) {
           System.out.println(e.getLocalizedMessage());
           System.exit(1);
       }
       return "";
   }
```

- For API details, see DescribeSeverityLevels in *AWS SDK for Java 2.x API Reference*.

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import { DescribeSeverityLevelsCommand } from "@aws-sdk/client-support";
```

```
import { client } from "../libs/client.js";

export const main = async () => {
  try {
    // Get the list of severity levels.
    // The available values depend on the support plan for the account.
    const response = await client.send(new DescribeSeverityLevelsCommand({}));
    console.log(response.severityLevels);
    return response;
  } catch (err) {
    console.error(err);
  }
};
```

- For API details, see DescribeSeverityLevels in *AWS SDK for JavaScript API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun displaySevLevels(): String {
    var levelName = ""
    val severityLevelsRequest =
        DescribeSeverityLevelsRequest {
            language = "en"
        }

    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response =
 supportClient.describeSeverityLevels(severityLevelsRequest)
        response.severityLevels?.forEach { sevLevel ->
            println("The severity level name is: ${sevLevel.name}")
            if (sevLevel.name == "High") {
                levelName = sevLevel.name!!
            }
```

```
        }
        return levelName
    }
}
```

- For API details, see DescribeSeverityLevels in *AWS SDK for Kotlin API reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns the list of severity levels that can be assigned to an AWS Support case.

```
Get-ASASeverityLevel
```

Example 2: Returns the list of severity levels that can be assigned to an AWS Support case. The names of the levels are returned in Japanese.

```
Get-ASASeverityLevel -Language "ja"
```

- For API details, see DescribeSeverityLevels in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns the list of severity levels that can be assigned to an AWS Support case.

```
Get-ASASeverityLevel
```

Example 2: Returns the list of severity levels that can be assigned to an AWS Support case. The names of the levels are returned in Japanese.

```
Get-ASASeverityLevel -Language "ja"
```

- For API details, see DescribeSeverityLevels in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
class SupportWrapper:
    """Encapsulates Support actions."""

    def __init__(self, support_client):
        """
        :param support_client: A Boto3 Support client.
        """
        self.support_client = support_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        support_client = boto3.client("support")
        return cls(support_client)


    def describe_severity_levels(self, language):
        """
        Get the descriptions of available severity levels for support cases for a
 language.

        :param language: The language for support severity levels.
        Currently, only "en" (English) and "ja" (Japanese) are supported.
        :return: The list of severity levels.
        """
        try:
            response =
 self.support_client.describe_severity_levels(language=language)
            severity_levels = response["severityLevels"]
        except ClientError as err:
```

```
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't get severity levels for language %s. Here's why:
%s: %s",
                    language,
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return severity_levels
```

- For API details, see DescribeSeverityLevels in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `DescribeTrustedAdvisorCheckRefreshStatuses` with a CLI

The following code examples show how to use `DescribeTrustedAdvisorCheckRefreshStatuses`.

CLI

AWS CLI

To list the refresh statuses of AWS Trusted Advisor checks

The following `describe-trusted-advisor-check-refresh-statuses` example lists the refresh statuses for two Trusted Advisor checks: Amazon S3 Bucket Permissions and IAM Use.

```
aws support describe-trusted-advisor-check-refresh-statuses \
    --check-id "Pfx0RwqBli" "zXCkfM1nI3"
```

Output:

```
{
    "statuses": [
        {
            "checkId": "Pfx0RwqBli",
            "status": "none",
            "millisUntilNextRefreshable": 0
        },
        {
            "checkId": "zXCkfM1nI3",
            "status": "none",
            "millisUntilNextRefreshable": 0
        }
    ]
}
```

For more information, see AWS Trusted Advisor in the *AWS Support User Guide*.

- For API details, see DescribeTrustedAdvisorCheckRefreshStatuses in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns the current status of refresh requests for the specified checks. Request-ASATrustedAdvisorCheckRefresh can be used to request that the status information of the checks be refreshed.

```
Get-ASATrustedAdvisorCheckRefreshStatus -CheckId @("checkid1", "checkid2")
```

- For API details, see DescribeTrustedAdvisorCheckRefreshStatuses in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns the current status of refresh requests for the specified checks. Request-ASATrustedAdvisorCheckRefresh can be used to request that the status information of the checks be refreshed.

```
Get-ASATrustedAdvisorCheckRefreshStatus -CheckId @("checkid1", "checkid2")
```

- For API details, see DescribeTrustedAdvisorCheckRefreshStatuses in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `DescribeTrustedAdvisorCheckResult` with a CLI

The following code examples show how to use `DescribeTrustedAdvisorCheckResult`.

CLI

AWS CLI

To list the results of an AWS Trusted Advisor check

The following `describe-trusted-advisor-check-result` example lists the results of the IAM Use check.

```
aws support describe-trusted-advisor-check-result \
    --check-id "zXCkfM1nI3"
```

Output:

```
{
    "result": {
        "checkId": "zXCkfM1nI3",
        "timestamp": "2020-05-13T21:38:05Z",
        "status": "ok",
        "resourcesSummary": {
            "resourcesProcessed": 1,
            "resourcesFlagged": 0,
```

```
            "resourcesIgnored": 0,
            "resourcesSuppressed": 0
        },
        "categorySpecificSummary": {
            "costOptimizing": {
                "estimatedMonthlySavings": 0.0,
                "estimatedPercentMonthlySavings": 0.0
            }
        },
        "flaggedResources": [
            {
                "status": "ok",
                "resourceId": "47DEQpj8HBSa-_TImW-5JCeuQeRkm5NMpJWZEXAMPLE",
                "isSuppressed": false
            }
        ]
    }
}
```

For more information, see AWS Trusted Advisor in the *AWS Support User Guide*.

- For API details, see DescribeTrustedAdvisorCheckResult in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns the results of a Trusted Advisor check. The list of available Trusted Advisor checks can be obtained using Get-ASATrustedAdvisorChecks. The output is the overall status of the check, the timestamp at which the check was last run and the unique checkid for the specific check. To have the results output in Japanese, add the -Language "ja" parameter.

```
Get-ASATrustedAdvisorCheckResult -CheckId "checkid1"
```

- For API details, see DescribeTrustedAdvisorCheckResult in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns the results of a Trusted Advisor check. The list of available Trusted Advisor checks can be obtained using Get-ASATrustedAdvisorChecks. The output is

the overall status of the check, the timestamp at which the check was last run and the unique checkid for the specific check. To have the results output in Japanese, add the -Language "ja" parameter.

```
Get-ASATrustedAdvisorCheckResult -CheckId "checkid1"
```

- For API details, see DescribeTrustedAdvisorCheckResult in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `DescribeTrustedAdvisorCheckSummaries` with a CLI

The following code examples show how to use `DescribeTrustedAdvisorCheckSummaries`.

CLI

AWS CLI

To list the summaries of AWS Trusted Advisor checks

The following `describe-trusted-advisor-check-summaries` example lists the results for two Trusted Advisor checks: Amazon S3 Bucket Permissions and IAM Use.

```
aws support describe-trusted-advisor-check-summaries \
    --check-ids "Pfx0RwqBli" "zXCkfM1nI3"
```

Output:

```
{
    "summaries": [
        {
            "checkId": "Pfx0RwqBli",
            "timestamp": "2020-05-13T21:38:12Z",
            "status": "ok",
            "hasFlaggedResources": true,
            "resourcesSummary": {
```

```
                "resourcesProcessed": 44,
                "resourcesFlagged": 0,
                "resourcesIgnored": 0,
                "resourcesSuppressed": 0
            },
            "categorySpecificSummary": {
                "costOptimizing": {
                    "estimatedMonthlySavings": 0.0,
                    "estimatedPercentMonthlySavings": 0.0
                }
            }
        },
        {
            "checkId": "zXCkfM1nI3",
            "timestamp": "2020-05-13T21:38:05Z",
            "status": "ok",
            "hasFlaggedResources": true,
            "resourcesSummary": {
                "resourcesProcessed": 1,
                "resourcesFlagged": 0,
                "resourcesIgnored": 0,
                "resourcesSuppressed": 0
            },
            "categorySpecificSummary": {
                "costOptimizing": {
                    "estimatedMonthlySavings": 0.0,
                    "estimatedPercentMonthlySavings": 0.0
                }
            }
        }
    ]
}
```


For more information, see AWS Trusted Advisor in the *AWS Support User Guide*.

- For API details, see DescribeTrustedAdvisorCheckSummaries in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns the latest summary for the specified Trusted Advisor check.

```
Get-ASATrustedAdvisorCheckSummary -CheckId "checkid1"
```

Example 2: Returns the latest summaries for the specified Trusted Advisor checks.

```
Get-ASATrustedAdvisorCheckSummary -CheckId @("checkid1", "checkid2")
```

- For API details, see DescribeTrustedAdvisorCheckSummaries in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns the latest summary for the specified Trusted Advisor check.

```
Get-ASATrustedAdvisorCheckSummary -CheckId "checkid1"
```

Example 2: Returns the latest summaries for the specified Trusted Advisor checks.

```
Get-ASATrustedAdvisorCheckSummary -CheckId @("checkid1", "checkid2")
```

- For API details, see DescribeTrustedAdvisorCheckSummaries in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `DescribeTrustedAdvisorChecks` with a CLI

The following code examples show how to use `DescribeTrustedAdvisorChecks`.

CLI

AWS CLI

To list the available AWS Trusted Advisor checks

The following `describe-trusted-advisor-checks` example lists the available Trusted Advisor checks in your AWS account. This information includes the check name, ID, description, category, and metadata. Note that the output is shortened for readability.

```
aws support describe-trusted-advisor-checks \
    --language "en"
```

Output:

```
{
    "checks": [
        {
            "id": "zXCkfM1nI3",
            "name": "IAM Use",
            "description": "Checks for your use of AWS Identity and Access
 Management (IAM). You can use IAM to create users, groups, and roles in
 AWS, and you can use permissions to control access to AWS resources. \n<br>
\n<br>\n<b>Alert Criteria</b><br>\nYellow: No IAM users have been created
 for this account.\n<br>\n<br>\n<b>Recommended Action</b><br>\nCreate one or
 more IAM users and groups in your account. You can then create additional
 users whose permissions are limited to perform specific tasks in your AWS
 environment. For more information, see <a href=\"https://docs.aws.amazon.com/
IAM/latest/UserGuide/IAMGettingStarted.html\" target=\"_blank\">Getting
 Started</a>. \n<br><br>\n<b>Additional Resources</b><br>\n<a href=\"https://
docs.aws.amazon.com/IAM/latest/UserGuide/IAM_Introduction.html\" target=\"_blank
\">What Is IAM?</a>",
            "category": "security",
            "metadata": []
        }
    ]
}
```

For more information, see AWS Trusted Advisor in the *AWS Support User Guide*.

- For API details, see DescribeTrustedAdvisorChecks in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns the collection of Trusted Advisor checks. You must specify the Language parameter which can accept either "en" for English output or "ja" for Japanese output.

```
Get-ASATrustedAdvisorCheck -Language "en"
```

- For API details, see DescribeTrustedAdvisorChecks in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns the collection of Trusted Advisor checks. You must specify the Language parameter which can accept either "en" for English output or "ja" for Japanese output.

```
Get-ASATrustedAdvisorCheck -Language "en"
```

- For API details, see DescribeTrustedAdvisorChecks in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `RefreshTrustedAdvisorCheck` with a CLI

The following code examples show how to use `RefreshTrustedAdvisorCheck`.

CLI

AWS CLI

To refresh an AWS Trusted Advisor check

The following `refresh-trusted-advisor-check` example refreshes the Amazon S3 Bucket Permissions Trusted Advisor check in your AWS account.

```
aws support refresh-trusted-advisor-check \
    --check-id "Pfx0RwqBli"
```

Output:

```
{
    "status": {
        "checkId": "Pfx0RwqBli",
        "status": "enqueued",
```

```
        "millisUntilNextRefreshable": 3599992
    }
}
```

For more information, see AWS Trusted Advisor in the *AWS Support User Guide*.

- For API details, see RefreshTrustedAdvisorCheck in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: Requests a refresh for the specified Trusted Advisor check.

```
Request-ASATrustedAdvisorCheckRefresh -CheckId "checkid1"
```

- For API details, see RefreshTrustedAdvisorCheck in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Requests a refresh for the specified Trusted Advisor check.

```
Request-ASATrustedAdvisorCheckRefresh -CheckId "checkid1"
```

- For API details, see RefreshTrustedAdvisorCheck in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `ResolveCase` with an AWS SDK or CLI

The following code examples show how to use `ResolveCase`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/// <summary>
/// Resolve a support case by caseId.
/// </summary>
/// <param name="caseId">Id for the support case.</param>
/// <returns>The final status of the case after resolving.</returns>
public async Task<string> ResolveCase(string caseId)
{
    var response = await _amazonSupport.ResolveCaseAsync(
        new ResolveCaseRequest()
        {
            CaseId = caseId
        });
    return response.FinalCaseStatus;
}
```

- For API details, see ResolveCase in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To resolve a support case

The following `resolve-case` example resolves a support case in your AWS account.

```
aws support resolve-case \
    --case-id "case-12345678910-2013-c4c1d2bf33c5cf47"
```

Output:

```
{
    "finalCaseStatus": "resolved",
    "initialCaseStatus": "work-in-progress"
}
```

For more information, see Case management in the *AWS Support User Guide*.

- For API details, see ResolveCase in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void resolveSupportCase(SupportClient supportClient, String
caseId) {
       try {
           ResolveCaseRequest caseRequest = ResolveCaseRequest.builder()
                   .caseId(caseId)
                   .build();

           ResolveCaseResponse response =
supportClient.resolveCase(caseRequest);
           System.out.println("The status of case " + caseId + " is " +
response.finalCaseStatus());

       } catch (SupportException e) {
           System.out.println(e.getLocalizedMessage());
           System.exit(1);
       }
   }
```

- For API details, see ResolveCase in *AWS SDK for Java 2.x API Reference*.

JavaScript

SDK for JavaScript (v3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import { ResolveCaseCommand } from "@aws-sdk/client-support";

import { client } from "../libs/client.js";

const main = async () => {
  try {
    const response = await client.send(
      new ResolveCaseCommand({
        caseId: "CASE_ID",
      }),
    );

    console.log(response.finalCaseStatus);
    return response;
  } catch (err) {
    console.error(err);
  }
};
```

- For API details, see ResolveCase in *AWS SDK for JavaScript API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun resolveSupportCase(caseIdVal: String) {
    val caseRequest =
        ResolveCaseRequest {
            caseId = caseIdVal
        }
    SupportClient.fromEnvironment { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.resolveCase(caseRequest)
        println("The status of case $caseIdVal is ${response.finalCaseStatus}")
    }
}
```

- For API details, see ResolveCase in *AWS SDK for Kotlin API reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns the initial state of the specified case and the current state after the call to resolve it is completed.

```
Resolve-ASACase -CaseId "case-12345678910-2013-c4c1d2bf33c5cf47"
```

- For API details, see ResolveCase in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns the initial state of the specified case and the current state after the call to resolve it is completed.

```
Resolve-ASACase -CaseId "case-12345678910-2013-c4c1d2bf33c5cf47"
```

- For API details, see ResolveCase in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

Python

SDK for Python (Boto3)

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
class SupportWrapper:
    """Encapsulates Support actions."""

    def __init__(self, support_client):
        """
        :param support_client: A Boto3 Support client.
        """
        self.support_client = support_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        support_client = boto3.client("support")
        return cls(support_client)


    def resolve_case(self, case_id):
        """
        Resolve a support case by its caseId.

        :param case_id: The ID of the case to resolve.
        :return: The final status of the case.
        """
        try:
            response = self.support_client.resolve_case(caseId=case_id)
            final_status = response["finalCaseStatus"]
        except ClientError as err:
            if err.response["Error"]["Code"] == "SubscriptionRequiredException":
                logger.info(
```

```
                    "You must have a Business, Enterprise On-Ramp, or Enterprise
Support "
                    "plan to use the AWS Support API. \n\tPlease upgrade your
subscription to run these "
                    "examples."
                )
            else:
                logger.error(
                    "Couldn't resolve case. Here's why: %s: %s",
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return final_status
```

- For API details, see ResolveCase in *AWS SDK for Python (Boto3) API Reference*.

For a complete list of AWS SDK developer guides and code examples, see Using AWS Support with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Monitoring and logging for AWS Support

Monitoring is an important part of maintaining the reliability, availability, and performance of Support and your other AWS solutions. AWS provides the following monitoring tools to watch Support, report when something is wrong, and take automatic actions when appropriate:

- *Amazon EventBridge* delivers a near real-time stream of system events that describe changes in AWS resources. EventBridge enables automated event-driven computing, as you can write rules that watch for certain events and trigger automated actions in other AWS services when these events happen. For more information, see the Amazon EventBridge User Guide.
- *AWS CloudTrail* captures API calls and related events made by or on behalf of your AWS account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see the AWS CloudTrail User Guide.

Topics

Integrating AWS Support into event-driven applications using Amazon EventBridge

You can incorporate AWS Support into event-driven applications (EDAs) that use events that occur in AWS Support to communicate between application components and initiate downstream processes.

For example, you can get notified whenever the following AWS Support events occur in your account:

- A support case is created, resolved, or reopened
- A correspondence is added to an existing support case

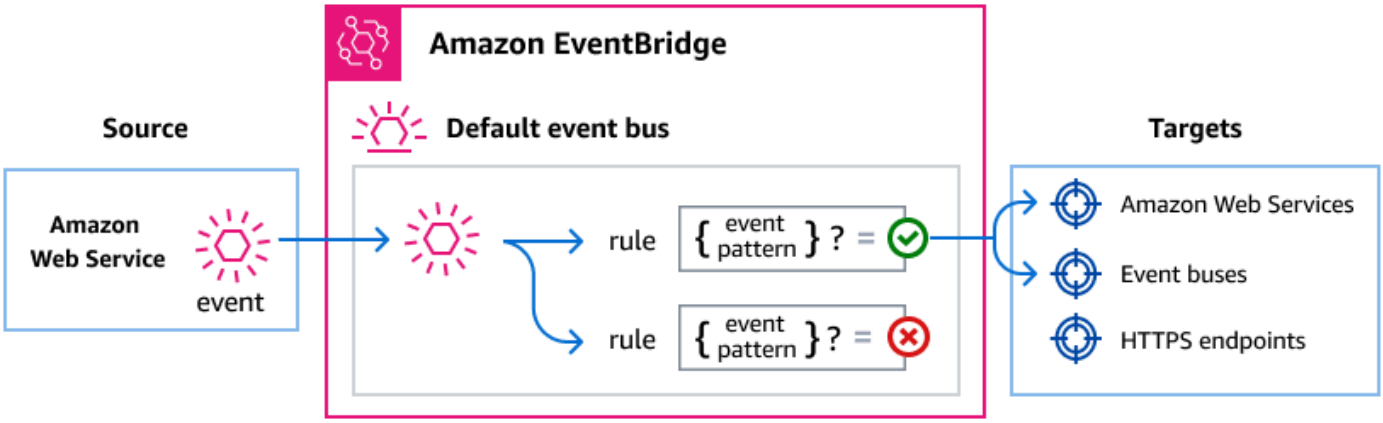
You do this by using Amazon EventBridge to route events from AWS Support to other software components. Amazon EventBridge is a serverless service that uses events to connect application components together, making it easier for you to integrate AWS services like AWS Support into event-driven architectures without additional code and operations.

How EventBridge routes AWS Support events

Here's how EventBridge works with AWS Support events:

As with many AWS services, AWS Support generates and sends events to the EventBridge default *event bus*. An event bus is a router that receives events and routes them to the destinations, or *targets*, that you specify. Targets can include other AWS services, custom applications, and SaaS partner applications.

EventBridge routes events according to *rules* you create on the event bus. For each rule, you specify a filter, or *event pattern*, to select only the events you want. Whenever an event is sent to the event bus, EventBridge compares it against each rule. If the event matches the rule, EventBridge routes the event to the specified target(s).



AWS Support events

AWS Support sends the following events to the default EventBridge event bus automatically.

Event detail type	Description
Support Case Update	Represents a change in a support case.

Event structure

All events from AWS services contain two types of data:

- A common set of fields containing metadata about the event, such as the AWS service that is the source of the event, the time the event was generated, the account and region in which the event took place, and others. For definitions of these general fields, see Event structure in the *Amazon EventBridge Events Reference*.
- A `detail` field that contains data specific to that particular service event.

AWS Support event delivery via AWS CloudTrail

AWS services can send events directly to the EventBridge default event bus. In addition, AWS CloudTrail sends events originating from numerous AWS services to EventBridge as well. These events can include API calls, console signins and actions, service events, and CloudTrail Insights. For more information, see AWS service events delivered via AWS CloudTrail in the *EventBridge User Guide*.

For a list of AWS Support events sent to EventBridge, refer to the AWS Support topic in the *EventBridge Events Reference*.

Creating event patterns that match AWS Support events

Event patterns are filters where specify the data that the events you want to select should have.

Each event pattern is a JSON object that contains:

- A `source` attribute that identifies the service sending the event. For AWS Support events, the source is `aws.support`.
- (Optional): A `detail-type` attribute that contains an array of the event names to match.
- (Optional): A `detail` attribute containing any other event data on which to match.

For example, the following event pattern would select all Support Case Update events from AWS Support:

```
{
  "source": ["aws.support"],
```

```
  "detail-type": ["Support Case Update"]
}
```

You can get more specific in your event selection by including values in the event itself. For example, the following event pattern matches Support Case Update events that represent a case being reopened:

```
{
  "source": ["aws.support"],
  "detail-type": ["Support Case Update"],
  "detail": {
      "event-name": "ReopenCase"
    }
}
```

For more information on writing event patterns, see Event patterns in the *EventBridge User Guide*.

See also

For more information about how to use EventBridge with AWS Support, see the following resources:

- How to automate AWS Support API with Amazon EventBridge
- AWS Support case activity notifier on GitHub

Support Case Update event

Below are the detail fields for the `Support Case Update` event.

The `source` and `detail-type` fields are included below because they contain specific values for AWS Support events. For definitions of the other metadata fields that are included in all events, see Event structure in the *Amazon EventBridge Events Reference*.

```
{
  . . .,
  "detail-type": "Support Case Update",
  "source": "aws.support",
  . . .,
  "detail": {
```

```
    "case-id" : "string",
    "display-id" : "string",
    "communication-id" : "string",
    "event-name" : "string",
    "origin" : "string"
  }
}
```

`detail-type`

Identifies the type of event.

For this event, this value is `Support Case Update`.

`source`

Identifies the service that generated the event. For AWS Support events, this value is `aws.support`.

`detail`

A JSON object that contains information about the event. The service generating the event determines the content of this field.

For this event, this data includes:

`case-id`

The support case ID. The case ID is an alphanumeric string in the following format: case-*12345678910-2013-c4c1d2bf33c5cf47*.

`display-id`

The identifier for the case on pages in the AWS Support Center.

`communication-id`

The communication ID.

`event-name`

Valid values: `CreateCase` | `AddCommunicationToCase` | `ResolveCase` | `ReopenCase`

Specifies the type of support case event.

`origin`

Valid values: `AWS` | `CUSTOMER`

Specifies whether you or an AWS Support agent added a case correspondence to a support case.

Currently, only events with an `event-name` of `AddCommunicationToCase` will contain have this value.

Example Support Case Update event example: Support case created

```
{
    "version": "0",
    "id": "3433df007-9285-55a3-f6d1-536944be45d7",
    "detail-type": "Support Case Update",
    "source": "aws.support",
    "account": "111122223333",
    "time": "2022-02-21T15:51:19Z",
    "region": "us-east-1",
    "resources": [],
    "detail": {
        "case-id": "case-111122223333-muen-2022-7118885805350839",
        "display-id": "1234563851",
        "communication-id": "",
        "event-name": "CreateCase",
        "origin": ""
    }
}
```

Example Support Case Update event example: AWS Support replies to a support case

```
{
    "version": "0",
    "id": "f90cb8cb-32be-1c91-c0ba-d50b4ca5e51b",
    "detail-type": "Support Case Update",
    "source": "aws.support",
    "account": "111122223333",
    "time": "2022-02-21T15:51:31Z",
    "region": "us-east-1",
    "resources": [],
    "detail": {
```

```
        "case-id": "case-111122223333-muen-2022-7118885805350839",
        "display-id": "1234563851",
        "communication-id": "ekko:us-east-1:12345678-268a-424b-be08-54613cab84d2",
        "event-name": "AddCommunicationToCase",
        "origin": "AWS"
    }
}
```

Example Support Case Update event example: Support case resolved

```
{
    "version": "0",
    "id": "1aa4458d-556f-732e-ddc1-4a5b2fbd14a5",
    "detail-type": "Support Case Update",
    "source": "aws.support",
    "account": "111122223333",
    "time": "2022-02-21T15:51:31Z",
    "region": "us-east-1",
    "resources": [],
    "detail": {
        "case-id": "case-111122223333-muen-2022-7118885805350839",
        "display-id": "1234563851",
        "communication-id": "",
        "event-name": "ResolveCase",
        "origin": ""
    }
}
```

Example Support Case Update event example: Support case reopened

```
{
    "version": "0",
    "id": "3bb9d8fe-6089-ad27-9508-804209b233ad",
    "detail-type": "Support Case Update",
    "source": "aws.support",
    "account": "111122223333",
    "time": "2022-02-21T15:47:19Z",
    "region": "us-east-1",
    "resources": [],
    "detail": {
        "case-id": "case-111122223333-muen-2021-27f40618fe0303ea",
        "display-id": "1234563851",
        "communication-id": "",
```

```
        "event-name": "ReopenCase",
        "origin": ""
    }
}
```

Logging AWS Support API calls with AWS CloudTrail

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

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon Simple Storage Service (Amazon S3) bucket, including events for AWS Support. 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 AWS Support, 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, including how to configure and enable it, see the AWS CloudTrail User Guide.

AWS Support information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When supported event activity occurs in AWS Support, 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 AWS Support, 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 accounts

All AWS Support API operations are logged by CloudTrail and are documented in the AWS Support API Reference.

For example, calls to the `CreateCase`, `DescribeCases` and `ResolveCase` operations 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.

You can also aggregate AWS Support log files from multiple AWS Regions and multiple AWS accounts into a single Amazon S3 bucket.

AWS Trusted Advisor information in CloudTrail logging

Trusted Advisor is an AWS Support service that you can use to check your AWS account for ways to save costs, improve security, and optimize your account.

All Trusted Advisor API operations are logged by CloudTrail and are documented in the AWS Support API Reference.

For example, calls to the `DescribeTrustedAdvisorCheckRefreshStatuses`, `DescribeTrustedAdvisorCheckResult` and `RefreshTrustedAdvisorCheck` operations generate entries in the CloudTrail log files.

> **Note**
>
> CloudTrail also logs Trusted Advisor console actions. See Logging AWS Trusted Advisor console actions with AWS CloudTrail.

Understanding AWS Support 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. It includes information about the requested operation, the date and time of the operation, 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.

Example : Log entry for CreateCase

The following example shows a CloudTrail log entry for the CreateCase operation.


```
{
   "Records": [
      {
         "eventVersion": "1.04",
         "userIdentity": {
            "type": "IAMUser",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:iam::111122223333:user/janedoe",
            "accountId": "111122223333",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "userName": "janedoe",
            "sessionContext": {
               "attributes": {
                  "mfaAuthenticated": "false",
                  "creationDate": "2016-04-13T17:51:37Z"
               }
            },
            "invokedBy": "signin.amazonaws.com"
         },
         "eventTime": "2016-04-13T18:05:53Z",
         "eventSource": "support.amazonaws.com",
         "eventName": "CreateCase",
         "awsRegion": "us-east-1",
         "sourceIPAddress": "198.51.100.15",
         "userAgent": "signin.amazonaws.com",
         "requestParameters": {
            "severityCode": "low",
            "categoryCode": "other",
            "language": "en",
            "serviceCode": "support-api",
            "issueType": "technical"
```

```
        },
        "responseElements": {
           "caseId": "case-111122223333-muen-2016-c3f2077e504940f2"
        },
        "requestID": "58c257ef-01a2-11e6-be2a-01c031063738",
        "eventID": "5aa34bfc-ad5b-4fb1-8a55-2277c86e746a",
        "eventType": "AwsApiCall",
        "recipientAccountId": "111122223333"
      }
   ],
   ...
}
```

Example : Log entry for RefreshTrustedAdvisorCheck

The following example shows a CloudTrail log entry for the RefreshTrustedAdvisorCheck operation.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/Admin",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "Admin"
    },
    "eventTime": "2020-10-21T16:34:13Z",
    "eventSource": "support.amazonaws.com",
    "eventName": "RefreshTrustedAdvisorCheck",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "72.21.198.67",
    "userAgent": "signin.amazonaws.com",
    "requestParameters": {
        "checkId": "Pfx0RwqBli"
    },
    "responseElements": null,
    "requestID": "4c4d5fc8-c403-4f82-9544-41f820e0fa01",
    "eventID": "2f4630ac-5c27-4f0d-b93f-63742d6fc85e",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

Logging AWS Support App in Slack API calls using AWS CloudTrail

The AWS Support App in Slack is integrated with AWS CloudTrail. CloudTrail provides a record of actions taken by a user, role, or an AWS service in the AWS Support App. To create this record, CloudTrail captures all public API calls for AWS Support App as events. These captured calls include calls from the AWS Support App console, and code calls to the AWS Support App public API operations. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket. These include events for AWS Support App. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. You can use the information that CloudTrail collects to determine that the request that was made to AWS Support App. You can also learn the IP address where the call originated, who made the request, when it was made, and additional details.

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

AWS Support App information in CloudTrail

When you create your AWS account, this activates CloudTrail on the account. When public API activity occurs in the AWS Support App, 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 AWS Support App, create a *trail*. 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 analyze further the event data collected in CloudTrail logs and act upon the data. 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 public AWS Support App actions. These actions are also documented in the AWS Support App in Slack API Reference. For example, calls to the `CreateSlackChannelConfiguration`, `GetAccountAlias` and `UpdateSlackChannelConfiguration` 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 AWS Support App 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. This means that the logs don't appear in any specific order.

Example : Log example for `CreateSlackChannelConfiguration`

The following example shows a CloudTrail log entry for the CreateSlackChannelConfiguration operation.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE:JaneDoe",
        "arn": "arn:aws:sts::111122223333:assumed-role/Administrator/JaneDoe",
        "accountId": "111122223333",
        "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
```

```
                "arn": "arn:aws:iam::111122223333:role/Administrator",
                "accountId": "111122223333",
                "userName": "Administrator"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-02-26T01:37:57Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-02-26T01:48:20Z",
    "eventSource": "supportapp.amazonaws.com",
    "eventName": "CreateSlackChannelConfiguration",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "205.251.233.183",
    "userAgent": "aws-cli/1.3.23 Python/2.7.6 Linux/2.6.18-164.el5",
    "requestParameters": {
        "notifyOnCreateOrReopenCase": true,
        "teamId": "T012ABCDEFG",
        "notifyOnAddCorrespondenceToCase": true,
        "notifyOnCaseSeverity": "all",
        "channelName": "troubleshooting-channel",
        "notifyOnResolveCase": true,
        "channelId": "C01234A5BCD",
        "channelRoleArn": "arn:aws:iam::111122223333:role/AWSSupportAppRole"
    },
    "responseElements": null,
    "requestID": "d06df6ca-c233-4ffb-bbff-63470c5dc255",
    "eventID": "0898ce29-a396-444a-899d-b068f390c361",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


Example : Log example for `ListSlackChannelConfigurations`

The following example shows a CloudTrail log entry for the ListSlackChannelConfigurations operation.


```
{
```

```
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE:AWSSupportAppRole",
        "arn": "arn:aws:sts::111122223333:assumed-role/AWSSupportAppRole",
        "accountId": "111122223333",
        "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/AWSSupportAppRole",
                "accountId": "111122223333",
                "userName": "AWSSupportAppRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-03-01T20:06:32Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-03-01T20:06:46Z",
    "eventSource": "supportapp.amazonaws.com",
    "eventName": "ListSlackChannelConfigurations",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "72.21.217.131",
    "userAgent": "aws-cli/1.3.23 Python/2.7.6 Linux/2.6.18-164.el5",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "20f81d63-31c5-4351-bd02-9eda7f76e7b8",
    "eventID": "70acb7fe-3f84-47cd-8c28-cc148ad06d21",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


Example : Log example for `GetAccountAlias`

The following example shows a CloudTrail log entry for the GetAccountAlias operation.


```
{
```

```
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE:devdsk",
        "arn": "arn:aws:sts::111122223333:assumed-role/AWSSupportAppRole/devdsk",
        "accountId": "111122223333",
        "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/AWSSupportAppRole",
                "accountId": "111122223333",
                "userName": "AWSSupportAppRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-03-01T20:31:27Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-03-01T20:31:47Z",
    "eventSource": "supportapp.amazonaws.com",
    "eventName": "GetAccountAlias",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "72.21.217.142",
    "userAgent": "aws-cli/1.3.23 Python/2.7.6 Linux/2.6.18-164.el5",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "a225966c-0906-408b-b8dd-f246665e6758",
    "eventID": "79ebba8d-3285-4023-831a-64af7de8d4ad",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

Monitoring and logging for AWS Support Plans

Monitoring is an important part of maintaining the reliability, availability, and performance of Support Plans and your other AWS solutions. AWS provides the following monitoring tools to watch Support Plans, report when something is wrong, and take automatic actions when appropriate:

- *AWS CloudTrail* captures API calls and related events made by or on behalf of your AWS account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see the AWS CloudTrail User Guide.

Topics

Logging AWS Support Plans API calls with AWS CloudTrail

AWS Support Plans is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service. CloudTrail captures API calls for AWS Support Plans as events. The calls captured include calls from the AWS Support Plans console and code calls to the AWS Support Plans API operations.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon Simple Storage Service (Amazon S3) bucket, including events for AWS Support Plans. 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 AWS Support Plans, 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, including how to configure and enable it, see the AWS CloudTrail User Guide.

AWS Support Plans information in CloudTrail

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

For an ongoing record of events in your account, including events for AWS Support Plans, 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 accounts

All AWS Support Plans API operations are logged by CloudTrail. 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.

You can also aggregate AWS Support Plans log files from multiple AWS Regions and multiple accounts into a single Amazon S3 bucket.

Understanding AWS Support Plans 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. It includes information about the requested operation, the date and time of the operation, 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.

Example : Log entry for `GetSupportPlan`

The following example shows a CloudTrail log entry for the `GetSupportPlan` operation.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:sts::111122223333:user/janedoe",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-06-29T16:30:04Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-06-29T16:39:11Z",
    "eventSource": "supportplans.amazonaws.com",
    "eventName": "GetSupportPlan",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.251.233.183",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:91.0) Gecko/20100101
 Firefox/91.0",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "7665c39a-d6bf-4d0d-8010-2f59740b8ecb",
    "eventID": "b711bc30-16a5-4579-8f0d-9ada8fe6d1ce",
    "readOnly": true,
```

```
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

Example : Log entry for `GetSupportPlanUpdateStatus`

The following example shows a CloudTrail log entry for the `GetSupportPlanUpdateStatus` operation.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:sts::111122223333:user/janedoe",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-06-29T16:30:04Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-06-29T16:39:02Z",
    "eventSource": "supportplans.amazonaws.com",
    "eventName": "GetSupportPlanUpdateStatus",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.251.233.183",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:91.0) Gecko/20100101
 Firefox/91.0",
    "requestParameters": {
```

```
        "supportPlanUpdateArn":
 "arn:aws:supportplans::111122223333:supportplanupdate/7f03b7a233a0e87ebc79e56d4d2bcaf19e976c37
    },
    "responseElements": null,
    "requestID": "75e5c767-8703-4ed3-b01e-4dda28020322",
    "eventID": "28d1c0e3-ccb6-4fd1-8793-65be010114cc",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

Example : Log entry for `StartSupportPlanUpdate`

The following example shows a CloudTrail log entry for the `StartSupportPlanUpdate` operation.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:sts::111122223333:user/janedoe",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-06-29T16:30:04Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-06-29T16:38:55Z",
    "eventSource": "supportplans.amazonaws.com",
```

```
    "eventName": "StartSupportPlanUpdate",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.251.233.183",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:91.0) Gecko/20100101
 Firefox/91.0",
    "requestParameters": {
        "clientToken": "98add111-dcc9-464d-8722-438d697fe242",
        "update": {
            "supportLevel": "BASIC"
        }
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-RequestId,x-amzn-ErrorType,x-amzn-
ErrorMessage,Date",
        "supportPlanUpdateArn":
 "arn:aws:supportplans::111122223333:supportplanupdate/7f03b7a233a0e87ebc79e56d4d2bcaf19e976c37
    },
    "requestID": "e5ff9382-5fb8-4764-9993-0f33fb0b1e17",
    "eventID": "5dba89f8-2e5b-42b9-9b8f-395580c52962",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


Example : Log entry for `CreateSupportPlanSchedule`

The following example shows a CloudTrail log entry for the `CreateSupportPlanSchedule` operation.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:sts::111122223333:user/janedoe",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
```

```
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-05-09T16:30:04Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-05-09T16:30:04Z",
    "eventSource": "supportplans.amazonaws.com",
    "eventName": "CreateSupportPlanSchedule",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.251.233.183",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:91.0) Gecko/20100101
 Firefox/91.0",
    "requestParameters": {
        "clientToken": "b998de5e-ad1c-4448-90db-2bf86d6d9e9a",
        "scheduleCreationDetails": {
            "startLevel": "BUSINESS",
            "startOffer": "TrialPlan7FB93B",
            "startTimestamp": "2023-06-03T17:23:56.109Z",
            "endLevel": "BUSINESS",
            "endOffer": "StandardPlan2074BB",
            "endTimestamp": "2023-09-03T17:23:55.109Z"
        }
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-RequestId,x-amzn-ErrorType,x-amzn-
ErrorMessage,Date",
        "supportPlanUpdateArn":
 "arn:aws:supportplans::111122223333:supportplanschedule/
b9a9a4336a3974950a6e670f7dab79b77a4b104db548a0d57050ce4544721d4b"
    },
    "requestID": "150450b8-e61a-4b15-93a8-c3b557a1ca48",
    "eventID": "a2a1ba44-610d-4dc8-bf16-29f1635b57a9",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
```

```
}
```

Example : Log entry for `ListSupportPlanModifiers`

The following example shows a CloudTrail log entry for the `ListSupportPlanModifiers` operation.

```
{
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:sts::111122223333:user/janedoe",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:sts::111122223333:user/janedoe",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "attributes": {
                "creationDate": "2024-08-15T15:44:43Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2024-08-15T16:29:59Z",
    "eventSource": "supportplans.amazonaws.com",
    "eventName": "ListSupportPlanModifiers",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.251.233.183",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:91.0) Gecko/20100101
 Firefox/91.0",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "7665c39a-d6bf-4d0d-8010-2f59740b8ecb",
    "eventID": "b711bc30-16a5-4579-8f0d-9ada8fe6d1ce",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
```

```
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

Logging changes to your Support plan

Important

As of August 3, 2022, the following operations are deprecated and won't appear in your new CloudTrail logs. For a list of supported operations, see Understanding AWS Support Plans log file entries.

- `DescribeSupportLevelSummary` – This action appears in your log when you open the Support plans page.
- `UpdateProbationAutoCancellation` – After you sign up for Developer Support or Business Support and then try to cancel within 30 days, your plan will be automatically canceled at the end of that period. This action appears in your log when you choose **Opt-out of automatic cancellation** in the banner that appears on the Support plans page. You will resume your plan for Developer Support or Business Support.
- `UpdateSupportLevel` – This action appears in your log when you change your support plan.

Note

The `eventSource` field has the `support-subscription.amazonaws.com` namespace for these actions.

Example : Log entry for DescribeSupportLevelSummary

The following example shows a CloudTrail log entry for the `DescribeSupportLevelSummary` action.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "Root",
    "principalId": "111122223333",
```

```
    "arn": "arn:aws:iam::111122223333:root",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {},
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-01-07T22:08:05Z"
      }
    }
  },
  "eventTime": "2021-01-07T22:08:07Z",
  "eventSource": "support-subscription.amazonaws.com",
  "eventName": "DescribeSupportLevelSummary",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "100.127.8.67",
  "userAgent": "AWS-SupportPlansConsole, aws-internal/3",
  "requestParameters": {
    "lang": "en"
  },
  "responseElements": null,
  "requestID": "b423b84d-829b-4090-a239-2b639b123abc",
  "eventID": "e1eeda0e-d77c-487b-a7e5-4014f7123abc",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "111122223333"
}
```


Example : Log entry for UpdateProbationAutoCancellation

The following example shows a CloudTrail log entry for the `UpdateProbationAutoCancellation` action.


```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "Root",
    "principalId": "111122223333",
    "arn": "arn:aws:iam::111122223333:root",
    "accountId": "111122223333",
```

```
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2021-01-07T23:28:43Z",
  "eventSource": "support-subscription.amazonaws.com",
  "eventName": "UpdateProbationAutoCancellation",
  "awsRegion": "us-east-1", "sourceIPAddress": "100.127.8.67",
  "userAgent": "AWS-SupportPlansConsole, aws-internal/3",
  "requestParameters": {
    "lang": "en"
  },
  "responseElements": null,
  "requestID": "5492206a-e200-4c33-9fcf-4162d4123abc",
  "eventID": "f4a58c09-0bb0-4ba2-a8d3-df6909123abc",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "111122223333"
}
```

Example : Log entry for UpdateSupportLevel

The following example shows a CloudTrail log entry for the `UpdateSupportLevel` action to change to Developer Support.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "Root",
    "principalId": "111122223333",
    "arn": "arn:aws:iam::111122223333:root",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {},
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-01-07T22:08:05Z"
      }
    }
  },
  "eventTime": "2021-01-07T22:08:43Z",
```

```
  "eventSource": "support-subscription.amazonaws.com",
  "eventName": "UpdateSupportLevel",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "100.127.8.247",
  "userAgent": "AWS-SupportPlansConsole, aws-internal/3",
  "requestParameters": {
    "supportLevel": "new_developer"
  },
  "responseElements": {
    "aispl": false,
    "supportLevel": "new_developer"
  },
  "requestID": "5df3da3a-61cd-4a3c-8f41-e5276b123abc",
  "eventID": "c69fb149-c206-47ce-8766-8df6ec123abc",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "111122223333"
}
```

Monitoring and logging for AWS Trusted Advisor

Monitoring is an important part of maintaining the reliability, availability, and performance of Trusted Advisor and your other AWS solutions. AWS provides the following monitoring tools to watch Trusted Advisor, report when something is wrong, and take automatic actions when appropriate:

- *Amazon EventBridge* delivers a near real-time stream of system events that describe changes in AWS resources. EventBridge enables automated event-driven computing, as you can write rules that watch for certain events and trigger automated actions in other AWS services when these events happen.

  For example, Trusted Advisor provides the **Amazon S3 Bucket Permissions** check. This check identifies if you have buckets that have open access permissions or allow access to any authenticated AWS user. If a bucket permission changes, the status changes for the Trusted Advisor check. EventBridge detects this event and then sends you a notification so that you can take action. For more information, see the Amazon EventBridge User Guide.
- AWS Trusted Advisor checks identify ways for you to reduce cost, increase performance, and improve security for your AWS account. You can use EventBridge to monitor the status of Trusted Advisor checks. You can then use Amazon CloudWatch to create alarms on Trusted Advisor metrics. These alarms notify you when the status changes for a Trusted Advisor check, such as an updated resource or a service quota that is reached.
- *AWS CloudTrail* captures API calls and related events made by or on behalf of your AWS account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see the AWS CloudTrail User Guide.

Topics

Monitoring AWS Trusted Advisor check results with Amazon EventBridge

You can use EventBridge to detect when your checks for Trusted Advisor change status. Then, based on the rules that you create, EventBridge invokes one or more target actions when the status changes to a value that you specify in a rule.

Depending on the status change, you can send notifications, capture status information, take corrective action, initiate events, or take other actions. For example, you can specify the following target types if a check changes status from no problems detected (green) to recommended action (red).

- Use an AWS Lambda function to pass a notification to a Slack channel.
- Push data about the check to an Amazon Kinesis stream to support comprehensive and real-time status monitoring.
- Send an Amazon Simple Notification Service topic to your email.
- Get notified with an Amazon CloudWatch alarm action.

For more information about on how to use EventBridge and Lambda functions to automate responses for Trusted Advisor, see Trusted Advisor tools in GitHub.

Notes

- Trusted Advisor delivers events on a best effort basis. Events are not always guaranteed to be delivered to EventBridge.
-
- You must have a Business, Enterprise On-Ramp, or Enterprise AWS Support plan to create a rule for Trusted Advisor checks. For more information, see Change AWS Support Plans.
- As Trusted Advisor is a Global service, all Events are emitted to EventBridge in the US East (N. Virginia) Region.

Follow this procedure to create an EventBridge rule for Trusted Advisor. Before you create event rules, do the following:

- Familiarize yourself with events, rules, and targets in EventBridge. For more information, see What is Amazon EventBridge? in the *Amazon EventBridge User Guide*.
- Create the target that you will use in your event rule.

To create an EventBridge rule for Trusted Advisor

1. Open the Amazon EventBridge console at https://eusc-de-east-1.console.amazonaws-eusc.eu/events/.
2. To change the Region, use the **Region selector** in the upper-right corner of the page and choose **US East (N. Virginia)**.
3. In the navigation pane, choose **Rules**.
4. Choose **Create rule**.
5. On the **Define rule detail** page, enter a name and description for your rule.
6. Keep the default values for **Event bus** and **Rule type**, and then choose **Next**.
7. On the **Build event pattern** page, for **Event source**, choose **AWS events or EventBridge partner events**.
8. Under **Event pattern**, keep the default value for **AWS services**.
9. For **AWS service**, choose **Trusted Advisor**.
10. For **Event type**, choose **Check Item Refresh Status**.
11. Choose one of the following options for check statuses:
    - Choose **Any status** to create a rule that monitors for any status change.
    - Choose **Specific status(es)**, and then choose the values that you want your rule to monitor.
        - **ERROR** – Trusted Advisor recommends an action for the check.
        - **INFO** – Trusted Advisor can't determine the status of the check.
        - **OK** – Trusted Advisor doesn't detect an issue for the check.
        - **WARN** – Trusted Advisor detects a possible issue for the check and recommends investigation.
12. Choose one of the following options for your checks:
    - Choose **Any check**.
    - Choose **Specific check(s)**, and then choose one or more check names from the list.

13. Choose one of the following options for AWS resources:
    - Choose **Any resource ID** to create a rule that monitors all resources.
    - Choose **Specific resource ID(s) by ARN**, and then enter the Amazon Resource Names (ARNs) that you want.
14. Choose **Next**.
15. In the **Select target(s)** page, choose the target type that you created for this rule, and then configure any additional options that are required for that type. For example, you might send the event to an Amazon SQS queue or an Amazon SNS topic.
16. Choose **Next**.
17. (Optional) On the **Configure tags** page, add any tags and then choose **Next**.
18. On the **Review and create** page, review your rule setup and ensure that it meets your event monitoring requirements.
19. Choose **Create rule**. Your rule will now monitor for Trusted Advisor checks and then send the event to the target that you specified.

Creating Amazon CloudWatch alarms to monitor AWS Trusted Advisor metrics

When AWS Trusted Advisor refreshes your checks, Trusted Advisor publishes metrics about your check results to CloudWatch. You can view the metrics in CloudWatch. You can also create alarms to detect status changes to Trusted Advisor checks and status changes for resources, and service quota usage (formerly referred to as limits).

Follow this procedure to create a CloudWatch alarm for a specific Trusted Advisor metric.

Topics

Prerequisites

Before you create CloudWatch alarms for Trusted Advisor metrics, review the following information:

- Understand how CloudWatch uses metrics and alarms. For more information, see How CloudWatch works in the *Amazon CloudWatch User Guide*.
- Use the Trusted Advisor console or the AWS Support API to refresh your checks and get the latest check results. For more information, see Refresh check results.

To create a CloudWatch alarm for Trusted Advisor metrics

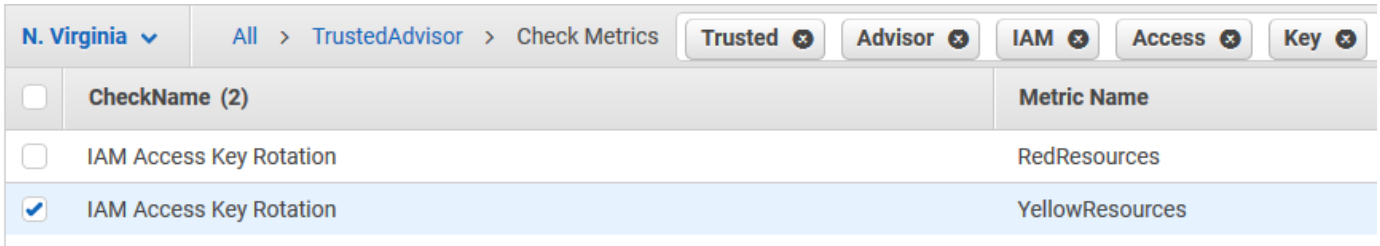
1. Open the CloudWatch console at https://eusc-de-east-1.console.amazonaws-eusc.eu/cloudwatch/.
2. Use the **Region selector** and choose the **US East (N. Virginia)** AWS Region.
3. In the navigation pane, choose **Alarms**.
4. Choose **Create alarm**.
5. Choose **Select metric**.
6. For **Metrics**, enter one or more dimension values to filter the metric list. For example, you can enter the metric name **ServiceLimitUsage** or the dimension, such as the Trusted Advisor check name.

   Tip

   - You can search for **`Trusted Advisor`** to list all metrics for the service.
   - For a list of metric and dimension names, see Trusted Advisor metrics and dimensions.

7. In the results table, select the check box for the metric.

   In the following example, the check name is **IAM Access Key Rotation** and the metric name is **YellowResources**.



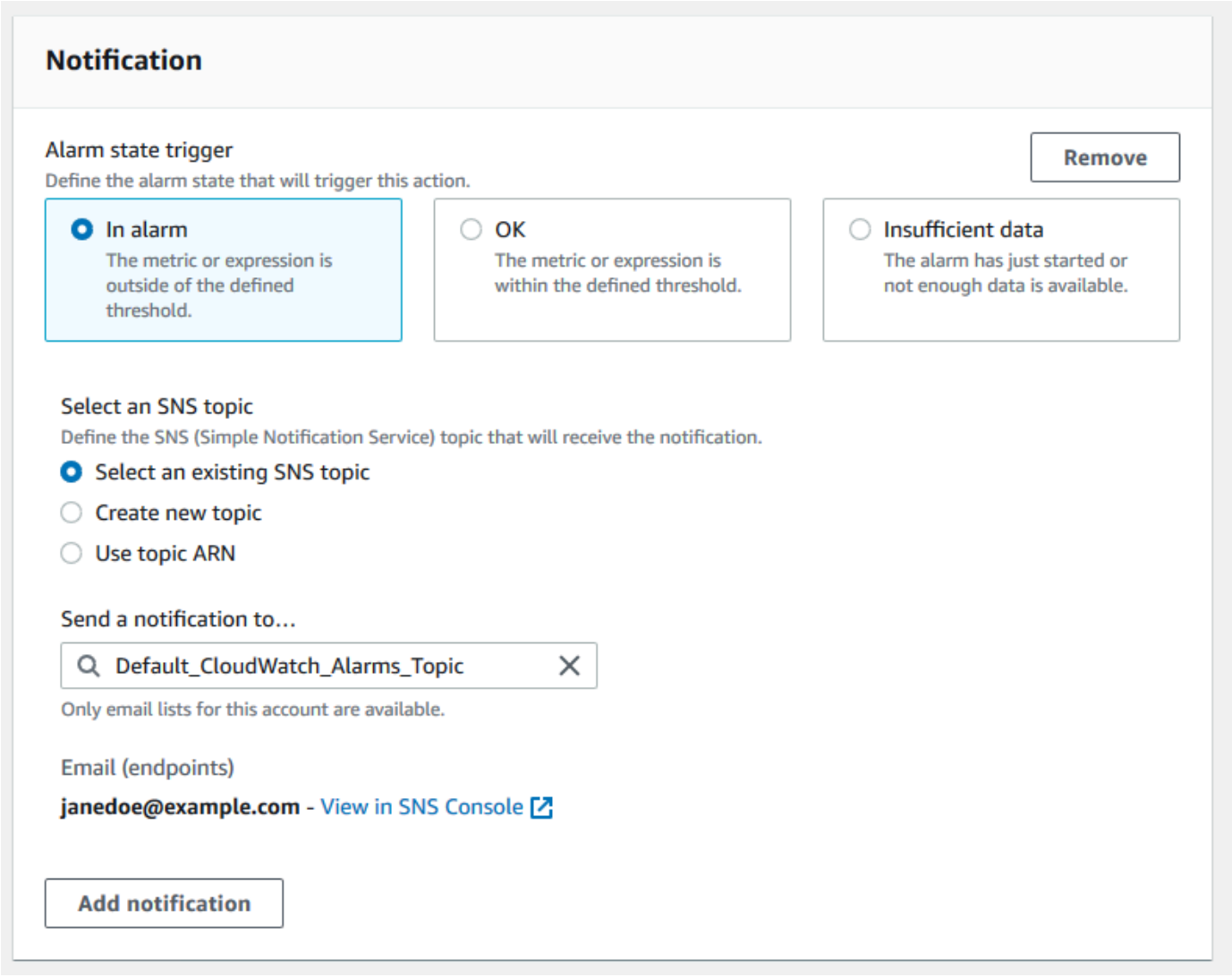
8. Choose **Select metric**.

9. On the **Specify metric and conditions** page, verify that the **Metric name** and **CheckName** that you chose appear on the page.
10. For **Period**, you can specify the time period that you want the alarm to start when the check status changes, such as 5 minutes.
11. Under **Conditions**, choose **Static**, and then specify the alarm condition for when the alarm should start.

    For example, if you choose **Greater/Equal >=threshold** and enter **1** for the threshold value, this means that the alarm starts when Trusted Advisor detects at least one IAM access key that hasn't been rotated in the last 90 days.

    Notes

    - For the **GreenChecks**, **RedChecks**, **YellowChecks**, **RedResources**, and **YellowResources** metrics, you can specify a threshold that is any whole number greater than or equal to zero.
    - Trusted Advisor doesn’t send metrics for **GreenResources**, which are resources for which Trusted Advisor hasn’t detected any issues.
12. Choose **Next**.
13. On the **Configure actions** page, for **Alarm state trigger**, choose **In alarm**.
14. For **Select an SNS topic**, choose an existing Amazon Simple Notification Service (Amazon SNS) topic or create one.



15. Choose **Next**.
16. For **Name and description**, enter a name and description for your alarm.
17. Choose **Next**.
18. On the **Preview and create** page, review your alarm details, and then choose **Create alarm**.

    When the status for the **IAM Access Key Rotation** check changes to red for 5 minutes, your alarm will send a notification to your SNS topic.

Example : Email notification for a CloudWatch alarm

The following email message shows that an alarm detected a change for the **IAM Access Key Rotation** check.

```
You are receiving this email because your Amazon CloudWatch Alarm
 "IAMAcessKeyRotationCheckAlarm" in the US East (N. Virginia) region has entered the
 ALARM state,
because "Threshold Crossed: 1 out of the last 1 datapoints [9.0 (26/03/21 22:44:00)]
 was greater than or equal to the threshold (1.0) (minimum 1 datapoint for OK -> ALARM
 transition)." at "Friday 26 March, 2021 22:49:42 UTC".

View this alarm in the AWS Management Console:
https://us-east-1.console.aws.amazon.com/cloudwatch/home?region=us-
east-1#s=Alarms&alarm=IAMAcessKeyRotationCheckAlarm

Alarm Details:
- Name:                       IAMAcessKeyRotationCheckAlarm
- Description:                This alarm starts when one or more AWS access keys in my
 AWS account have not been rotated in the last 90 days.
- State Change:               INSUFFICIENT_DATA -> ALARM
- Reason for State Change:    Threshold Crossed: 1 out of the last 1 datapoints [9.0
 (26/03/21 22:44:00)] was greater than or equal to the threshold (1.0) (minimum 1
 datapoint for OK -> ALARM transition).
- Timestamp:                  Friday 26 March, 2021 22:49:42 UTC
- AWS Account:                123456789012
- Alarm Arn:                  arn:aws:cloudwatch:us-
east-1:123456789012:alarm:IAMAcessKeyRotationCheckAlarm

Threshold:
- The alarm is in the ALARM state when the metric is GreaterThanOrEqualToThreshold 1.0
 for 300 seconds.

Monitored Metric:
- MetricNamespace:                    AWS/TrustedAdvisor
- MetricName:                         RedResources
- Dimensions:                         [CheckName = IAM Access Key Rotation]
- Period:                             300 seconds
- Statistic:                          Average
- Unit:                               not specified
- TreatMissingData:                   missing


State Change Actions:
- OK:
- ALARM: [arn:aws:sns:us-east-1:123456789012:Default_CloudWatch_Alarms_Topic]
- INSUFFICIENT_DATA:
```

CloudWatch metrics for Trusted Advisor

You can use the CloudWatch console or the AWS Command Line Interface (AWS CLI) to find the metrics available for Trusted Advisor.

For a list of the namespaces, metrics, and dimensions for all services that publish metrics, see AWS services that publish CloudWatch metrics in the *Amazon CloudWatch User Guide*.

View Trusted Advisor metrics (console)

You can sign in to the CloudWatch console and view the available metrics for Trusted Advisor.

To view available Trusted Advisor metrics (console)

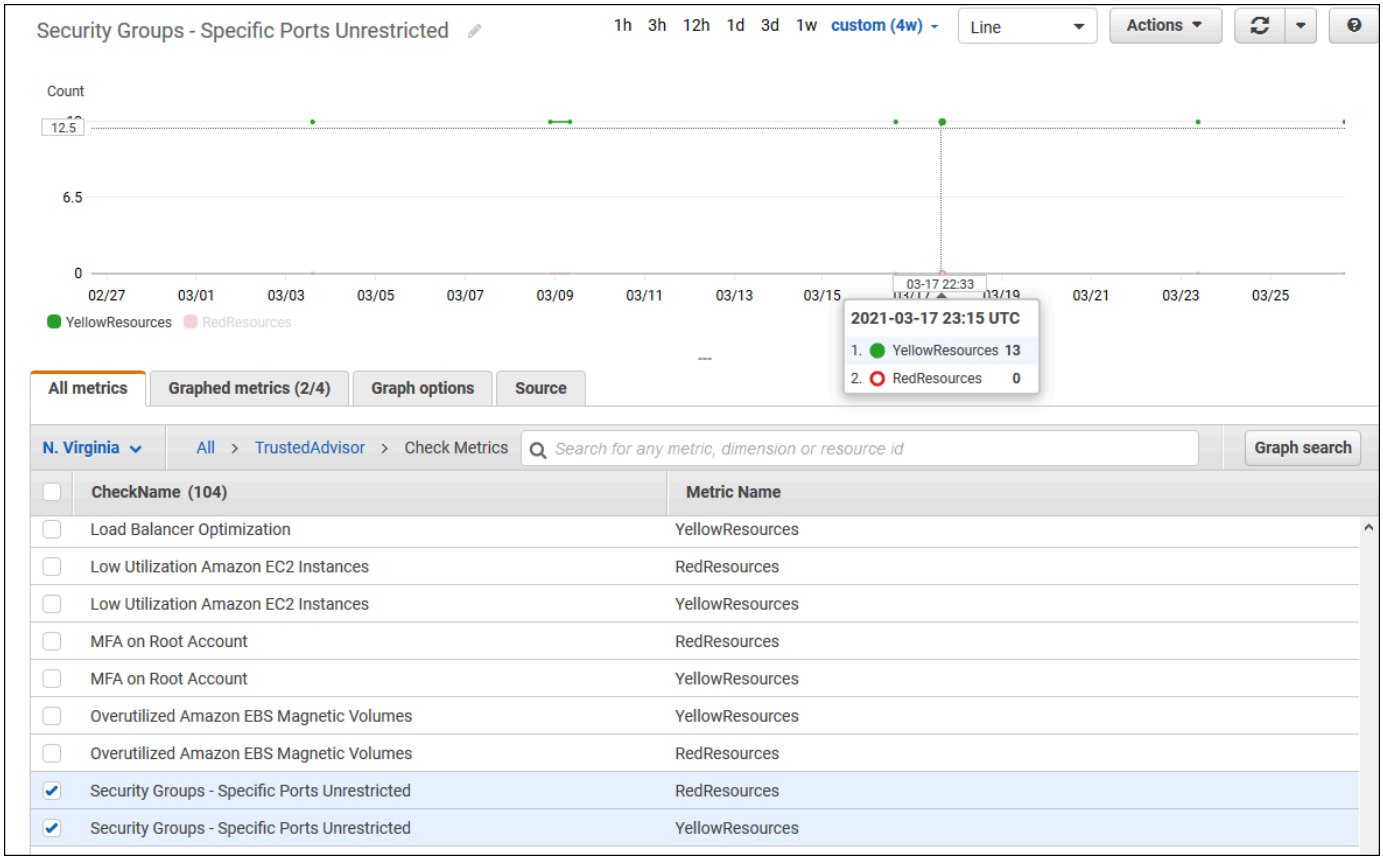
1. Open the CloudWatch console at https://eusc-de-east-1.console.amazonaws-eusc.eu/cloudwatch/.
2. Use the **Region selector** and choose the **US East (N. Virginia)** AWS Region.
3. In the navigation pane, choose **Metrics**.
4. Enter a metric namespace, such as **`TrustedAdvisor`**.
5. Choose a metric dimension, such as **Check Metrics**.
6. The **All metrics** tab shows metrics for that dimension in the namespace. You can do the following:

    a. To sort the table, choose the column heading.

    b. To graph a metric, select the check box next to the metric. To select all metrics, select the check box in the heading row of the table.

    c. To filter by metric, choose the metric name, and then choose **Add to search**.

    The following example shows the results for the **Security Groups - Specific Ports Unrestricted** check. The check identifies 13 resources that are yellow. Trusted Advisor recommends that you investigate checks that are yellow.



7. (Optional) To add this graph to a CloudWatch dashboard, choose **Actions**, and then choose **Add to dashboard**.

   For more information about creating a graph to view your metrics, see Graphing a metric in the *Amazon CloudWatch User Guide*.

View Trusted Advisor metrics (CLI)

Example : List all metrics for Trusted Advisor

The following example specifies the `AWS/TrustedAdvisor` namespace to view all metrics for Trusted Advisor.

```
aws cloudwatch list-metrics --namespace AWS/TrustedAdvisor
```

Your output might look like the following.

```
{
    "Metrics": [
        {
```

```
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "ServiceName",
                    "Value": "EBS"
                },
                {
                    "Name": "ServiceLimit",
                    "Value": "Magnetic (standard) volume storage (TiB)"
                },
                {
                    "Name": "Region",
                    "Value": "ap-northeast-2"
                }
            ],
            "MetricName": "ServiceLimitUsage"
        },
        {
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "CheckName",
                    "Value": "Overutilized Amazon EBS Magnetic Volumes"
                }
            ],
            "MetricName": "YellowResources"
        },
        {
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "ServiceName",
                    "Value": "EBS"
                },
                {
                    "Name": "ServiceLimit",
                    "Value": "Provisioned IOPS"
                },
                {
                    "Name": "Region",
                    "Value": "eu-west-1"
                }
            ],
            "MetricName": "ServiceLimitUsage"
```

```
        },
        {
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "ServiceName",
                    "Value": "EBS"
                },
                {
                    "Name": "ServiceLimit",
                    "Value": "Provisioned IOPS"
                },
                {
                    "Name": "Region",
                    "Value": "ap-south-1"
                }
            ],
            "MetricName": "ServiceLimitUsage"
        },
    ...
  ]
}
```

Example : List all metrics for a dimension

The following example specifies the `AWS/TrustedAdvisor` namespace and the `Region` dimension to view the metrics available for the specified AWS Region.

```
aws cloudwatch list-metrics --namespace AWS/TrustedAdvisor --dimensions
 Name=Region,Value=us-east-1
```

Your output might look like the following.

```
{
    "Metrics": [
        {
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "ServiceName",
                    "Value": "SES"
                },
                {
```

```
                    "Name": "ServiceLimit",
                    "Value": "Daily sending quota"
                },
                {
                    "Name": "Region",
                    "Value": "us-east-1"
                }
            ],
            "MetricName": "ServiceLimitUsage"
        },
        {
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "ServiceName",
                    "Value": "AutoScaling"
                },
                {
                    "Name": "ServiceLimit",
                    "Value": "Launch configurations"
                },
                {
                    "Name": "Region",
                    "Value": "us-east-1"
                }
            ],
            "MetricName": "ServiceLimitUsage"
        },
        {
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "ServiceName",
                    "Value": "CloudFormation"
                },
                {
                    "Name": "ServiceLimit",
                    "Value": "Stacks"
                },
                {
                    "Name": "Region",
                    "Value": "us-east-1"
                }
            ],
```

```
            "MetricName": "ServiceLimitUsage"
        },
    ...
  ]
}
```

Example : List metrics for a specific metric name

The following example specifies the `AWS/TrustedAdvisor` namespace and the `RedResources` metric name to view the results for only this specific metric.

```
aws cloudwatch list-metrics --namespace AWS/TrustedAdvisor --metric-name RedResources
```

Your output might look like the following.

```
{
    "Metrics": [
        {
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "CheckName",
                    "Value": "Amazon RDS Security Group Access Risk"
                }
            ],
            "MetricName": "RedResources"
        },
        {
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "CheckName",
                    "Value": "Exposed Access Keys"
                }
            ],
            "MetricName": "RedResources"
        },
        {
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "CheckName",
```

```
                    "Value": "Large Number of Rules in an EC2 Security Group"
                }
            ],
            "MetricName": "RedResources"
        },
        {
            "Namespace": "AWS/TrustedAdvisor",
            "Dimensions": [
                {
                    "Name": "CheckName",
                    "Value": "Auto Scaling Group Health Check"
                }
            ],
            "MetricName": "RedResources"
        },
    ...
  ]
}
```

Trusted Advisor metrics and dimensions

See the following tables for the Trusted Advisor metrics and dimensions that you can use for your CloudWatch alarms and graphs.

Trusted Advisor check-level metrics

You can use the following metrics for Trusted Advisor checks.

Metric	Description
`RedResources`	The number of resources that are in a red state (action recommended).
`YellowResources`	The number of resources that are in a yellow state (investigation recommended).

Trusted Advisor service quota-level metrics

You can use the following metrics for AWS service quotas.

Metric	Description
ServiceLimitUsage	The percentage of resource usage against a service quota (formerly referred to as limits).

Dimensions for check-level metrics

You can use the following dimension for Trusted Advisor checks.

Dimension	Description
CheckName	The name of the Trusted Advisor check. You can find all check names in the Trusted Advisor console or the AWS Trusted Advisor check reference.

Dimensions for service quota metrics

You can use the following dimensions for Trusted Advisor service quota metrics.

Dimension	Description
Region	The AWS Region for a service quota.
ServiceName	The name of the AWS service.
ServiceLimit	The name of the service quota. For more information about service quotas, see AWS service quotas in the *AWS General Reference*.

Logging AWS Trusted Advisor console actions with AWS CloudTrail

Trusted Advisor is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Trusted Advisor. CloudTrail captures actions for Trusted Advisor

as events. The calls captured include calls from the Trusted Advisor console. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon Simple Storage Service (Amazon S3) bucket, including events for Trusted Advisor. 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 Trusted Advisor, 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, including how to configure and enable it, see theAWS CloudTrail User Guide.

Trusted Advisor information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When supported event activity occurs in the Trusted Advisor console, 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, seeViewing Events with CloudTrail Event History.

For an ongoing record of events in your AWS account, including events for Trusted Advisor, 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

Trusted Advisor supports logging a subset of the Trusted Advisor console actions as events in CloudTrail log files. CloudTrail logs the following actions:

- `BatchUpdateRecommendationResourceExclusion`
- `CreateEngagement`
- `CreateEngagementAttachment`

- CreateEngagementCommunication
- CreateExcelReport
- DescribeAccount
- DescribeAccountAccess
- DescribeCheckItems
- DescribeCheckRefreshStatuses
- DescribeCheckSummaries
- DescribeChecks
- DescribeNotificationPreferences
- DescribeOrganization
- DescribeOrganizationAccounts
- DescribeReports
- DescribeServiceMetadata
- ExcludeCheckItems
- GenerateReport
- GetEngagement
- GetEngagementAttachment
- GetEngagementType
- GetExcelReport
- GetOrganizationRecommendation
- GetRecommendation
- IncludeCheckItems
- ListAccountsForParent
- ListChecks
- ListEngagementCommunications
- ListEngagementTypes
- ListEngagements
- ListOrganizationRecommendationAccounts
- ListOrganizationRecommendationResources
- ListOrganizationRecommendations

- ListOrganizationalUnitsForParent
- ListRecommendationResources
- ListRecommendations
- ListRoots
- RefreshCheck
- SetAccountAccess
- SetOrganizationAccess
- UpdateEngagement
- UpdateEngagementStatus
- UpdateNotificationPreferences
- UpdateOrganizationRecommendationLifecycle
- UpdateRecommendationLifecycle

For a complete list of Trusted Advisor console actions, seeTrusted Advisor actions.

> **Note**
>
> CloudTrail also logs the Trusted Advisor API operations in the AWS Support API Reference. For more information, seeLogging AWS Support API calls with AWS CloudTrail.

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 theCloudTrail userIdentity Element.

Example: Trusted Advisor 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.

Example : Log entry for RefreshCheck

The following example shows a CloudTrail log entry that demonstrates the `RefreshCheck` action for the Amazon S3 Bucket Versioning check (ID R365s2Qddf).


```
{
        "eventVersion":"1.04",
        "userIdentity":{
        "type":"IAMUser",
        "principalId":"AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/janedoe",
        "accountId":"123456789012",
        "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
        "userName":"janedoe",
        "sessionContext":{
        "attributes":{
        "mfaAuthenticated":"false",
        "creationDate":"2020-10-21T22:06:18Z"
        }
        }
        },
        "eventTime":"2020-10-21T22:06:33Z",
        "eventSource":"trustedadvisor.amazonaws.com",
        "eventName":"RefreshCheck",
        "awsRegion":"us-east-1",
        "sourceIPAddress":"100.127.34.136",
        "userAgent":"signin.amazonaws.com",
        "requestParameters":{
        "checkId":"R365s2Qddf"
        },
        "responseElements":{
        "status":{
        "checkId":"R365s2Qddf",
        "status":"enqueued",
        "millisUntilNextRefreshable":3599993
        }
        },
        "requestID":"d23ec729-8995-494c-8054-dedeaEXAMPLE",
        "eventID":"a49d5202-560f-4a4e-b38a-02f1cEXAMPLE",
```

```
        "eventType":"AwsApiCall",
        "recipientAccountId":"123456789012"
        }
```

Example : Log entry for UpdateNotificationPreferences

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

```
{
        "eventVersion":"1.04",
        "userIdentity":{
        "type":"IAMUser",
        "principalId":"AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/janedoe",
        "accountId":"123456789012",
        "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
        "userName":"janedoe",
        "sessionContext":{
        "attributes":{
        "mfaAuthenticated":"false",
        "creationDate":"2020-10-21T22:06:18Z"
        }
        }
        },
        "eventTime":"2020-10-21T22:09:49Z",
        "eventSource":"trustedadvisor.amazonaws.com",
        "eventName":"UpdateNotificationPreferences",
        "awsRegion":"us-east-1",
        "sourceIPAddress":"100.127.34.167",
        "userAgent":"signin.amazonaws.com",
        "requestParameters":{
        "contacts":[
        {
        "id":"billing",
        "type":"email",
        "active":false
        },
        {
        "id":"operational",
        "type":"email",
        "active":false
```

```
        },
        {
        "id":"security",
        "type":"email",
        "active":false
        }
        ],
        "language":"en"
        },
        "responseElements":null,
        "requestID":"695295f3-c81c-486e-9404-fa148EXAMPLE",
        "eventID":"5f923d8c-d210-4037-bd32-997c6EXAMPLE",
        "eventType":"AwsApiCall",
        "recipientAccountId":"123456789012"
        }
```


Example : Log entry for GenerateReport

The following example shows a CloudTrail log entry that demonstrates the `GenerateReport` action. This action creates a report for your AWS organization.


```
{
        "eventVersion":"1.04",
        "userIdentity":{
        "type":"IAMUser",
        "principalId":"AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/janedoe",
        "accountId":"123456789012",
        "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
        "userName":"janedoe",
        "sessionContext":{
        "attributes":{
        "mfaAuthenticated":"false",
        "creationDate":"2020-11-03T13:03:10Z"
        }
        }
        },
        "eventTime":"2020-11-03T13:04:29Z",
        "eventSource":"trustedadvisor.amazonaws.com",
        "eventName":"GenerateReport",
        "awsRegion":"us-east-1",
        "sourceIPAddress":"100.127.36.171",
```

```
        "userAgent":"signin.amazonaws.com",
        "requestParameters":{
        "refresh":false,
        "includeSuppressedResources":false,
        "language":"en",
        "format":"JSON",
        "name":"organizational-view-report",
        "preference":{
        "accounts":[

        ],
        "organizationalUnitIds":[
        "r-j134"
        ],
        "preferenceName":"organizational-view-report",
        "format":"json",
        "language":"en"
        }
        },
        "responseElements":{
        "status":"ENQUEUED"
        },
        "requestID":"bb866dc1-60af-47fd-a660-21498EXAMPLE",
        "eventID":"2606c89d-c107-47bd-a7c6-ec92fEXAMPLE",
        "eventType":"AwsApiCall",
        "recipientAccountId":"123456789012"
        }
```

Troubleshooting resources

For Amazon EC2 Windows instances, you can use EC2Rescue to examine your instances to help identify common problems, collect log files, and help Support troubleshoot your issues. You can also use EC2Rescue to analyze boot volumes from non-functional instances. For more information, see How can I use EC2Rescue to troubleshoot and fix common issues on my EC2 Windows instance?

Service-specific troubleshooting

Most AWS service documentation contains troubleshooting topics that can get you started before contacting AWS Support. The following table provides links to troubleshooting topics, arranged by service.

> **Note**
>
> The following table provides a list of the most common services. To search for other troubleshooting topics, use the search text box on the AWS Documentation landing page.

Service	Link
Amazon Web Services	Troubleshooting AWS Signature Version 4 errors
Amazon API Gateway	Troubleshooting issues with HTTP APIs
Amazon AppStream	Troubleshoot Amazon AppStream
Amazon Athena	Troubleshoot in Athena
Amazon Aurora MySQL	Troubleshoot for Amazon Aurora
Amazon Aurora PostgreSQL	Troubleshoot for Amazon Aurora
Amazon EC2 Auto Scaling	Troubleshooting Auto Scaling
AWS Certificate Manager (ACM)	Troubleshooting

Service	Link
AWS CloudFormation	Troubleshooting AWS CloudFormation
Amazon CloudFront	Troubleshooting | Troubleshooting RTMP distributions
AWS CloudHSM	Troubleshooting
Amazon CloudSearch	Troubleshooting Amazon CloudSearch
AWS CodeDeploy	Troubleshooting AWS CodeDeploy
Amazon CloudWatch	Troubleshooting
AWS Database Migration Service	Troubleshooting migration tasks in AWS Database Migration Service
AWS Data Pipeline	Troubleshooting
AWS Direct Connect	Troubleshooting AWS Direct Connect
AWS Directory Service	Troubleshooting AWS Directory Service administration issues
Amazon DynamoDB	Troubleshooting | Troubleshooting SSL/TLS connection establishment issues
AWS Elastic Beanstalk	Troubleshooting
Amazon Elastic Compute Cloud (Amazon EC2)	Troubleshooting instances | Troubleshooting Windows instances | Troubleshooting VM Import/Export | Troubleshooting API request errors
Amazon Elastic Container Service (Amazon ECS)	Amazon ECS troubleshooting
Amazon Elastic Kubernetes Service (Amazon EKS)	Amazon EKS troubleshooting
Elastic Load Balancing	Troubleshoot your application load balancers | Troubleshoot your Classic Load Balancer

Service	Link
Amazon ElastiCache (Memcached)	Troubleshooting applications
Amazon ElastiCache (Redis OSS)	Troubleshooting applications
Amazon EMR	Troubleshoot a cluster
AWS Flow Framework	Troubleshooting and debugging tips
AWS Glue	Troubleshooting AWS Glue
AWS Glue DataBrew	Troubleshooting identity and access in AWS Glue DataBrew
AWS GovCloud (US)	Troubleshooting
AWS Identity and Access Management (IAM)	Troubleshooting IAM
Amazon Keyspaces (for Apache Cassandra)	Troubleshooting Amazon Keyspaces (for Apache Cassandra)
Amazon Kinesis Data Streams	Troubleshooting Amazon Kinesis Data Streams producers | Troubleshooting Amazon Kinesis Data Streams consumers
Amazon Managed Service for Apache Flink	Troubleshooting Performance | Troubleshooting Amazon Managed Service for Apache Flink for SQL Applications
Amazon Data Firehose	Troubleshooting Amazon Data Firehose
AWS Lambda	Troubleshooting and monitoring AWS Lambda functions with CloudWatch
Amazon OpenSearch Service	Troubleshooting Amazon OpenSearch Service
Amazon Personalize	Troubleshooting
Amazon Quick Suite	Troubleshooting Amazon Quick Suite | Troubleshooting skipped row errors

Service	Link
AWS Resource Access Manager (AWS RAM)	Troubleshooting issues with AWS RAM
Amazon Redshift	Troubleshooting queries | Troubleshooting data loads | Troubleshooting connection issues in Amazon Redshift | Troubleshooting Amazon Redshift audit logging | Troubleshooting queries in Amazon Redshift Spectrum
Amazon Relational Database Service (Amazon RDS)	Troubleshooting | Troubleshooting applications on Amazon RDS | Troubleshooting DB issues for Amazon RDS Custom
Amazon Route 53	Troubleshooting Amazon Route 53
Amazon SageMaker AI	Troubleshoot errors | Troubleshooting Amazon SageMaker AI Studio
Amazon Silk	Troubleshooting
Amazon Simple Email Service (Amazon SES)	Troubleshooting Amazon SES
Amazon Simple Storage Service (Amazon S3)	Troubleshooting
Amazon Simple Workflow Service (Amazon SWF)	AWS flow framework for Java: Troubleshooting and debugging tips | AWS flow framework for Ruby: Troubleshooting and debugging workflows
AWS Storage Gateway	Troubleshooting your gateway
AWS Systems Manager	Troubleshooting SSM Agent
Amazon Virtual Private Cloud (Amazon VPC)	Troubleshooting
AWS Virtual Private Network (Site-to-Site VPN)	Troubleshooting your customer gateway device

Service	Link
AWS WAF	Testing and tuning your AWS WAF protections
Amazon WorkMail	Troubleshooting the Amazon WorkMail web application
Amazon WorkSpaces	Troubleshooting Amazon WorkSpaces issues | Troubleshooting Amazon WorkSpaces client issues

Document history

The following table describes the important changes to the documentation since the last release of the AWS Support service.

- **AWS Support API version:** 2013-04-15
- **AWS Support App API version:** 2021-08-20

The following table describes important updates to the AWS Support and AWS Trusted Advisor documentation, beginning May 10, 2021. You can subscribe to the RSS feed to receive notifications about the updates.

Change	Description	Date
New section **Set up permissions to use AI-enhanced troubleshooting**	Added a new section outlining how to configure the required permissions for AI-enhanced troubleshooting in Service Catalog. For details, see the Set up permissions to use AI-enhanced troubleshooting .	December 22, 2025
Updated Trusted Advisor check reference	For details, see the Change log for AWS Trusted Advisor checks.	December 18, 2025
Updated Trusted Advisor check reference	For details, see the Change log for AWS Trusted Advisor checks.	December 17, 2025
Added information regarding virtual meetings with AWS Support	For more information, see Virtual meetings with AWS Support.	December 9, 2025
Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services	December 8, 2025

	for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	
Added information regarding new AWS Support plans	For more information, see AWS Support Plans.	December 2, 2025
Added information regarding AI-enhanced troubleshooting	For more information, see AI-enhanced troubleshooting in Support Center Console.	December 2, 2025
Added information a new section for AWS Unified Operations	For more information, see What is AWS Unified Operations.	December 2, 2025
Added description for Support Assistant APIs	Added AWS Support API Actions for Support Assistant: ListInteractions, ListInteractionEntries, and ResolveInteraction in Manage access to AWS Support Center.	December 2, 2025
Updated Trusted Advisor check reference	For details, see the Change log for AWS Trusted Advisor checks.	November 21, 2025
Updated Trusted Advisor check reference	For details, see the Change log for AWS Trusted Advisor checks.	November 17, 2025

Updated **Editing and deleting a service-linked role for Support**	Customers onboarded to AWS Organizations with an Enterprise Support plan can delete the `AWSServiceRoleForSupport` service-linked role. For more information, see Editing and deleting a service-linked role for Support.	October 31, 2025
Updated Trusted Advisor check reference	For details, see the Change log for AWS Trusted Advisor checks.	October 15, 2025
Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	September 30, 2025
New topic	Added a new topic with information on how to enable promotional plan expiration notifications. For more information, see Configure promotional plan expiration notifications.	September 12, 2025
Updated Trusted Advisor check	For details, see the Change log for AWS Trusted Advisor checks.	September 11, 2025

Updated **Testing Support Center Console API calls**	Step 3 has been updated to indicate the correct event source, **support-console.amazonaws.com**. For more information see Testing Support Center Console API calls.	September 2, 2025
Updated **Changing AWS Support plans**	The steps to upgrade or downgrade your AWS Support subscription have been updated. For more information see Changing AWS Support plans.	August 27, 2025
Updated Trusted Advisor check	For details, see the Change log for AWS Trusted Advisor checks.	August 19, 2025
Removed Trusted Advisor Engage section	Removed **Get started with AWS Trusted Advisor Engage (Preview) section**.	August 7, 2025
Updated how to add the AWS Support App to a Slack channel	For details, see Configuring a Slack channel.	August 6, 2025
Updated managed policies: AWSSupportAccess	For details, see AWS Support updates to AWS managed policies.	July 18, 2025
Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	July 15, 2025

Updated check: Amazon EC2 instances with Ubuntu LTS end of standard support	For details, see the Change log for AWS Trusted Advisor checks.	July 3, 2025
Updated Trusted Advisor check: Amazon S3 Bucket Permissions	For details, see the Change log for AWS Trusted Advisor checks.	July 3, 2025
Updated check: MFA on root account	For details, see the Change log for AWS Trusted Advisor checks.	July 2, 2025
Updated check: Amazon ECS AWSLogs driver in blocking mode.	For details, see the Change log for AWS Trusted Advisor checks.	July 2, 2025
Updated **Request a service quota increase**	Added information on how to create a service quota increase request if your AWS service or AWS Region isn't supported in the Service Quotas console. For more information, see Request service quota increase.	July 2, 2025
Creating a service quota increase request in the Support Center Console is no longer supported.	For more information, see Creating a service quota increase.	June 23, 2025
Added description for UpdateInteraction in Support API	Added AWS Support API UpdateInteraction Actions description in Manage access to AWS Support Center.	June 23, 2025

Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	June 17, 2025
New section: About the Support Center Console API	The Support Center Console API enhances your experience with the Support Center Console. For details, see About the Support Center Console API.	June 16, 2025
Updated AWS Trusted Advisor chapter introduction to reflect that Basic and Developer Support plans don't support automatic check refresh. You must manually refresh Security checks to see the most recent check status.	For details, see AWS Trusted Advisor.	June 11, 2025
Updated check: AWS STS global endpoint usage across AWS Regions is now available at all AWS Support plan tiers.	For details, see the Change log for AWS Trusted Advisor checks.	June 9, 2025
New check: Amazon Aurora cost optimization recommendations for DB cluster storage	For details, see the Change log for AWS Trusted Advisor checks.	June 9, 2025
New check: AWS STS global endpoint usage across AWS Regions	For details, see the Change log for AWS Trusted Advisor checks.	June 2, 2025

15 new AWS Cost Optimization Hub checks add to Trusted Advisor	For details, see the Change log for AWS Trusted Advisor checks.	May 30, 2025
Updated three Trusted Advisor checks	For details, see the Change log for AWS Trusted Advisor checks.	May 21, 2025
New feature: Update support case severity	For details, see the *Changing a support case severity level* section Creating support cases and case management.	May 21, 2025
Updated time you can view AWS Support case details.	For details, see Security for your AWS Support cases.	April 29, 2025
Updated two Trusted Advisor checks	For details, see the Change log for AWS Trusted Advisor checks.	April 2, 2025
Added description for Support API	Added AWS Support API Actions description in Manage access to AWS Support Center.	March 7, 2025
Deprecated 6 AWS Security Hub CSPM checks	For details, see the Change log for AWS Trusted Advisor checks.	March 5, 2025
Removed references to category-level metrics for Trusted Advisor	Category-level metrics for Trusted Advisor are deprecated. References to category-level metrics are removed from Creating Amazon CloudWatch alarms to monitor AWS Trusted Advisor metrics.	January 27, 2025

Updated documentation for Trusted Advisor	Added two new checks: AWS CloudTrail Management Events Logging and Amazon RDS Continuous Backups Not Enabled. For more informati on, see Change log for AWS Trusted Advisor checks.	December 23, 2024
Updated documentation for Trusted Advisor	Updated Auto Scaling Group Resources. For more informati on, see Change log for AWS Trusted Advisor checks.	December 23, 2024
Updated documentation for Trusted Advisor	Updated IAM Access Analyzer External Access check. For more information, see Change log for AWS Trusted Advisor checks.	December 23, 2024
Updated documentation for `AWSSupportServiceR olePolicy`	Added new permissions to provide billing, administr ative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSuppor tServiceRolePolicy.	November 25, 2024
Updated documentation for Trusted Advisor	Added 1 new Trusted Advisor check. For more informati on, see Change log for AWS Trusted Advisor checks.	November 22, 2024

Added documentation for AWS managed policies for AWS Partner-Led Support	Added documentation for a new AWS managed policy `AWSPartnerLedSupportReadOnlyAccess`. For more information, see AWS managed policies for AWS Partner-Led Support.	November 22, 2024
Updated documentation for Trusted Advisor	Updated 3 Trusted Advisor checks. For more information, see Change log for AWS Trusted Advisor checks.	November 7, 2024
Updated documentation for AWS Support Plans	Added a new log example for the `ListSupportPlanModifiers` operation to the Logging Support Plans API calls with AWS CloudTrail page.	November 6, 2024
Updated documentation for `AWSTrustedAdvisorServiceRolePolicy`	Added new IAM actions `elasticloadbalancing:DescribeListeners` and `elasticloadbalancing:DescribeRules`, to onboard a new security check. For more information, see AWS managed policy: AWSTrustedAdvisorServiceRolePolicy.	October 30, 2024
Updated documentation for Trusted Advisor	Added 4 new Trusted Advisor checks. For more information, see Change log for AWS Trusted Advisor checks.	October 11, 2024

Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	October 8, 2024
Updated documentation for Trusted Advisor	Moved 1 Cost Optimization check under the Fault Tolerance pillar. Updated 1 Security check and 1 Fault Tolerance check. For more information, see Change log for AWS Trusted Advisor checks.	October 2, 2024
Updated AWS Trusted Advisor Engage section	Updated the AWS Trusted Advisor Engage section to reference AWS Countdown. For more information, see Get started with AWS Trusted Advisor Engage (Preview).	September 16, 2024
Updated documentation for AWS Support Plans	Added a new permission and CloudTrail documentation for viewing a list of support plan modifiers. For more information, see Manage access to AWS Support Plans, AWS managed policies for AWS Support Plans and Logging AWS Support Plans API calls with AWS CloudTrail.	September 9, 2024

Updated documentation for Trusted Advisor	Trusted Advisor added 9 new checks on Aug 23rd. For more information, see Change log for AWS Trusted Advisor checks.	August 23, 2024
Updated documentation for Trusted Advisor	Updated 1 Trusted Advisor Operational Excellence check and added 1 new Trusted Advisor Security check. For more information, see Change log for AWS Trusted Advisor checks.	August 22, 2024
Updated documentation for Trusted Advisor	Updated 6 Trusted Advisor Security checks. For more information, see Change log for AWS Trusted Advisor checks.	August 20, 2024
Updated documentation for Trusted Advisor	Updated 2 Trusted Advisor checks. For more informati on, see Change log for AWS Trusted Advisor checks.	August 12, 2024
Updated documentation for `AWSSupportServiceR olePolicy`	Added new permissions to provide billing, administr ative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSuppor tServiceRolePolicy.	August 5, 2024
Updated documentation for Trusted Advisor	Updated 9 Trusted Advisor Checks. For more informati on, see Change log for AWS Trusted Advisor checks.	July 21, 2024

Updated documentation for AWSTrustedAdvisorServiceRolePolicy	Added new IAM actions `access-analyzer:ListAnalyzers`, `cloudwatch:ListMetrics`, `dax:DescribeClusters`, `ec2:DescribeNatGateways`, `ec2:DescribeRouteTables`, `ec2:DescribeVpcEndpoints`, `ec2:GetManagedPrefixListEntries`, `elasticloadbalancing:DescribeTargetHealth`, `iam:ListSAMLProviders`, `kafka:DescribeClusterV2` `network-firewall:ListFirewalls` `network-firewall:DescribeFirewall` and `sqs:GetQueueAttributes` to onboard new checks. For more information, see AWS managed policy: AWSTrustedAdvisorServiceRolePolicy.	June 11, 2024
Removed 5 AWS Trusted Advisor checks from documentation	Removed 5 AWS Trusted Advisor checks that are now deprecated. For more information, see Change log for AWS Trusted Advisor checks.	May 15, 2024

Added 1 new AWS Trusted Advisor Security check to documentation	Added 1 new AWS Trusted Advisor Security check to documentation. For more information, see Change log for AWS Trusted Advisor checks.	May 15, 2024
Removed 3 Fault Tolerance checks from documentation	Removed 3 Fault Tolerance checks that are now deprecated. For more information, see Change log for AWS Trusted Advisor checks.	April 25, 2024
Updated Fault Tolerance and Security check documentation	Added 1 new fault tolerance check. Updated 1 fault tolerance and 1 security check. For more informati on, see Change log for AWS Trusted Advisor checks.	March 29, 2024
Updated documentation for `AWSSupportServiceR olePolicy`	Added new permissions to provide billing, administr ative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSuppor tServiceRolePolicy.	March 22, 2024
Updated documentation for Support plan	Updates to the Features of Support Plans. For more information, see Support plans.	March 11, 2024

Updated documentation for Trusted Advisor	Added 1 fault tolerance check. For more informati on, see Change log for AWS Trusted Advisor checks.	February 29, 2024
Updated documentation for Trusted Advisor	Added 1 fault tolerance check. For more informati on, see Change log for AWS Trusted Advisor checks.	January 31, 2024
Updated documentation for AWSTrustedAdvisorS erviceRolePolicy	Added new IAM actions `cloudtrail:GetTrai l` , `cloudtrail:ListTra ils` , `cloudtrai l:GetEventSelectors` , `outposts:GetOutpost` , `outposts:ListAssets` and `outposts:ListOutpo sts` to onboard new checks. For more information, see AWS managed policy: AWSTrustedAdvisorServiceRol ePolicy.	January 18, 2024
Updated documentation for AWSSupportServiceR olePolicy	Added new permissions to provide billing, administr ative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSuppor tServiceRolePolicy.	January 17, 2024

Updated documentation for Trusted Advisor	Updated 1 fault tolerance check to amend title and description. For more information, see Change log for AWS Trusted Advisor checks.	January 8, 2024
Updated documentation for Trusted Advisor	Updated 1 security check to reflect change in deprecati on period. For more informati on, see Change log for AWS Trusted Advisor checks.	December 21, 2023
Updated documentation for Trusted Advisor	Added 2 security checks and 2 performance checks. For more information, see Change log for AWS Trusted Advisor checks.	December 20, 2023
Updated documentation for Trusted Advisor	Added 1 security check. For more information, see Change log for AWS Trusted Advisor checks.	December 15, 2023
Updated documentation for Trusted Advisor Engage	Updated Trusted Advisor Engage documentation with changes for email notification option.	December 14, 2023
Updated documentation for Trusted Advisor Engage	Updated Trusted Advisor Engage documentation with changes for scheduled engagements.	December 11, 2023

Updated documentation for Trusted Advisor	Added 2 new fault tolerance checks and 1 cost optimization check. For more information, see Change log for AWS Trusted Advisor checks.	December 7, 2023
Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	December 6, 2023
Updated AWS managed policies for Trusted Advisor	Updated the `AWSTrustedAdvisorPriorityFullAccess` and `AWSTrustedAdvisorPriorityReadOnlyAccess` AWS managed policies to include statement IDs. For more information, see AWS managed policies for AWS Trusted Advisor.	December 6, 2023
Updated documentation for Trusted Advisor	Added 3 new fault tolerance checks. For more information, see Change log for AWS Trusted Advisor checks.	November 17, 2023
Updated documentation for Trusted Advisor	Added 37 new checks for Amazon RDS. For more information, see Change log for AWS Trusted Advisor checks.	November 15, 2023

Updated documentation for AWSTrustedAdvisorServiceRolePolicy	Added new IAM actions `ec2:DescribeRegions` , `s3:GetLifecycleConfiguration` , `ecs:DescribeTaskDefinition` and `ecs:ListTaskDefinitions` to onboard new checks. For more information, see AWS managed policy: AWSTrustedAdvisorServiceRolePolicy.	November 9, 2023
Updated documentation for AWSSupportServiceRolePolicy	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	October 27, 2023
Updated documentation for Trusted Advisor	Added 64 new checks integrated from AWS Config. For more information, see Change log for AWS Trusted Advisor checks.	October 26, 2023
Updated documentation for Trusted Advisor	Added six new fault tolerance checks in Trusted Advisor. For more information, see the Change log for AWS Trusted Advisor checks.	October 12, 2023

Updated documentation for AWSTrustedAdvisorServiceRolePolicy	Added new IAM actions `route53resolver:ListResolverEndpoints` , `route53resolver:ListResolverEndpointIpAddresses` , `ec2:DescribeSubnets` , `kafka:ListClustersV2` and `kafka:ListNodes` to onboard new resilience checks. For more information, see AWS managed policy: AWSTrustedAdvisorServiceRolePolicy.	September 14, 2023
Updated documentation for AWSSupportServiceRolePolicy	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	August 28, 2023
Updated documentation for Trusted Advisor	Added 1 new service limits checks for AWS Lambda. For more information, see the Change log for AWS Trusted Advisor checks .	August 17, 2023
Updated documentation for Trusted Advisor	Added 1 new fault tolerance checks for Lambda. For more information, see the Change log for AWS Trusted Advisor checks.	August 3, 2023

Updated documentation for Trusted Advisor Engage	Updated Trusted Advisor Engage documentation with changes to forms for creating and editing engagements. Added page with Example Service Control Policies for AWS Trusted Advisor.	July 27, 2023
Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	June 26, 2023
Updated documentation for Trusted Advisor	Added two new fault tolerance checks for Amazon MQ. Added one new fault tolerance check and one new performance check for Amazon Elastic File System. For more information, see the Change log for AWS Trusted Advisor checks.	June 1, 2023
Updated documentation for Trusted Advisor	Added two new fault tolerance checks for NAT Gateway. For more information, see the Change log for AWS Trusted Advisor checks.	May 16, 2023

Updated documentation for AWS Support Plans	Added a new permission and CloudTrail documentation for the creation of support plan schedules. For more informati on, see Manage access to AWS Support Plans, AWS managed policies for AWS Support Plans and Logging AWS Support Plans API calls with AWS CloudTrail.	May 8, 2023
Updated documentation for `AWSSupportServiceR olePolicy`	Added new permissions to provide billing, administr ative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSuppor tServiceRolePolicy.	May 2, 2023
Updated documentation for Trusted Advisor Engage and Trusted Advisor Priority	Clarified prerequisites for Trusted Advisor Engage and Trusted Advisor Priority. Added example IAM policy with ability to use Trusted Advisor Engage and to enable trusted access to Trusted Advisor.	April 28, 2023
Updated documentation for Trusted Advisor	Added two new fault tolerance checks for AWS Resilience Hub and Incident Manager. For more informati on, see the Change log for AWS Trusted Advisor checks.	April 27, 2023

Added documentation for Trusted Advisor Engage	You can use AWS Trusted Advisor Engage to get the most out of your AWS Support Plans by making it easy for you to see, request and track all your proactive engagements, and communicate with your AWS account team about ongoing engagements. For more information, see Get started with AWS Trusted Advisor Engage.	April 6, 2023
Updated documentation for Trusted Advisor	Added two new fault tolerance checks for Amazon ECS. For more information, see the Change log for AWS Trusted Advisor checks.	March 30, 2023
Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	March 16, 2023

Added documentation for Trusted Advisor Priority	Updated the Trusted Advisor Priority console: • The **Acknowledge** and **Dismiss** buttons have replaced the **Accept** and **Reject** buttons. • You don't need to enter your job title or name to acknowledge, resolve, dismiss, or reopen recommendations. For more information, see Getting started with Trusted Advisor Priority.	February 16, 2023
Updated code examples for Support	Added .NET, Java, and Kotlin code examples that show how to use Support with an AWS software development kit (SDK). For more informati on, see Code examples for Support using AWS SDKs.	January 16, 2023
Updated documentation for `AWSSupportServiceR olePolicy`	Added new permissions to provide billing, administr ative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupport ServiceRolePolicy.	January 10, 2023

Updated documentation for AWS Support App	You can search for support cases in Slack by using filter options or searching by case ID. For more information, see Searching for support cases in Slack.	December 29, 2022
Updated documentation for AWS Support App	You can also use Terraform to create your resources for the AWS Support App. For more information, see Create AWS Support App resources by using Terraform.	December 22, 2022
Updated documentation for Trusted Advisor	Added three new fault tolerance checks for Amazon MemoryDB, Amazon ElastiCache, and AWS CloudHSM. For more information, see the Change log for AWS Trusted Advisor checks.	December 15, 2022
Updated documentation for the AWS Support App in Slack	You can now request live chat support for the following options: • Account and billing support cases. • Japanese language support for technical support cases. • For more information, see Creating support cases in a Slack channel.	December 14, 2022

Updated documentation for AWS Support	Added documentation about new endpoints for the Support API. For more information, see About the AWS Support API.	December 14, 2022
Added documentation for CloudFormation templates to use for the AWS Support App in Slack	You can use CloudFormation templates to create Slack configuration workspaces and channels for AWS accounts in AWS Organizations. For more information, see Creating AWS Support App resources with AWS CloudFormation.	December 5, 2022
Updated documentation for Trusted Advisor	Added two new fault tolerance checks for AWS Resilience Hub. For more information, see the Change log for AWS Trusted Advisor checks.	November 17, 2022
Added documentation for your AWS Security Hub CSPM findings in Trusted Advisor	Your findings from Security Hub CSPM controls are removed from Trusted Advisor faster. For more information, see the Change log for AWS Trusted Advisor checks.	November 17, 2022
Updated documentation for AWS Trusted Advisor	Added documentation for Trusted Advisor Recommend ations. For more information, see the Change log for AWS Trusted Advisor checks.	November 16, 2022

Updated documentation for the AWS Support App in Slack	Added documentation for Japanese language support. For more information, see Creating support cases in a Slack channel.	November 11, 2022
Updated documentation for AWS Support Plans	Added troubleshooting information to allow Support Plans access in an organizat ion. For more information, see Troubleshooting.	November 9, 2022
Updated documentation for the AWS Support App in Slack	Added documentation for `supportapp` permissions. For more information, see Permissions required for the AWS Support App to connect to Slack.	November 1, 2022
Updated documentation for the AWS Support App in Slack	You can use the `RegisterSlackWorkspaceForOrganization` API operation to register a Slack workspace for your AWS account. To call this API, your account must be part of an organization in AWS Organizations. For more information, see the AWS Support App in Slack API Reference.	October 19, 2022

Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	October 4, 2022
Updated documentation for Support Plans	You can now use AWS Identity and Access Management (IAM) to manage permissions to change the support plan for your AWS account. For more information, see the following topics: • Managing access for AWS Support Plans • AWS managed policies for AWS Support Plans • Changing AWS Support Plans • Logging AWS Support Plans API calls with AWS CloudTrail	September 29, 2022
Updated documentation for the AWS Support App in Slack	Added documentation on how to configure a public or private channel to use with the AWS Support App. For more information, see Configuring a Slack channel.	September 22, 2022

Updated documentation for AWS Support	Added a new section about security for your support cases. For more information, see Security for your AWS Support cases.	September 9, 2022
Updated documentation for Trusted Advisor	Added a new security check for Amazon EC2. For more information, see the Change log for AWS Trusted Advisor checks.	September 1, 2022
Updated documentation for the AWS Support App in Slack	See the following topics: You can use the AWS Support App to manage your support cases, request service quota increases, and chat with support agents directly in your Slack channels. For more information, see the AWS Support App in Slack documentation. You can attach AWS managed policies to your IAM roles to use the AWS Support App. For more information, see AWS managed policies for AWS Support App in Slack. New API reference for the AWS Support App. See the AWS Support App API Reference.	August 24, 2022

Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	August 17, 2022
Added documentation for Trusted Advisor Priority	Trusted Advisor Priority adds support for the following features: • Delegated administrators • Daily and weekly email notifications for recommendation summaries • Reopen resolved or rejected recommendations • AWS managed policies For more information, see Getting started with Trusted Advisor Priority.	August 17, 2022
Updated documentation for Trusted Advisor	The **Preferences** page in the Trusted Advisor console has been updated. For more information, see Getting started with AWS Trusted Advisor.	July 15, 2022

Updated documentation for Trusted Advisor	Updated the checks to include the following information: • **Alert Criteria** • **Recommended Action** • **Additional Resources** • **Report columns** For more information, see the AWS Trusted Advisor check reference.	July 7, 2022
Updated documentation for AWS Support	Added documentation that explains how to manage your support cases. • Updating an existing support case • Troubleshooting	June 28, 2022
Updated documentation for `AWSSupportServiceRolePolicy`	Updated permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	June 23, 2022

Updated documentation for Trusted Advisor	Trusted Advisor supports additional AWS Foundational Security Best Practices security standard controls that are sourced from AWS Security Hub CSPM. For more information, see the Change log for AWS Trusted Advisor checks.	June 23, 2022
Updated documentation for Trusted Advisor	Added information about how to request service quota increases. For more information, see Service limits.	June 21, 2022
Updated documentation for AWS Support	The create case experience has been updated in the Support Center Console. For more information, see Creating support cases and case management.	May 18, 2022
Updated documentation for Trusted Advisor	Added four checks for Amazon EBS and AWS Lambda. For more information, see Opt in AWS Compute Optimizer to add Trusted Advisor checks.	May 4, 2022
Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	April 27, 2022

Updated documentation for the Exposed Access Keys check	This check is now automatically refreshed for you. For more information, see Change log for AWS Trusted Advisor checks.	April 25, 2022
Updated documentation for Trusted Advisor	The AWS Direct Connect checks in the fault tolerance category are updated. For more information, see Change log for AWS Trusted Advisor checks.	March 29, 2022
Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	March 14, 2022
Added documentation for Trusted Advisor Priority	You can use Trusted Advisor Priority to view a list of prioritized recommendations from your technical account manager (TAM). For more information, see Getting started with Trusted Advisor Priority.	February 28, 2022
Updated documentation for using Amazon EventBridge for Trusted Advisor	You can create an EventBridge rule to monitor changes to your Trusted Advisor checks. For more information, see Monitoring AWS Trusted Advisor check results with EventBridge.	February 21, 2022

New documentation for using Amazon EventBridge to monitor AWS Support cases	You can create an EventBridge rule to monitor and receive notifications about your support cases. For more information, see Monitoring Support cases with EventBridge.	February 21, 2022
Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see AWS managed policy: AWSSupportServiceRolePolicy.	February 17, 2022
Added documentation for integrating with AWS Security Hub CSPM	In the Trusted Advisor console, you can now view the findings for your Security Hub CSPM controls that are part of the AWS Foundational Security Best Practices security standard. For more information, see Viewing AWS Security Hub CSPM controls in the AWS Trusted Advisor console.	January 18, 2022
Updated documentation	If you have an Enterprise On-Ramp Support plan, you have access to all Trusted Advisor checks and the AWS Support API.	November 24, 2021

Updated documentation for Trusted Advisor	The check name for Amazon OpenSearch Service Reserved Instance Optimization was updated. For more informati on, see Change log for AWS Trusted Advisor checks.	September 8, 2021
Updated documentation for Trusted Advisor checks	Added a reference topic for all Trusted Advisor checks. For more information, see AWS Trusted Advisor check reference.	September 1, 2021
Updated documentation for Trusted Advisor managed policies	Updated documentation for the Trusted Advisor managed policies. For more informati on, see AWS managed policies for AWS Support and AWS Trusted Advisor.	August 10, 2021
Updated documentation for Trusted Advisor	Updated documentation for the Trusted Advisor console. For more information, see Get started with AWS Trusted Advisor.	July 16, 2021
Updated documentation for creating Support cases	Added documentation about how to create a related support case for cases that are permanently closed. For more information, see Reopening a closed case and Creating a related case.	June 8, 2021

Updated documentation for Trusted Advisor	Trusted Advisor added two new checks for Amazon Elastic Block Store (Amazon EBS) volume storage. For more information, see Change log for AWS Trusted Advisor checks.	June 8, 2021
Updated documentation	The following topics are updated: • Updated procedures and added content to the Creating Amazon CloudWatch alarms to monitor AWS Trusted Advisor metrics topic • Added the Service quotas for the AWS Support API section	May 12, 2021

Earlier updates

Change	Description	Date
Updated documentation for Trusted Advisor	Added documentation to filter, refresh, and download check results. For more information, see the following sections: • Filter your checks • Refresh check results • Download check results	March 16, 2021
Updated documentation about AWS managed policies	Added information about the `AWSSupportServiceRolePolicy` AWS managed	March 16, 2021

Change	Description	Date
	policy. For more information, see Using service-linked roles for AWS Support.	
Added checks for AWS Lambda	Added four AWS Trusted Advisor checks for Lambda in the Change log for AWS Trusted Advisor.	March 8, 2021
Updated service limit checks for Amazon Elastic Block Store	Updated five AWS Trusted Advisor checks for Amazon EBS in the Change log for AWS Trusted Advisor.	March 5, 2021
Updated documentation for CloudTrail logging	CloudTrail supports logging for console actions when you change your AWS Support plan. For more information, see Logging changes to your Support plan.	February 9, 2021
Updated documentation for Trusted Advisor	Updated the Get started with Trusted Advisor Recommendations topic.	January 29, 2021
Updated documentation for Trusted Advisor reports	Added a Troubleshooting section for using Trusted Advisor reports with other AWS services.	December 4, 2020
Added AWS Trusted Advisor support for AWS CloudTrail logging	CloudTrail supports logging for a subset of Trusted Advisor console actions. For more information, see Logging AWS Trusted Advisor console actions with AWS CloudTrail.	November 23, 2020
Added a change log topic	View changes to AWS Trusted Advisor checks and categories in the Change log for AWS Trusted Advisor.	November 18, 2020
Added support for organizational units	You can now create reports for Trusted Advisor checks for organizational units (OUs). For more information, see Create organizational view reports.	November 17, 2020

Change	Description	Date
Updated the logging with AWS CloudTrail topic	Added an example log entry for a Trusted Advisor API operation. See AWS Trusted Advisor information in CloudTrail logging.	October 22, 2020
Added AWS Support quotas	Added information about the current quotas and restrictions for Support. See the Support endpoints and quotas in the *AWS General Reference*.	August 4, 2020
Organizational view for AWS Trusted Advisor	You can now create reports for Trusted Advisor checks for accounts that are part of AWS Organizations. See Organizational view for AWS Trusted Advisor.	July 17, 2020
Security and AWS Support	Updated information about security considera tions when using AWS Support and Trusted Advisor. See Security in AWS Support	May 5, 2020
Security and AWS Support	Added information about security considera tions when using AWS Support.	January 10, 2020
Using Trusted Advisor as a web service	Added updated instructions to refresh Trusted Advisor data after getting list of Trusted Advisor checks.	November 1, 2018
Using Service-linked roles	Added new section.	July 11, 2018
Getting Started: Troubleshooting	Added troubleshooting links for Route 53 and AWS Certificate Manager.	September 1, 2017
Case Management Example: Creating a Case	Added a note about the **CC** box for users who have the Basic support plan.	August 1, 2017

Change	Description	Date
Monitoring Trusted Advisor Check Results with CloudWatch Events	Added new section.	November 18, 2016
Case Management	Updated the names of case severity levels.	October 27, 2016
Logging AWS Support Calls with AWS CloudTrail	Added new section.	April 21, 2016
Getting Started: Troubleshooting	Added more troubleshooting links.	May 19, 2015
Getting Started: Troubleshooting	Added more troubleshooting links.	November 18, 2014
Getting Started: Case Management	Updated to reflect Service Catalog in the AWS Management Console.	October 30, 2014
Programming the Life of an AWS Support Case	Added information about new API elements for adding attachments to cases and for omitting case communications when retrievin g case history.	July 16, 2014
Accessing AWS Support	Removed named support contacts as an access method.	May 28, 2014
Getting Started	Added the Getting Started section.	December 13, 2013
Initial publication	New AWS Support service released.	April 30, 2013