亚马逊云科技

User Guide

Amazon Web Services Support

API Version 2013-04-15

Amazon Web Services Support: User Guide

Table of Contents

Get started with Amazon Web Services Support 1
Create support cases and case management 1
Creating a support case 2
Describing your problem 5
Choosing a severity 5
Example: Create a support case for account and billing 8
Create a service quota increase 14
Update, resolve, and reopen your cases 15
Update an existing support case 16
Resolve a support case 17
Reopen a resolved case 18
Creating a related case 19
Case history 21
Troubleshooting 21
I want to reopen a live chat for my case 22
I can't connect to a live chat 22
Working with Amazon SDKs 22
About the Amazon Web Services Support API 24
Support case management 24
Amazon Trusted Advisor 25
Endpoints 25
Support in Amazon SDKs 26
Amazon Web Services Support Plans 27
Features of Amazon Web Services Support Plans 27
Changing Amazon Web Services Support Plans 29
Related information 30
Amazon Trusted Advisor 31
Get started with Trusted Advisor Recommendations 32
Sign in to the Trusted Advisor console 32
View check categories 33
View specific checks 34
Filter your checks 35
Refresh check results 36
Download check results 37

Organizational view ... 38
Preferences ... 38
Get started with the Trusted Advisor API ... 39
Using Trusted Advisor as a web service ... 41
Get the list of available Trusted Advisor checks ... 41
Refresh the list of available Trusted Advisor checks ... 42
Poll a Trusted Advisor check for status changes ... 42
Request a Trusted Advisor check result ... 44
Print details of a Trusted Advisor check ... 45
Organizational view for Amazon Trusted Advisor ... 46
Prerequisites ... 46
Enable organizational view ... 47
Refresh Trusted Advisor checks ... 48
Create organizational view reports ... 49
View the report summary ... 50
Download an organizational view report ... 51
Disable organizational view ... 55
Using IAM policies to allow access to organizational view ... 56
Using other Amazon services to view Trusted Advisor reports ... 59
View Trusted Advisor checks powered by Amazon Config ... 68
Troubleshooting ... 68
View your Security Hub controls in Trusted Advisor ... 69
Prerequisites ... 70
View your Security Hub findings ... 71
Refresh your Security Hub findings ... 72
Disable Security Hub from Trusted Advisor ... 72
Troubleshooting ... 73
Opt in Amazon Compute Optimizer for Trusted Advisor checks ... 76
Related information ... 77
Get started with Amazon Trusted Advisor Priority ... 78
Prerequisites ... 79
Enable Trusted Advisor Priority ... 79
View prioritized recommendations ... 80
Acknowledge a recommendation ... 82
Dismiss a recommendation ... 83
Resolve a recommendation ... 85

Reopen a recommendation ... 86
Download recommendation details ... 87
Register delegated administrators ... 88
Deregister delegated administrators ... 88
Manage Trusted Advisor Priority notifications ... 89
Disable Trusted Advisor Priority ... 90
Get started with Amazon Trusted Advisor Engage (Preview) ... 90
Prerequisites ... 91
View the Engagements Dashboard ... 92
View the Catalog of Engagement Types ... 92
Request an Engagement ... 92
Edit an Engagement ... 93
Submit Attachments and Notes ... 94
Change the Engagement Status ... 95
Differentiate Between Recommended and Requested Engagements ... 95
Search Engagements ... 95
Trusted Advisor check reference ... 96
Cost optimization ... 97
Performance ... 103
Security ... 111
Fault tolerance ... 122
Service limits ... 136
Change log for Amazon Trusted Advisor ... 142
New fault tolerance check ... 142
New fault tolerance check ... 142
Updated fault tolerance check ... 143
Updated security check ... 143
New security and performance checks ... 143
New security check ... 143
New fault tolerance and cost optimization checks ... 144
Trusted Advisor check removal ... 144
Updates to the Trusted Advisor integration with Amazon Security Hub ... 144
Update to the Trusted Advisor console ... 145
Added Security Hub checks to Trusted Advisor ... 145
Added checks from Amazon Compute Optimizer ... 145
Updated checks for Amazon Direct Connect ... 146

Updated check name for Amazon OpenSearch Service ... 147
Added checks for Amazon Elastic Block Store volume storage ... 147
Added checks for Amazon Lambda ... 148
Trusted Advisor check removal ... 148
Updated checks for Amazon Elastic Block Store ... 148
Trusted Advisor check removal ... 149
Trusted Advisor check removal ... 150
Amazon Web Services Support App in Slack ... 151
Prerequisites ... 152
Manage access to the Amazon Web Services Support App widget ... 153
Manage access to the Amazon Web Services Support App ... 154
Authorize a Slack workspace ... 160
Authorize multiple accounts ... 162
Configure a Slack channel ... 162
Update your Slack channel configuration ... 165
Create support cases in Slack ... 165
Reply to support cases in Slack ... 168
Join a live chat session with Amazon Web Services Support ... 169
Search for support cases in Slack ... 172
Use your search results ... 173
Resolve support cases in Slack ... 174
Reopen support cases in Slack ... 174
Request service quota increases ... 175
Delete a Slack channel configuration from the Amazon Web Services Support App ... 175
Delete a Slack workspace configuration from the Amazon Web Services Support App ... 176
Amazon Web Services Support App in Slack commands ... 176
Slack channel commands ... 176
Live chat channel commands ... 177
View Amazon Web Services Support App correspondences in the Amazon Support Center Console ... 178
Create Amazon CloudFormation resources for the Amazon Web Services Support App in Slack ... 178
Amazon Web Services Support App and Amazon CloudFormation templates ... 179
Create Slack configuration resources for your organization ... 179
Learn more about CloudFormation ... 184
Create Amazon Web Services Support App resources by using Terraform ... 185

Security ... 186
Data protection ... 187
Security for support cases ... 188
Identity and access management ... 189
Audience ... 189
Authenticating with identities ... 190
Managing access using policies ... 193
How Amazon Web Services Support works with IAM ... 194
Identity-based policy examples ... 197
Using service-linked roles ... 199
Amazon managed policies ... 207
Manage access to Amazon Web Services Support Center ... 257
Manage access to Amazon Web Services Support Plans ... 261
Manage access to Amazon Trusted Advisor ... 265
Example Service Control Policies for Amazon Trusted Advisor ... 278
Troubleshooting ... 279
Incident response ... 282
Logging and monitoring in Amazon Web Services Support and Amazon Trusted Advisor ... 282
Compliance validation ... 283
Resilience ... 284
Infrastructure security ... 284
Configuration and vulnerability analysis ... 285
Code examples ... 286
Actions ... 294
Add a communication to a case ... 294
Add an attachment to a set ... 300
Create a case ... 306
Describe an attachment ... 313
Describe cases ... 318
Describe communications ... 325
Describe services ... 332
Describe severity levels ... 339
Resolve case ... 345
Scenarios ... 350
Get started with cases ... 351
Monitoring and logging for Amazon Web Services Support ... 409

Monitoring Amazon Web Services Support cases with EventBridge 409
Creating an EventBridge rule for Amazon Web Services Support cases 410
Example Amazon Web Services Support events 412
See also 414
Logging Amazon Web Services Support API calls with Amazon CloudTrail 414
Amazon Web Services Support information in CloudTrail 414
Amazon Trusted Advisor information in CloudTrail logging 416
Understanding Amazon Web Services Support log file entries 416
Logging Amazon Web Services Support App API calls with CloudTrail 418
Amazon Web Services Support App information in CloudTrail 418
Understanding Amazon Web Services Support App log file entries 419
Monitoring and logging for Support Plans 424
Logging Amazon Web Services Support Plans API calls with Amazon CloudTrail 424
Amazon Web Services Support Plans information in CloudTrail 425
Understanding Amazon Web Services Support Plans log file entries 426
Logging console actions for changes to your Amazon Web Services Support plan 431
Monitoring and logging for Trusted Advisor 435
Monitoring Trusted Advisor check results with EventBridge 436
Creating CloudWatch alarms to monitor Trusted Advisor metrics 438
Prerequisites 439
CloudWatch metrics for Trusted Advisor 443
Trusted Advisor metrics and dimensions 449
Logging Amazon Trusted Advisor console actions with Amazon CloudTrail 451
Trusted Advisor information in CloudTrail 452
Example: Trusted Advisor Log File Entries 454
Troubleshooting resources 459
Service-specific troubleshooting 459
Document history 464
Earlier updates 487
Amazon Glossary 491

Getting started with Amazon Web Services Support

Amazon Web Services Support offers a range of plans that provide access to tools and expertise that support the success and operational health of your Amazon solutions. All support plans provide 24/7 access to customer service, Amazon documentation, technical papers, and support forums. For technical support and more resources to plan, deploy, and improve your Amazon environment, you can choose a support plan for your Amazon use case.

Notes

- To create a support case in the Amazon Web Services Management Console, see Creating a support case.
- For more information about the different Amazon Web Services Support plans, see Compare Amazon Web Services Support plans and Changing Amazon Web Services Support Plans.
- Support plans offer different response times for your support cases. See Choosing a severity and Response times.

Topics

- Creating support cases and case management
- Creating a service quota increase
- Updating, resolving, and reopening your case
- Troubleshooting
- Using Amazon Web Services Support with an Amazon SDK

Creating support cases and case management

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

- **Account and billing** support cases are available to all Amazon customers. You can get help with billing and account questions.

- **Service limit increase** requests are available to all Amazon customers. For more information about the default service quotas, formerly referred to as limits, see Amazon service quotas in the *Amazon Web Services 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 Changing Amazon Web Services Support Plans.
> - To close your account, see Closing an Account in the *Amazon Billing User Guide*.
> - To find common troubleshooting topics for Amazon Web Services, see Troubleshooting resources.
> - If you're a customer of an Amazon Partner that is part of the Amazon Partner Network, and you use Resold Support, contact your Amazon Partner directly for any billing related issues. Amazon Web Services 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 Amazon Partners can determine Amazon Web Services Support plans in an organization
>   - Amazon Partner-Led Support

Creating a support case

You can create a support case in the Support Center of the Amazon Web Services Management Console.

> **Notes**
>
> - You can sign in to Support Center as the *root user* of your Amazon account or as an Amazon Identity and Access Management (IAM) user. For more information, see Manage access to Amazon Web Services 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 Amazon Support Center Console.

   > **Tip**
   >
   > In the Amazon Web Services 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 limit increases?** and then follow the instructions for Creating 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:
     - Amazon Web Services Management Console
     - Amazon Command Line Interface (Amazon 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 Business, Enterprise On-Ramp, or Enterprise Support 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 a severity

You might be inclined to always create a support case at the highest severity that your support plan allows. However, we recommend that you choose the highest severities for cases that can't be worked around or that directly affect production applications. For information about building your services so that losing single resources doesn't affect your applications, see the Building Fault-Tolerant Applications on Amazon technical paper.

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

Notes

- You can't change the severity code for a support case after you create one. If your situation changes, work with the Amazon Web Services Support agent for your support case.
- For more information about the severity level, see the Amazon Web Services 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,

Severity	Severity level code	First-response time	Description and support plan
			Business, Enterprise On-Ramp, or Enterprise Support plan)
System impaired	`normal`	12 hours	Non-critical functions of your application are behaving abnormally, or you have a time-sensitive development question. (*Developer, Business, Enterprise On-Ramp, or Enterprise Support plan)
Production system impaired	`high`	4 hours	Important functions of your application are impaired or degraded. (Business, Enterprise On-Ramp, or Enterprise Support plan)
Production system down	`urgent`	1 hour	Your business is significantly impacted. Important functions of your application aren't available. (Business, Enterprise On-Ramp, or Enterprise Support 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.

Response times

We make every reasonable effort to respond to your initial request within the indicated timeframe. For information about the scope of support for each Amazon Web Services Support plan, see Amazon Web Services Support features.

If you have a Business, Enterprise On-Ramp, or Enterprise Support plan, you have 24/7 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 Amazon Web Services 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 Business, Enterprise On-Ramp, or Enterprise Support 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 Business, Enterprise On-Ramp, or Enterprise Support plan, technical support is available 24/7 in Chinese.

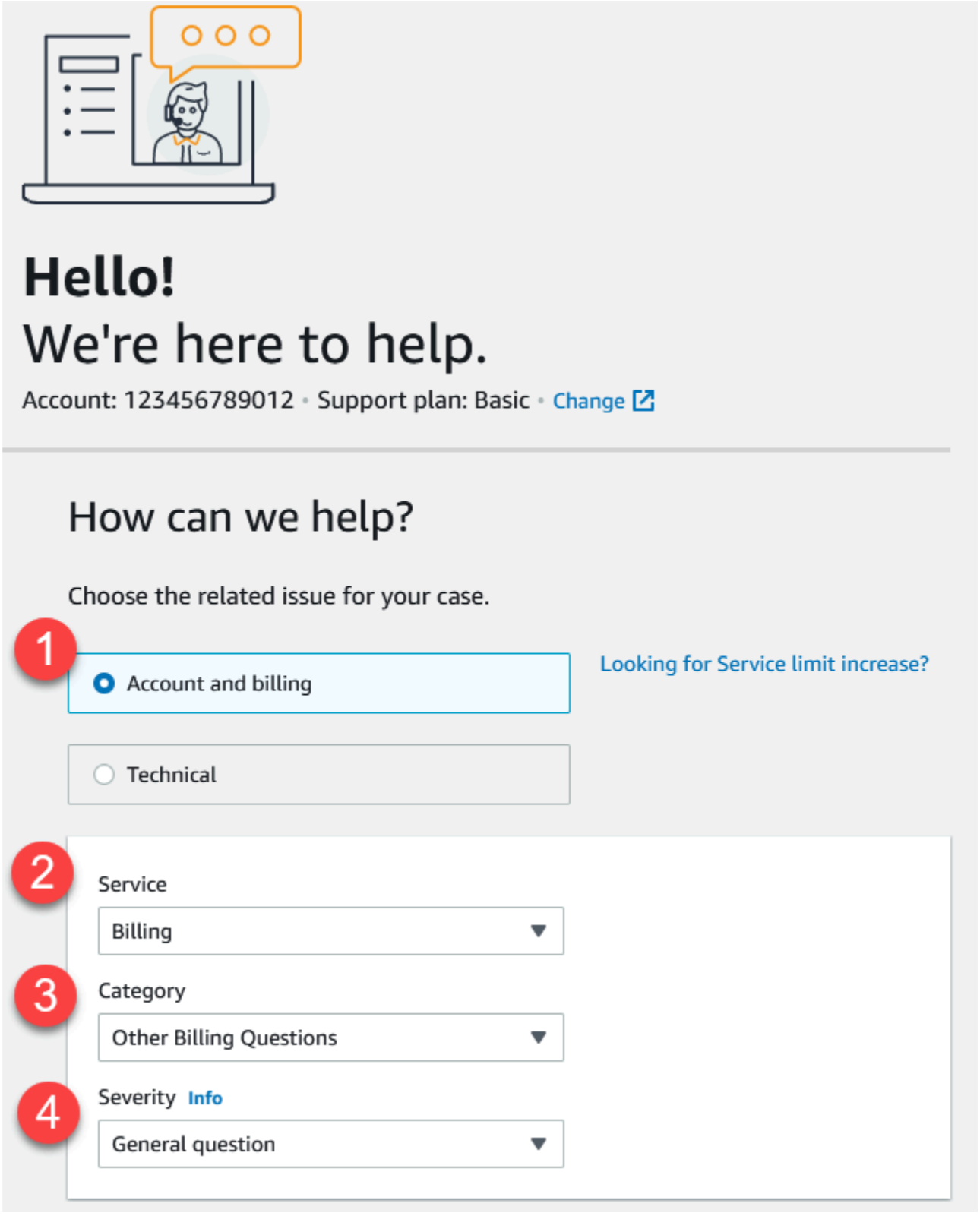
If you choose Korean as your preferred contact language for support cases, support in Korean may be available as follows:

- 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 Business, Enterprise On-Ramp, or Enterprise Support plan, technical support is available 24/7 in Korean.

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**.

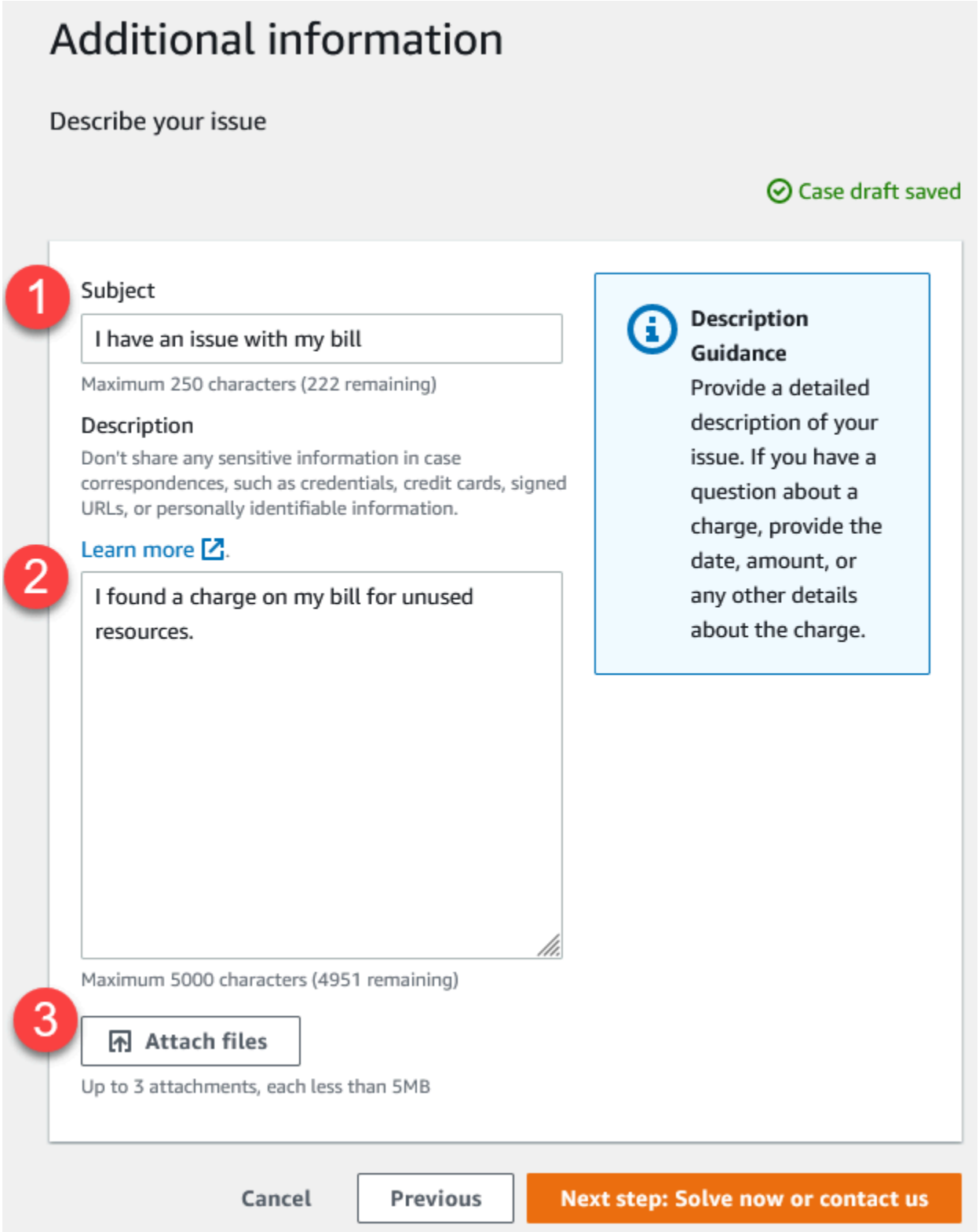
> **Note**
>
> If you have the Basic Support plan, you can't create a technical support case.

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 Amazon Web Services 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 a severity.

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

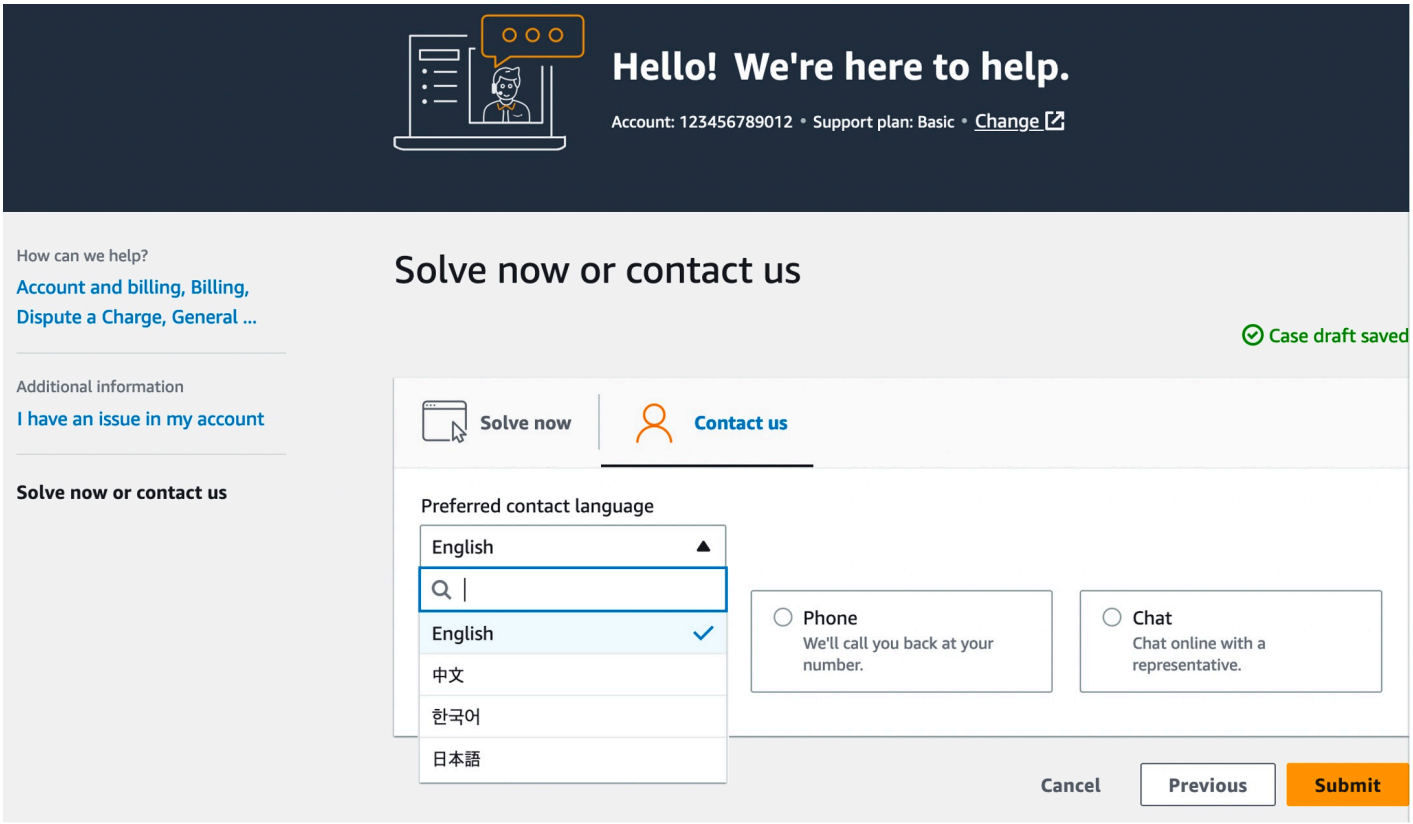
After you specify the case type and classification, you can specify the description and how you want to be contacted.



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 Amazon Web Services 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 Business, Enterprise On-Ramp, or Enterprise Support 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 Business, Enterprise On-Ramp, or Enterprise Support 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 Business, Enterprise On-Ramp, or Enterprise Support plan, technical support is available 24/7 in Korean.

Creating a service quota increase

To improve the performance of your service, request increases to your service quotas (formerly referred to as limits).

> **Note**
>
> You can also use the Service Quotas service to request increases directly for your services. Currently, Service Quotas doesn't support service quotas for all services. For more information, see What is Service Quotas? in the *Service Quotas User Guide*.

To create a support case for service quota increases

1. Sign in to the Amazon Support Center Console.

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

2. Choose **Create case**.
3. Choose **Looking for service limit increases?**
4. To request an increase, follow the prompts. Possible options include the following:

   - **Limit type**
   - **Severity**

   > **Note**
   >
   > Based on your category choice, the prompts might request more information.

5. For **Requests**, choose the **Region**.
6. For **Limit**, choose the service limit type.
7. For **New limit value**, enter the value that you want.
8. (Optional) To request another increase, choose **Add another request**.
9. For **Case description**, describe your support case.
10. For **Contact options** page, choose your preferred language and how you want to be contacted. You can 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**
11. Choose **Submit**. Your case ID number and summary appear.

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 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 Amazon Web Services account that submitted the support case. If you sign in as an Amazon Identity and Access Management (IAM) user, you must have the

required permissions to view support cases. For more information, see Manage access to Amazon Web Services Support Center.

- If you don't respond to the case within a few days, Amazon Web Services 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
- Resolving a support case
- Reopening a resolved case
- Creating a related case
- Case history

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. However, you can't update the severity of a case after you've created it. For more information, see Choosing a severity.

To update an existing support case

1. Sign in to the Amazon Support Center Console.

   Tip

   In the Amazon Web Services 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 Amazon Support Center Console.

   > **Tip**
   >
   > In the Amazon Web Services 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 Amazon Web Services Support resolved your case for you, you can use the feedback link to provide more information about your experience with Amazon Web Services 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 Amazon Support Center Console.

   > **Tip**
   >
   > In the Amazon Web Services 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

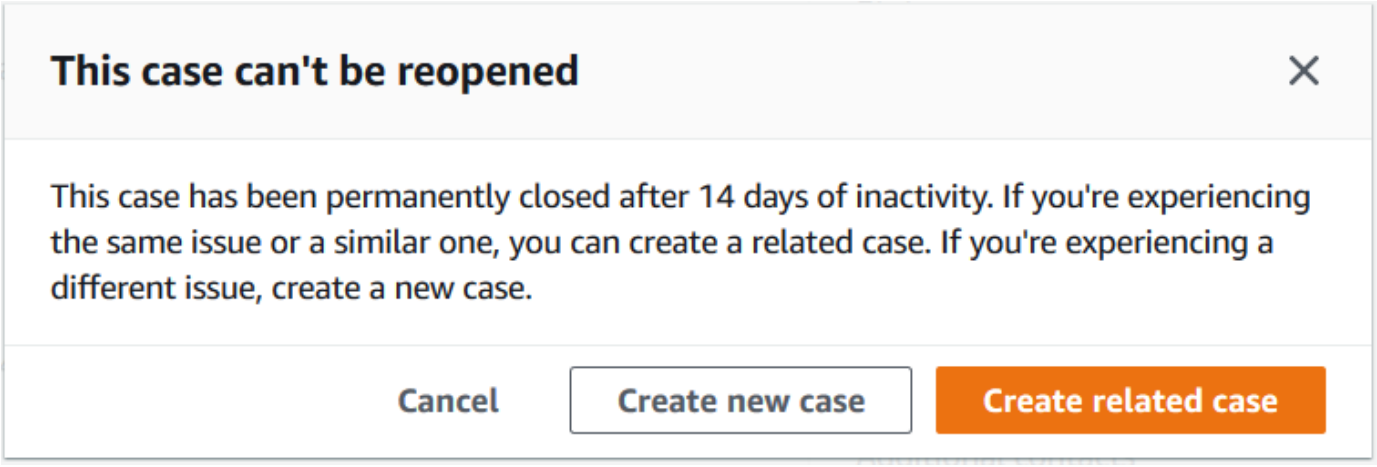
After 14 days of inactivity, you can't reopen a resolved case. If you have a similar issue that is related to the resolved case, you can create a related case. This related case will include a link to the previously resolved case, so that the support agent can review the previous case details and correspondences. 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 Amazon Support Center Console.

   > **Tip**
   >
   > In the Amazon Web Services 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**.

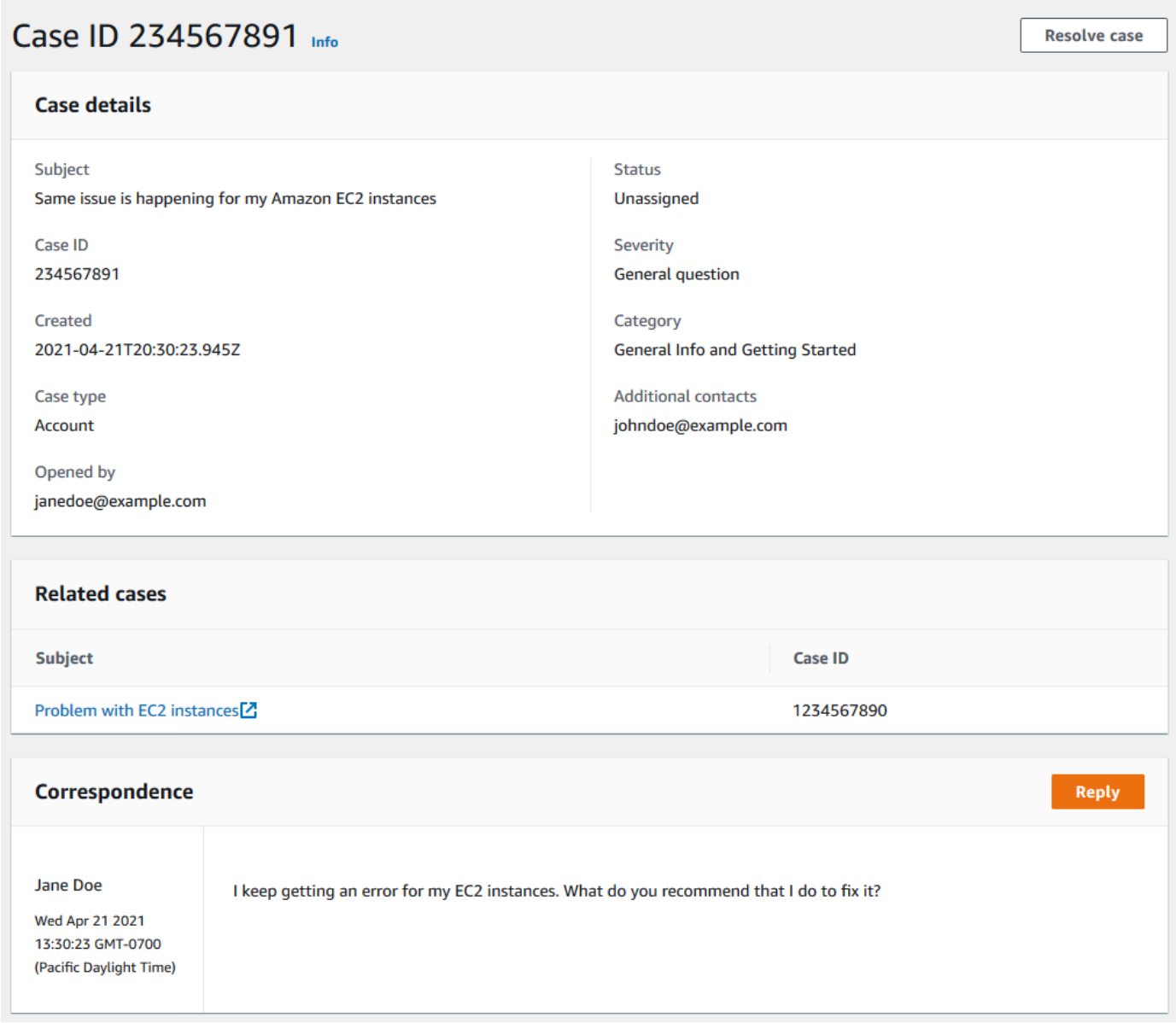


5. Follow the same steps to create your case. See Creating a support case.

> **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.

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 Amazon Web Services Support:
  - Your firewall supports web socket connections.

If you still can’t connect to the chat window, contact Amazon Web Services Support using email or phone contact options.

Using Amazon Web Services Support with an Amazon SDK

Amazon 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.

SDK documentation
Amazon SDK for Java
Amazon SDK for JavaScript
Amazon SDK for .NET
Amazon SDK for PHP

SDK documentation
Amazon SDK for Python (Boto3)
Amazon SDK for Ruby
Amazon SDK for SAP ABAP

About the Amazon Web Services Support API

The Amazon Web Services Support API provides access to some of the features in the Amazon Support Center.

The API provides two different groups of operations:

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

> **Note**
>
> You must have a Business, Enterprise On-Ramp, or Enterprise Support plan to use the Amazon Web Services Support API. For more information, see Amazon Web Services Support.

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

Topics

- Support case management
- Amazon Trusted Advisor
- Endpoints
- Support in Amazon SDKs

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 Amazon Web Services Support API supports CloudTrail logging for support case management operations. For more information, see Logging Amazon Web Services Support API calls with Amazon CloudTrail.

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

Amazon 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 Amazon 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 Amazon Web Services Support API supports CloudTrail logging for Trusted Advisor operations. For more information, see Amazon 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 Amazon 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

Amazon Web Services 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 Amazon Web Services Support API:

https://support.cn-north-1.amazonaws.com.cn

Important

- If you call the CreateCase operation to create test support cases, 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 Amazon Trusted Advisor operations in the Amazon Web Services 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 Amazon endpoints, see Amazon Web Services Support endpoints and quotas in the *Amazon Web Services General Reference*.

Support in Amazon SDKs

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

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

Amazon Web Services Support Plans

You can change your Amazon Web Services Support Plans for your account based on your business needs.

Topics

- Features of Amazon Web Services Support Plans
- Changing Amazon Web Services Support Plans

Features of Amazon Web Services Support Plans

Amazon Web Services Support offers five support plans:

- Basic
- Developer
- Business
- Enterprise On-Ramp
- Enterprise

Basic Support offers support 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 Amazon 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 Amazon 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 On-Ramp, or Enterprise Support plan have access to these features:

- Use-case guidance – What Amazon products, features, and services to use to best support your specific needs.
- Amazon Trusted Advisor – A feature of Amazon Web Services 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 Amazon Web Services Support API to interact with Support Center and Trusted Advisor. You can use the Amazon Web Services 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 Amazon. Third-party software support isn't available for customers on Basic or Developer Support plans.
- Supports an unlimited number of Amazon 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 Amazon Web Services 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.
- White-glove case routing.
- Management business reviews.

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

Changing Amazon Web Services Support Plans

You can use the Amazon Web Services Support Plans console to change your support plan for your Amazon Web Services account. To change your support plan, you must have Amazon Identity and Access Management (IAM) permissions or sign in to your account as a root user. For more information, see Manage access to Amazon Web Services Support Plans and Amazon managed policies for Amazon Web Services Support Plans.

To change your support plan

1. Sign in to the Amazon Web Services Support Plans console at https://console.amazonaws.cn/support/plans/home.
2. (Optional) On the **Amazon Web Services Support Plans** page, compare the support plans. For more information about the pricing, visit the pricing detail page.
3. (Optional) Under **Amazon Web Services Support pricing example**, choose **See examples**, and then choose one of the support plan options to see the estimated cost.
4. When you decide on a plan, choose **Review downgrade** or **Review upgrade** for the plan that you want.

   Notes

   - If you sign up for a paid support plan, you're responsible for a minimum one month subscription of Amazon Web Services Support. For more information, see the Amazon Web Services Support FAQs.
   - If you have an Enterprise On-Ramp or Enterprise Support plan, on the **Change plan confirmation** dialog box, contact Amazon Web Services Support to change your support plan.
5. In the **Change plan confirmation** dialog box, you can expand the support items to see the features that you want to add or remove from your account.

   Under **Pricing**, you can view the projected one-time charges for the new support plan.
6. Choose **Accept and agree**.

Related information

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

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

Amazon Trusted Advisor

Trusted Advisor draws upon best practices learned from serving hundreds of thousands of Amazon customers. Trusted Advisor inspects your Amazon 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 six checks in the Security category.

If you have a Business, Enterprise On-Ramp, or Enterprise Support plan, you can use the Trusted Advisor console and the Amazon 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 Amazon Trusted Advisor check results with Amazon EventBridge.

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

For more information, see Trusted Advisor.

Topics

- Get started with Trusted Advisor Recommendations
- Get started with the Trusted Advisor API
- Using Trusted Advisor as a web service
- Organizational view for Amazon Trusted Advisor
- View Amazon Trusted Advisor checks powered by Amazon Config
- Viewing Amazon Security Hub controls in Amazon Trusted Advisor
- Opt in Amazon Compute Optimizer for Trusted Advisor checks
- Get started with Amazon Trusted Advisor Priority
- Get started with Amazon Trusted Advisor Engage (Preview)
- Amazon Trusted Advisor check reference
- Change log for Amazon 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 Amazon Web Services 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 Amazon Trusted Advisor API to perform operations on your Trusted Advisor checks. For more information, see the Amazon 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 Amazon Identity and Access Management (IAM) permissions to access the Trusted Advisor console. For more information, see Manage access to Amazon Trusted Advisor.

To sign in to the Trusted Advisor console

1. Sign in to the Trusted Advisor console at https://console.amazonaws.cn/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 Amazon solution more secure.
- **Fault tolerance** – Recommendations that help increase the resiliency of your Amazon 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 Amazon services and resources.

- **Operational Excellence** – Recommendations to help you operate your Amazon environment effectively, and at scale.

To view check categories

1. Sign in to the Trusted Advisor console at https://console.amazonaws.cn/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://console.amazonaws.cn/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 Amazon 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 Amazon 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://console.amazonaws.cn/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 Amazon resources, such as Amazon EC2 instances or Amazon 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:

- Amazon Web Services Support enables tagging capabilities for Trusted Advisor
- Tagging Amazon resources in the *Amazon Web Services 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 a Business, Enterprise On-Ramp, or Enterprise Support plan, Trusted Advisor automatically refreshes the checks in your account on a weekly basis.

To refresh Trusted Advisor checks

1. Navigate to the Amazon Trusted Advisor console at https://console.amazonaws.cn/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 **Amazon 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 Amazon Security Hub for your account, you can't use the Trusted Advisor console to refresh Security Hub controls. For more information, see Refresh your Security Hub 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 Amazon Trusted Advisor console at https://console.amazonaws.cn/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 Amazon organization. For more information, see Organizational view for Amazon 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 Amazon Organizations. This is required for the Organizational view for Amazon Trusted Advisor feature, Trusted Advisor Priority, and Trusted Advisor Engage.

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://console.amazonaws.cn/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 Amazon 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 Amazon Organizations, you can create reports for all member accounts in your organization. For more information, see Organizational view for Amazon 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://console.amazonaws.cn/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 Trusted Advisor service role 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?
- Amazon Trusted Advisor check reference

Get started with the Trusted Advisor API

The Amazon 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 Amazon Organization. The Trusted Advisor API uses HTTP methods that returns results in JSON format.

Note

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

You can use the Amazon 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.
- 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 Amazon Trusted Advisor in the Amazon Support User Guide.

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

Using Trusted Advisor as a web service

> **Note**
>
> Trusted Advisor operations will not be supported by the Support API in 2024. Please use the new Amazon Trusted Advisor API to programmatically access best practice checks and recommendations

The Amazon Web Services Support service enables you to write applications that interact with Amazon 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
- Refresh the list of available Trusted Advisor checks
- Poll a Trusted Advisor check for status changes
- Request a Trusted Advisor check result
- Print details of a Trusted Advisor check

Get the list of available Trusted Advisor checks

The following Java code snippet creates an instance of an Amazon Web Services 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 Amazon Web Services 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.

Print 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.

```
// Print 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 Amazon Trusted Advisor

Organizational view lets you view Trusted Advisor checks for all accounts in your Amazon 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 Amazon 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.

> **Note**
>
> The organizational view feature isn't available in the China Regions.

Topics

- Prerequisites
- Enable organizational view
- Refresh Trusted Advisor checks
- Create organizational view reports
- View the report summary
- Download an organizational view report
- Disable organizational view
- Using IAM policies to allow access to organizational view
- Using other Amazon services to view Trusted Advisor reports

Prerequisites

You must meet the following requirements to enable organizational view:

- Your accounts must be members of an Amazon Organization.
- Your organization must have all features enabled for Organizations. For more information, see Enabling all features in your organization in the *Amazon Organizations User Guide*.
- The management account in your organization must have a Business, Enterprise On-Ramp, or Enterprise Support plan. You can find your support plan from the Amazon Web Services Support Center or from the Support plans page. See Compare Amazon Web Services 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 Amazon services in the *Amazon 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 Amazon Trusted Advisor console at https://console.amazonaws.cn/trustedadvisor.
2. In the navigation pane, under **Preferences**, choose **Your organization**.
3. Under **Enable trusted access with Amazon 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 Amazon Web Services 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 a Business, Enterprise On-Ramp, or Enterprise Support 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 Amazon Trusted Advisor console at https://console.amazonaws.cn/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 Amazon Trusted Advisor console at https://console.amazonaws.cn/trustedadvisor.
2. In the navigation pane, choose **Organizational View**.
3. Choose **Create report**.
4. By default, the report includes all Amazon 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 Amazon 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 **Amazon Organization**, select the organizational units (OUs) to include in your report. For more information about OUs, see Managing organizational units in the *Amazon Organizations User Guide*.
6. Choose **Create report**.

Example : Create report filter options

The following example creates a JSON report for the following:

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

In the following example, the report includes the **support-team** OU and one Amazon 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 Amazon Trusted Advisor console at https://console.amazonaws.cn/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 Amazon Trusted Advisor console at https://console.amazonaws.cn/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 a Business, Enterprise On-Ramp, or Enterprise Support 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 Amazon service such as Amazon Athena to run queries on your reports. You can also use Amazon QuickSight to create dashboards and visualize your data. For more information, see Using other Amazon 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 Amazon Trusted Advisor console at https://console.amazonaws.cn/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 Amazon 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 Amazon Command Line Interface (Amazon CLI). For more information, see Deleting a service-linked role in the *IAM User Guide*.

> **Note**
>
> You can use other Amazon services to query and visualize your data for organizational view reports. For more information, see the following resources:
>
> - View Amazon Trusted Advisor recommendations at scale with Amazon Organizations in the *Amazon Management & Governance Blog*
> - Using other Amazon services to view Trusted Advisor reports

Using IAM policies to allow access to organizational view

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

```
{
    "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.

```
{
    "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 Amazon 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 Amazon services to view Trusted Advisor reports

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

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

Prerequisites

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

- Sign in as an Amazon Identity and Access Management (IAM) user with administrator permissions.
- Use the US East (N. Virginia) Amazon Region to quickly set up your Amazon services and resources.
- Create an Amazon QuickSight account. For more information, see Getting Started with Data Analysis in Amazon QuickSight in the *Amazon QuickSight 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 Amazon Web Services Management Console at https://console.amazonaws.cn/.
2. Use the **Region selector** and choose the US East (N. Virginia) Region.
3. Open the Amazon S3 console at https://console.amazonaws.cn/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 Amazon CloudFormation

After you upload your report to Amazon S3, upload the following YAML template to Amazon CloudFormation. This template tells Amazon 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, Amazon Lambda, and Amazon Glue.

To create your resources with Amazon 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 Amazon CloudFormation console at https://console.amazonaws.cn/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 Amazon CloudFormation might create IAM resources**.
15. Choose **Create stack**.

    The stack takes about 5 minutes to create.
16.

Query the data in Amazon Athena

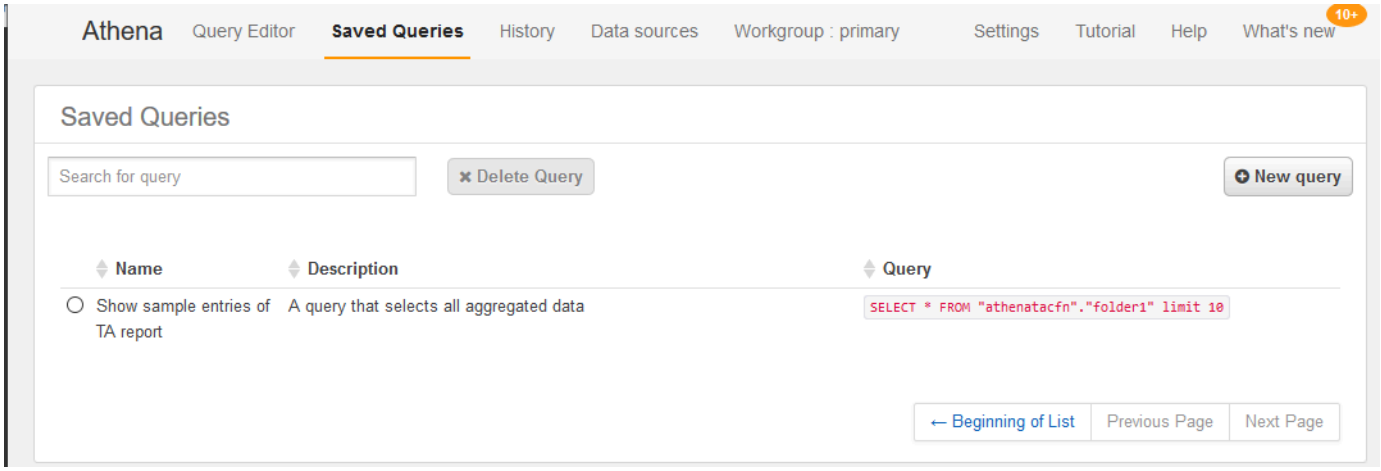
After you have your resources, you can view the data in Athena. Use Athena to create queries and analyze the results of the report, such as looking up specific check results for accounts in the organization.

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://console.amazonaws.cn/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**.



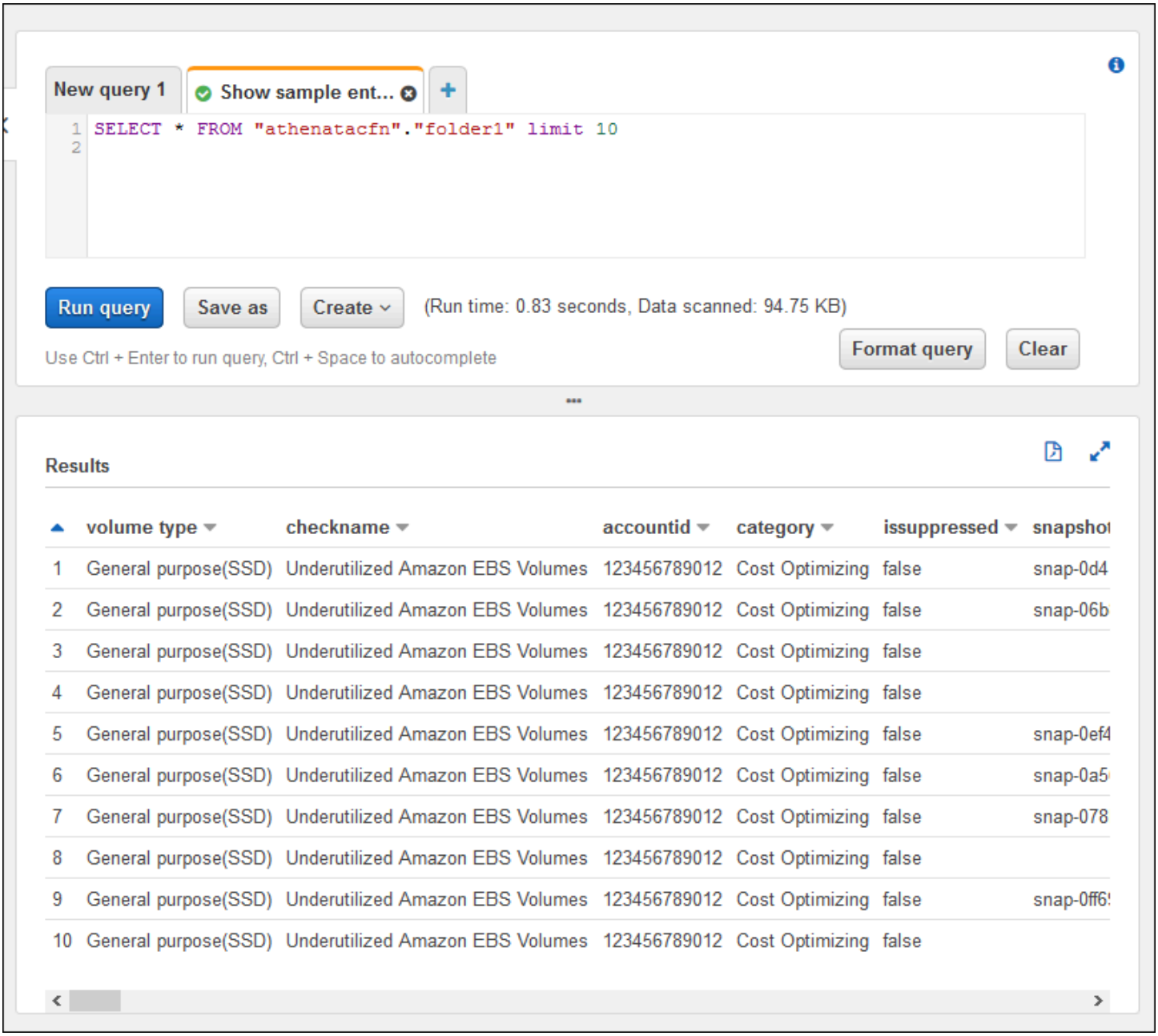
   The query should look like the following.

```
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.



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

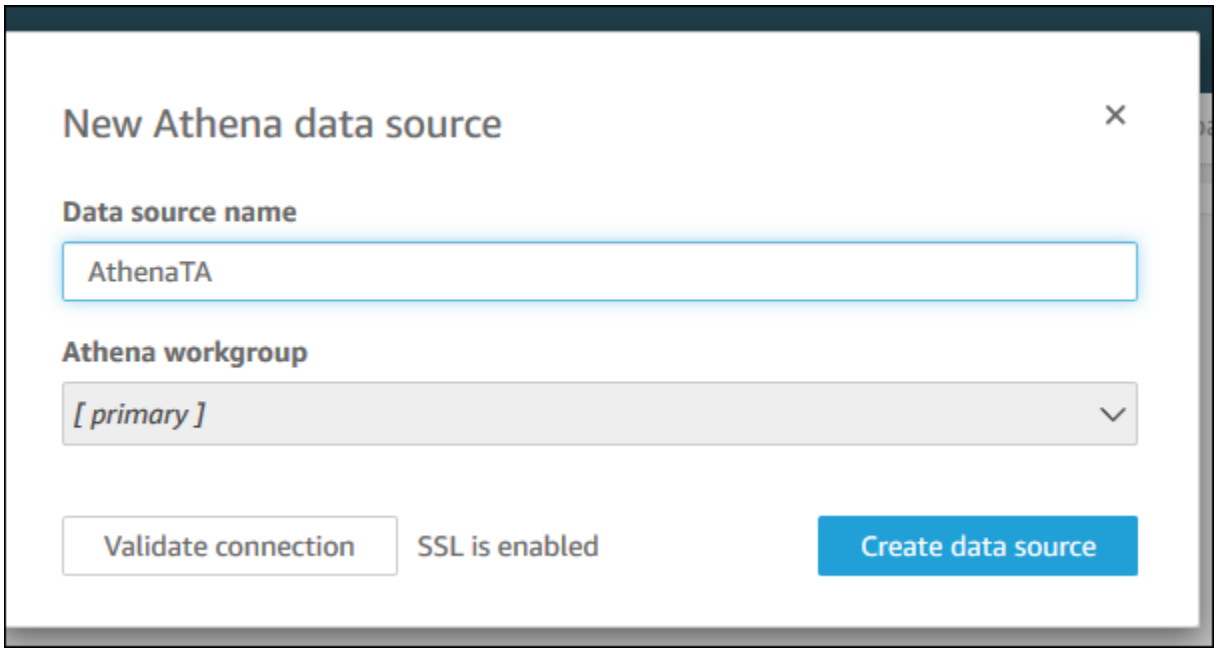
Create a dashboard in Amazon QuickSight

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

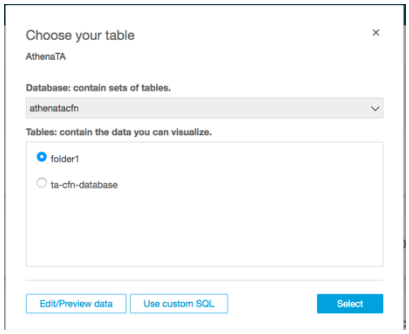
> **Note**
>
> You must use the US East (N. Virginia) Region.

To create a dashboard in Amazon QuickSight

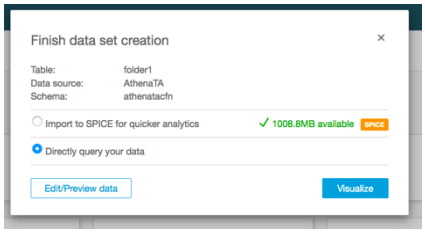
1. Navigate to the Amazon QuickSight console and sign in to your account.
2. Choose **New analysis**, **New dataset**, and then choose **Athena**.
3. In the **New Athena data source** dialog box, enter a data source name such as **AthenaTA**, and then choose **Create data source**.



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**.

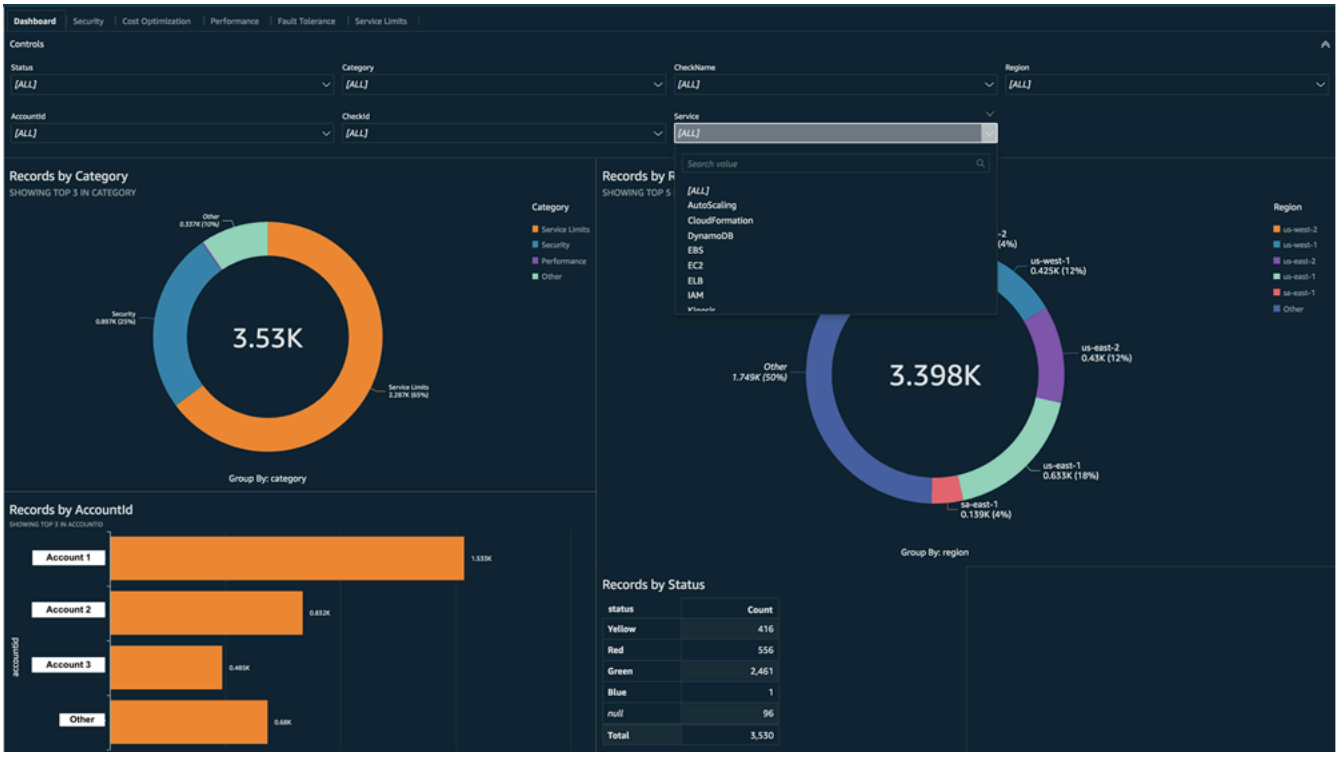


You can now create a dashboard in Amazon QuickSight. For more information, see Working with Dashboards in the *Amazon QuickSight User Guide*.

Example : Amazon QuickSight dashboard

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

- Affected account IDs
- Summary by Amazon 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 Amazon QuickSight can use Athena. For more information, see I Can't Connect to Amazon Athena in the *Amazon QuickSight User Guide*.

For more information and examples for visualizing your report data, see the View Amazon Trusted Advisor recommendations at scale with Amazon Organizations in the *Amazon 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 Amazon Glue console to view the schema and quickly identify if you have duplicate columns in your report.

To find duplicate columns

1. Open the Amazon Glue console at https://console.amazonaws.cn/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 Amazon Web Services Management Console and open the Amazon S3 console at https://console.amazonaws.cn/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 Amazon 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 Amazon Web Services Support Center.

View Amazon Trusted Advisor checks powered by Amazon Config

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

The Amazon Config console guides you through the configuration and activation of managed rules. You can also use the Amazon Command Line Interface (Amazon CLI) or Amazon 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 Amazon Config, see the Amazon Config Developer Guide.

Amazon 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 Amazon Config managed rules, see Amazon Trusted Advisor check reference.

The Amazon Config powered checks are available to customers with Amazon Business Support, Amazon Enterprise On-Ramp, and Amazon Enterprise Support plans. If you enable Amazon Config and you have one of these Amazon Support plans, then you automatically see recommendations powered by corresponding deployed Amazon Config managed rules.

> **Note**
>
> Results for these checks are automatically refreshed based on change-triggered updates to Amazon 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 just enabled recording and managed rules for Amazon Config, but I don't see corresponding Trusted Advisor checks.
- I deployed the same Amazon Config managed rule twice, what will I see in Trusted Advisor?
- I turned off recording for Amazon Config in an Amazon Region. What will I see in Trusted Advisor?

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

After the Amazon 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 Amazon Web Services Support Center.

I deployed the same Amazon 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 Amazon Config in an Amazon Region. What will I see in Trusted Advisor?

If you turned off resource recording for Amazon Config in an Amazon Region, then Trusted Advisor no longer receives data for corresponding managed rules and checks in that Region. Existing managed rule results remain in Amazon Config and in Trusted Advisor until Amazon 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 Amazon Security Hub controls in Amazon Trusted Advisor

After you enable Amazon Security Hub for your Amazon Web Services account, you can view your security controls and their findings in the Trusted Advisor console. You can use Security Hub 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 recommendations to address your security issues. You can use this feature to find security recommendations from Trusted Advisor and Security Hub in one convenient location.

Notes

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

  For more information about the Security Hub categories, see Control categories.
- Currently, when Security Hub adds new controls to the Amazon Foundational Security Best Practices security standard, there can be a delay of two to four weeks before you can view them in Trusted Advisor. This time frame is best effort and isn't guaranteed.

Topics

- Prerequisites
- View your Security Hub findings
- Refresh your Security Hub findings
- Disable Security Hub from Trusted Advisor
- Troubleshooting

Prerequisites

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

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

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

About Amazon 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 Amazon Web Services Support to enable this feature. However, member accounts in your organization must enable Security Hub 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 Amazon Organizations accounts.

View your Security Hub findings

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

To view your Security Hub 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 **Amazon Security Hub** to filter for Security Hub controls.

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

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

- **Security Hub resources** – You can find the resources in your account where Security Hub has detected an issue.

Notes

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

Refresh your Security Hub findings

After you enable a security standard, it can take up to two hours for Security Hub 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 **Amazon Foundational Security Best Practices v1.0.0** security standard, check the Trusted Advisor console again later.

Note

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

Disable Security Hub from Trusted Advisor

Follow this procedure if you don't want your Security Hub information to appear in the Trusted Advisor console. This procedure only disables the Security Hub integration with Trusted Advisor.

It won't affect your configurations with Security Hub. You can continue to use the Security Hub console to view your security controls, resources, and recommendations.

To disable the Security Hub integration

1. Contact Amazon Web Services Support and request to disable the Security Hub integration with Trusted Advisor.

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

Disable this feature for Amazon Organizations accounts

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

If you're a member account in an organization, you can contact Amazon Web Services 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 don't see Security Hub findings in the Trusted Advisor console
- I configured Security Hub and Amazon Config correctly, but my findings are still missing
- I want to disable specific Security Hub controls
- I want to find my excluded Security Hub resources
- I want to enable or disable this feature for a member account that belongs to an Amazon organization
- I see multiple Amazon Web Services Regions for the same affected resource for a Security Hub check
- I turned off Security Hub or Amazon Config in a Region
- My control is archived in Security Hub, but I still see the findings in Trusted Advisor

- I still can't view my Security Hub findings

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

Verify that you completed the following steps:

- You have a Business, Enterprise On-Ramp, or Enterprise Support plan.
- You enabled resource recording in Amazon Config within the same Region as Security Hub.
- You enabled Security Hub and selected the **Amazon Foundational Security Best Practices v1.0.0** security standard.
- New controls from Security Hub 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 and Amazon Config correctly, but my findings are still missing

It can take up to two hours for Security Hub 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 Amazon 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 or Security Hub 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 controls

Security Hub sends your data to Trusted Advisor automatically. If you disable a Security Hub 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 console and verify if your control is enabled or disabled.

If you disable a Security Hub control or disable all controls for the Amazon 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 resources

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

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 resources from the Trusted Advisor console. To do so, use the Security Hub 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 Amazon organization

By default, member accounts inherit the feature from the management account for Amazon 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 Amazon Web Services Support.

I see multiple Amazon Web Services Regions for the same affected resource for a Security Hub check

Some Amazon Web 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 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 and Amazon Config so that multiple Regions won't appear for the same resource. For more information, see Amazon Foundational Best Practices controls that you might want to disable.

I turned off Security Hub or Amazon Config in a Region

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

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

My control is archived in Security Hub, 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 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 findings

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

Opt in Amazon Compute Optimizer for Trusted Advisor checks

Compute Optimizer is a service that analyzes the configuration and utilization metrics of your Amazon 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 Amazon Web Services account only, or all member accounts that are part of an organization in Amazon Organizations. For more information, see Getting started in the *Amazon 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
- Amazon Lambda over-provisioned functions for memory size

Performance

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

Notes

- Results for these checks are automatically refreshed several times daily. Refresh requests are not allowed. It might take a few hours for changes to appear. Currently, you can't exclude resources from these checks.
- Trusted Advisor already has the Underutilized Amazon EBS Volumes and the Overutilized Amazon EBS Magnetic Volumes checks.

  Once 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 *Amazon Compute Optimizer User Guide*
- Viewing Lambda function recommendations in the *Amazon Compute Optimizer User Guide*
- Configuring Lambda function memory in the *Amazon Lambda Developer Guide*

- Request modifications to your Amazon EBS volumes in the *Amazon EC2 User Guide for Linux Instances*

Get started with Amazon Trusted Advisor Priority

Trusted Advisor Priority helps you secure and optimize your Amazon Web Services account to follow Amazon Web Services best practices. With Trusted Advisor Priority, your Amazon Web Services 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 Amazon 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:

- Amazon Web Services – Services such as Trusted Advisor, Amazon Security Hub, and Amazon 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
- Enable Trusted Advisor Priority
- View prioritized recommendations
- Acknowledge a recommendation
- Dismiss a recommendation
- Resolve a recommendation

- Reopen a recommendation
- Download recommendation details
- Register delegated administrators
- Deregister delegated administrators
- Manage Trusted Advisor Priority notifications
- Disable Trusted Advisor Priority

Prerequisites

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

- You must have an Enterprise Support plan.
- Your account must be part of an organization that has enabled all features in Amazon Organizations. For more information, see Enabling all features in your organization in the *Amazon 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 Amazon 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 Amazon 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 Amazon Trusted Advisor and Amazon Web Services managed policies for Amazon Trusted Advisor.

Enable Trusted Advisor Priority

Ask your account team to enable this feature for you. You must have an Enterprise Support 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 Amazon Organizations, then choose **Enable trusted access with Amazon 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 Amazon account.

To view your prioritized recommendations

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

   If you're using an Amazon 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 Amazon Web Services, your Amazon Web Services 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 Amazon 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://console.amazonaws.cn/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 Amazon Web Services, your Amazon Web Services 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://console.amazonaws.cn/trustedadvisor/home.
2. If you're using an Amazon 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 Amazon 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://console.amazonaws.cn/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 Amazon Web Services 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://console.amazonaws.cn/trustedadvisor/home.

2. If you're using an Amazon 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 Amazon 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://console.amazonaws.cn/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.

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://console.amazonaws.cn/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 Amazon 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://console.amazonaws.cn/trustedadvisor/home.
2. If you're using an Amazon 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://console.amazonaws.cn/trustedadvisor/home.
2. If you're using an Amazon 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 Amazon 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://console.amazonaws.cn/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.

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://console.amazonaws.cn/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 Amazon Identity and Access Management permissions to access Trusted Advisor Priority. For more information, see Manage access to Amazon Trusted Advisor and Amazon Web Services managed policies for Amazon 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://console.amazonaws.cn/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://console.amazonaws.cn/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://console.amazonaws.cn/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 Amazon 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.

Get started with Amazon Trusted Advisor Engage (Preview)

> **Note**
>
> Amazon Trusted Advisor Engage is in preview release and is subject to change. You can see preview service terms here https://aws.amazon.com/service-terms/.

You can use Amazon Trusted Advisor Engage to get the most out of your Amazon Web Services Support Plans by making it easy for you to see, request and track all your proactive engagements, and communicate with your Amazon Web Services account team about ongoing engagements.

For example, you can request a “Management Business Review” towards your Amazon Web Services account team by going into the **Engage** page within the Amazon Trusted Advisor console. Then, an Amazon Web Services expert will be assigned to your request, and follow through the entire engagement.

Topics

- Prerequisites

- View the Engagements Dashboard
- View the Catalog of Engagement Types
- Request an Engagement
- Edit an Engagement
- Submit Attachments and Notes
- Change the Engagement Status
- Differentiate Between Recommended and Requested Engagements
- Search Engagements

Prerequisites

You must take necessary action to satisfy the following requirements in order to use Trusted Advisor Engage:

- You must have an Enterprise On-Ramp Support plan.
- Your account must be part of an organization which has enabled all features in Amazon Organizations. For more information, see Enabling all features in your organization in the *Amazon Organizations User Guide*.
- Your organization must have enabled trusted access to Trusted Advisor. You can enable trusted access by logging in as the management account and going to the Your organization page in the Trusted Advisor console.
- You must have Amazon Identity and Access Management (IAM) permissions to access Trusted Advisor Engage. For information about how to control access to Trusted Advisor Engage, see Manage access to Amazon Trusted Advisor.

> **Note**
>
> Any account within an Amazon Organization can create an engagement request. If an Engagement-owning account moves to a different Amazon Web Services Organization, the Engagement will only be accessible by the account. To limit controls, see Example Service Control Policies for Amazon Trusted Advisor.

View the Engagements Dashboard

After you have obtained access rights, you can access the Trusted Advisor Engage page within the Trusted Advisor console to view a dashboard where you can manage engagements with your Amazon Web Services account team.

To manage your Engagements:

1. Sign in to the Trusted Advisor console at https://console.amazonaws.cn/trustedadvisor/home.
2. On the **Trusted Advisor Engage** page, you can view the:
   - **Request Engagement** Button
   - **Active Engagements** Table
   - **Closed Engagements** Table
   - **All Available Engagements** Catalog

View the Catalog of Engagement Types

You can view the catalog of engagement types to find the latest types of engagements that you can request towards your Amazon Web Services account team.

To view the catalog of engagement types:

1. Sign in to the Trusted Advisor console at https://console.amazonaws.cn/trustedadvisor/home.
2. On the **Trusted Advisor Engage** page, you can find the catalog of Engagement types.

Request an Engagement

You can request engagements to your Amazon Web Services account team according to the engagement types included in your Amazon Web Services Support Plan.

To request an Engagement:

1. Sign in to the Trusted Advisor console at https://console.amazonaws.cn/trustedadvisor/home.
2. On the **Trusted Advisor Engage** page, choose **Request Engagement**.
3. Fill out the:

- **Title**
- **Select Engagement**: the type of Engagement you want to request.
- **Desired Completion Date**: the desired completion date of the Engagement. Each Engagement Type has a different lead time which is calculated in the minimum desired completion date.
- **Request Visibility**:
  - **My account**: this engagement request is visible only to your account.
  - **My account and Admin accounts**: this engagement request is visible to your account, and the Management account and all Delegated Admin accounts of your Amazon Web Services Organization.
  - **Organization**: This engagement request is visible to all accounts in your Amazon Web Services Organization.
- **Engagement Requester Email**: the email address that Amazon Web Services will use as the primary point of contact for this Engagement.
- **Email notification settings**: choose if the Engagement Requester Email will receive email notifications about the engagement.
- **Point of escalation**: the email address that Amazon Web Services will use when an escalation is required for this Engagement.
- **Correspondence**: a note and an optional file attachment for you to provide details regarding this Engagement.

4. Choose **Send Request**.

Edit an Engagement

You can edit details on your engagement request.

To edit an Engagement:

1. Sign in to the Trusted Advisor console at https://console.amazonaws.cn/trustedadvisor/home.
2. On the **Trusted Advisor Engage** page, select an existing engagement.
3. Select **Edit**.
4. You can edit the:

   - **Title**

- **Desired Completion Date**: the desired completion date of the Engagement. Each Engagement Type has a different lead time which is calculated in the minimum desired completion date.
- **Request Visibility**:
  - **My account**: this engagement request is visible only to your account.
  - **My account and Admin accounts**: this engagement request is visible to your account, and the Management account and all Delegated Admin accounts of your Amazon Web Services Organization.
  - **Organization**: This engagement request is visible to all accounts in your Amazon Web Services Organization.
- **Engagement Requester Email**: the email address that Amazon Web Services will use as the primary point of contact for this Engagement.
- **Email notification settings**: choose if the Engagement Requester Email will receive email notifications about the engagement.
- **Point of escalation**: the email address that Amazon Web Services will use when an escalation is required for this Engagement.

5. Choose **Save**.

Submit Attachments and Notes

You can communicate with your Amazon Web Services account team on individual engagements by sending notes and file attachments to support your engagement request. You can include a single attachment and note per communication, you can only attach files to an engagement with the same Amazon Web Services account which requested the engagement, and you can not delete attachments or notes after a communication has been sent.

To attach files or add notes to an Active Engagement request:

1. Sign in to the Trusted Advisor console at https://console.amazonaws.cn/trustedadvisor/home.
2. On the **Trusted Advisor Engage** page, choose the ID of the **active engagement** to which you would like to attach files or add notes.
3. Choose **Correspondence** to expand the form.
4. Enter a note for your assigned TAM and optionally attach a file. Don't share any sensitive information in correspondences, such as passwords, credit card data, signed URLs, or personally identifiable information.

5. Choose **Save**.

Change the Engagement Status

You can change that status of engagements to cancel engagements which are pending response, complete engagements which are in progress, and reopen engagements which have been marked as cancelled or closed.

To change the status of an Engagement:

1. Sign in to the Trusted Advisor console at https://console.amazonaws.cn/trustedadvisor/home.
2. On the **Trusted Advisor Engage** page, choose the ID of the **active engagement** of which you would like to change the status.
3. On the **Engagement** details page, you can change the status to **Cancelled** or **Complete**.

   - You are able to select **Cancel** when engagement status is **Pending Response**.
   - You are able to select **Complete** when engagement status is **In Progress**.
   - You are able to select **Reopen** for closed engagements. Cancelled engagements move to **Pending Response**, while Complete engagements move to **In Progress**.

Differentiate Between Recommended and Requested Engagements

You can identify the source of engagements to know whether an engagement was requested by you or recommended by your Amazon Web Services account team.

To view different sources of Active Engagements:

1. Sign in to the Trusted Advisor console at https://console.amazonaws.cn/trustedadvisor/home.
2. On the **Trusted Advisor Engage** page, view the **Effective Date** column to distinguish between **Recommended** and **Requested** Engagements:

   - **Recommended**: Engagement request created by your Amazon Web Services account teams.
   - **Requested**: Engagement request created by the user.

Search Engagements

You can search your existing active and closed engagements using filters.

To search Engagements:

1. Sign in to the Trusted Advisor console at https://console.amazonaws.cn/trustedadvisor/home.
2. On the **Trusted Advisor Engage** page, you can select from the following filters:
   - **Age (days)**
   - **Engagement Type**
   - **Request Title**
   - **Status**
   - **Desired Completion Date**
   - **Effective Date**

Amazon 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 Business, Enterprise On-Ramp, or Enterprise Support plan, you can also use the Amazon Trusted Advisor API and the Amazon Command Line Interface (Amazon CLI) to access your checks. For more information, see the following topics:

- Get started with the Trusted Advisor API

> **Note**
>
> If you have a Basic Support and Developer Support plan, you can use the Trusted Advisor console to access all checks in the Service limits category and the following checks in the security category:
>
> - Amazon S3 Bucket Permissions
> - Security Groups – Specific Ports Unrestricted

Note

You can use the following checks in the China Regions.

Cost optimization

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

Check names

- Amazon EC2 instances over-provisioned for Microsoft SQL Server
- Amazon EC2 Instances Stopped
- Amazon S3 Incomplete Multipart Upload Abort Configuration
- Idle Load Balancers
- Unassociated Elastic IP Addresses

Amazon EC2 instances over-provisioned for Microsoft SQL Server

Description

Checks your Amazon Elastic Compute Cloud (Amazon EC2) instances that are running SQL Server in the past 24 hours. An SQL Server database has a compute capacity limit for each instance. An instance with SQL Server Standard edition can use up to 48 vCPUs. An instance with SQL Server Web can use up to 32 vCPUs. This check alerts you if an instance exceeds this vCPU limit.

If your instance is over-provisioned, you pay full price without realizing an improvement in performance. You can manage the number and size of your instances to help lower costs.

Estimated monthly savings are calculated by using the same instance family with the maximum number of vCPUs that an SQL Server instance can use and the On-Demand pricing. Actual savings will vary if you're using Reserved Instances (RI) or if the instance isn't running for a full day.

> **Note**
>
> Results for this check are automatically refreshed several times daily, and refresh requests are not allowed. It might take a few hours for changes to appear. Currently, you can't exclude resources from this check.

Check ID

`Qsdfp3A4L1`

Alert Criteria

- Red: An instance with SQL Server Standard edition has more than 48 vCPUs.
- Red: An instance with SQL Server Web edition has more than 32 vCPUs.

Recommended Action

For SQL Server Standard edition, consider changing to an instance in the same instance family with 48 vCPUs. For SQL Server Web edition, consider changing to an instance in the same instance family with 32 vCPUs. If it is memory intensive, consider changing to memory optimized R5 instances. For more information, see Best Practices for Deploying Microsoft SQL Server on Amazon EC2.

Additional Resources

- Microsoft SQL Server on Amazon
- You can use Launch Wizard to simplify your SQL Server deployment on EC2.

Report columns

- Status
- Region
- Instance ID
- Instance Type
- vCPU
- SQL Server Edition
- Maximum vCPU
- Recommended Instance Type
- Estimated Monthly Savings

- Last Updated Time

Amazon EC2 Instances Stopped

Description

Checks if there are Amazon EC2 instances that have been stopped for more than 30 days.

You can specify the allowed number of days value in the **AllowedDays** of Amazon Config parameters.

For more information, see Why am I being charged for Amazon EC2 when all my instances were terminated?

> **Note**
>
> Results for this check are automatically refreshed several times daily, and refresh requests are not allowed. It might take a few hours for changes to appear. Currently, you can't exclude resources from this check.

Check ID

`c18d2gz150`

Source

`AWS Config Managed Rule: ec2-stopped-instance`

Alert Criteria

- Yellow: There are Amazon EC2 instances stopped for more than the allowed number of days.

Recommended Action

Review the Amazon EC2 instances that have been stopped for 30 days or more. To avoid incuring unnecessary costs, terminate any instances that are no longer needed.

For more information, see Terminate your instance.

Additional Resources

- Amazon EC2 On-Demand Pricing

Report columns

- Status
- Region
- Resource
- Amazon Config Rule
- Input Parameters
- Last Updated Time

Amazon S3 Incomplete Multipart Upload Abort Configuration

Description

Checks that each Amazon S3 bucket is configured with a lifecycle rule to abort multipart uploads that remain incomplete after 7 days. Using a lifecycle rule to abort these incomplete uploads and delete the associated storage is recommended.

> **Note**
>
> Results for this check are automatically refreshed one or more times each day, and refresh requests are not allowed. It might take a few hours for changes to appear. Currently, you can't exclude resources from this check.

Check ID

`c1cj39rr6v`

Alert Criteria

Yellow: The lifecycle configuration bucket does not contain a lifecycle rule to abort all multipart uploads that remain incomplete after 7 days.

Recommended Action

Review lifecycle configuration for buckets without a lifecycle rule that would cleanup all incomplete multipart uploads. Uploads that are not completed after 24 hours are unlikely to be completed. Click here to follow instructions to create a lifecycle rule. It is recommended that this is applied to all objects in your bucket. If you have a need to apply other lifecycle actions

to selected objects in your bucket, you can have multiple rules with different filters. Check the storage lens dashboard or call the ListMultipartUpload API for more information.

Additional Resources

Creating a lifecycle configuration

Discovering and Deleting Incomplete Multipart Uploads to Lower Amazon S3 Costs

Uploading and copying objects using multipart upload

Lifecycle configuration elements

Elements to describe lifecycle actions

Lifecycle configuration to abort multipart uploads

Report columns

- Status
- Region
- Bucket Name
- Bucket ARN
- Lifecycle rule for deleting incomplete MPU
- Days After Initiation
- Last Updated Time

Idle Load Balancers

Description

Checks your Elastic Load Balancing configuration for load balancers that are idle.

Any load balancer that is configured accrues charges. If a load balancer has no associated back-end instances, or if network traffic is severely limited, the load balancer is not being used effectively. This check currently only checks for Classic Load Balancer type within ELB service. It does not include other ELB types (Application Load Balancer, Network Load Balancer).

Check ID

`hjLMh88uM8`

Alert Criteria

- Yellow: A load balancer has no active back-end instances.
- Yellow: A load balancer has no healthy back-end instances.
- Yellow: A load balancer has had less than 100 requests per day for the last 7 days.

Recommended Action

If your load balancer has no active back-end instances, consider registering instances or deleting your load balancer. See Registering Your Amazon EC2 Instances with Your Load Balancer or Delete Your Load Balancer.

If your load balancer has no healthy back-end instances, see Troubleshooting Elastic Load Balancing: Health Check Configuration.

If your load balancer has had a low request count, consider deleting your load balancer. See Delete Your Load Balancer.

Additional Resources

- Managing Load Balancers
- Troubleshoot Elastic Load Balancing

Report columns

- Region
- Load Balancer Name
- Reason
- Estimated Monthly Savings

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.

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 Aurora DB cluster under-provisioned for read workload
- Amazon EBS Provisioned IOPS (SSD) Volume Attachment Configuration
- Amazon RDS instance under-provisioned for system capacity
- High Utilization Amazon EC2 Instances
- Large Number of EC2 Security Group Rules Applied to an Instance
- Large Number of Rules in an EC2 Security Group

- Overutilized Amazon EBS Magnetic Volumes

Amazon Aurora DB cluster under-provisioned for read workload

Description

Checks whether Amazon Aurora DB cluster has the resources to support a read workload.

Check ID

`c1qf5bt038`

Alert Criteria

Yellow:

Increased database reads: The database load was high and the database was reading more rows than writing or updating the rows.

Recommended Action

We recommend that you tune your queries to decrease the database load or add a reader DB instance to your DB cluster with the same instance class and size as the writer DB instance in the cluster. The current configuration has at least one DB instance with a continuously high database load caused mostly by read operations. Distribute these operations by adding another DB instance to the cluster and directing the read workload to the DB cluster read-only endpoint.

Additional Resources

An Aurora DB cluster has one reader endpoint for read-only connections. This endpoint uses load balancing to manage the queries contributing the most to database load in your DB cluster. The reader endpoint directs these statements to the Aurora Read Replicas and reduces the load on the primary instance. The reader endpoint also scales the capacity to handle concurrent SELECT queries with the number of Aurora Read Replicas in the cluster.

For more information, see Adding Aurora Replicas to a DB Cluster and Managing performance and scaling for Aurora DB clusters.

Report columns

- Status
- Region

- Resource
- Increased database read (count)
- Last detection period
- Last Updated Time

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

Amazon RDS instance under-provisioned for system capacity

Description

Checks whether Amazon RDS instance or Amazon Aurora DB instance has the required system capacity to operate.

Check ID

`c1qf5bt039`

Alert Criteria

Yellow:

Out-of-memory kills: When a process on the database host is stopped because of memory reduction at the OS level, the Out Of Memory (OOM) kills counter increases.

Excessive swapping: os.memory.swap.in and os.memory.swap.out metric values were high.

Recommended Action

We recommend that you tune your queries to use less memory or use a DB instance type with higher allocated memory. When the instance is running low on memory, this impacts the database performance.

Additional Resources

Out-of-memory kills were detected: Linux kernel invokes the Out of Memory (OOM) Killer when the processes running on the host require more than the memory physically available from the operating system. In this case, the OOM Killer reviews all the running processes, and stops one or more processes, in order to free up system memory and keep the system running.

Swapping is detected: When the memory isn't sufficient on the database host, the operating system sends a few minimum used pages to the disk in the swap space. This offloading process impacts the database performance.

For more information, see Amazon RDS Instance Types and Scaling yourAmazon RDS instance.

Report columns

- Status
- Region
- Resource
- Out-of-memory kills (count)
- Excessive swapping (count)
- Last detection period
- Last Updated Time

High 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.

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 Amazon Elastic Compute Cloud (Amazon EC2) instances that have a large number of security group rules. Performance can be degraded if an instance has a large number of rules.

Check ID

`j3DFqYTe29`

Alert Criteria

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

Amazon 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.

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 Deleting Rules from a Security Group.

Additional Resources

Amazon EC2 Security Groups

Report columns

- Region
- Security Group Name
- Group ID
- Description
- Instance Count
- VPC ID
- Total Inbound Rules

- Total Outbound Rules

Overutilized Amazon EBS Magnetic Volumes

Description

Checks for Amazon Elastic Block Store (Amazon EBS) magnetic volumes that are potentially overutilized 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 (Amazon EBS)

Report columns

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

> **Note**
>
> If you opted in your account for Amazon Compute Optimizer, we recommend that you use the Amazon EBS under-provisioned volumes check instead. For more information, see Opt in Amazon Compute Optimizer for Trusted Advisor checks.

Security

You can use the following checks for the security category.

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

Check names

- Amazon EC2 instances with Ubuntu LTS end of standard support
- Amazon EFS clients not using data-in-transit encryption
- Amazon Route 53 mismatching CNAME records pointing directly to S3 buckets
- Amazon S3 Bucket Permissions

- ELB Listener Security
- ELB Security Groups
- IAM Password Policy
- Security Groups – Specific Ports Unrestricted
- Security Groups – Unrestricted Access

Amazon EC2 instances with Ubuntu LTS end of standard support

Description

This check alerts you if the versions are near or have reached the end of standard support. It is important to take action – either by migrating to the next LTS or upgrading to Ubuntu Pro. After the end of support, your 18.04 LTS machines will not receive any security updates. With an Ubuntu Pro subscription, your Ubuntu 18.04 LTS deployment can receive Expanded Security Maintenance (ESM) until 2028. Security vulnerabilities that remain unpatched open your systems to hackers and the potential of a major breach.

Check ID

`c1dfprch15`

Alert Criteria

Red: An Amazon EC2 instance has an Ubuntu version that reached the end of standard support (Ubuntu 18.04 LTS, 18.04.1 LTS, 18.04.2 LTS, 18.04.3 LTS, 18.04.4 LTS, 18.04.5 LTS, and 18.04.6 LTS).

Yellow: An Amazon EC2 instance has an Ubuntu version that will reach the end of standard support in less than 6 months (Ubuntu 20.04 LTS, 20.04.1 LTS, 20.04.2 LTS, 20.04.3 LTS, 20.04.4 LTS, 20.04.5 LTS, and 20.04.6 LTS).

Green: All Amazon EC2 instances are compliant.

Recommended Action

To upgrade the Ubuntu 18.04 LTS instances to a supported LTS version, please follow the steps mentioned in this article. To upgrade the Ubuntu 18.04 LTS instances to Ubuntu Pro, visit Amazon License Manager console and follow the steps mentioned in the Amazon License Manager user guide. You can also refer to the Ubuntu blog showing a step by step demo of upgrading Ubuntu instances to Ubuntu Pro.

Additional Resources

For information about pricing, reach out to Amazon Web Services Support.

Report columns

- Status
- Region
- Ubuntu Lts Version
- Expected End Of Support Date
- Instance ID
- Support Cycle
- Last Updated Time

Amazon EFS clients not using data-in-transit encryption

Description

Checks if Amazon EFS file system is mounted using data-in-transit encryption. Amazon recommends that customers use data-in-transit encryption for all data flows to protect data from accidental exposure or unauthorized access. Amazon EFS recommends clients use the '-o tls' mount setting using the Amazon EFS mount helper to encrypt data in transit using TLS v1.2.

Check ID

`c1dfpnchv1`

Alert Criteria

Yellow: One or more NFS clients for your Amazon EFS file system are not using the recommended mount settings that provide data-in-transit encryption.

Green: All NFS clients for your Amazon EFS file system are using the recommended mount settings that provide data-in-transit encryption.

Recommended Action

To take advantage of data-in-transit encryption feature on Amazon EFS, we recommend that you remount your file system using the Amazon EFS mount helper and the recommended mount settings.

> **Note**
>
> Some distributions of Linux don't include a version of stunnel that supports TLS features by default. If you are using an unsupported Linux distribution (see supported distributions here), then we recommend that you upgrade it prior to remounting with the recommended mount setting.

Additional Resources

- Encrypting data in transit

Report columns

- Status
- Region
- EFS File System ID
- AZs with Unencrypted Connections
- Last Updated Time

Amazon Route 53 mismatching CNAME records pointing directly to S3 buckets

Description

Checks the Amazon Route 53 Hosted Zones with CNAME records pointing directly to Amazon S3 bucket hostnames and alerts if your CNAME does not match with your S3 bucket name.

Check ID

`c1ng44jvbm`

Alert Criteria

Red: Amazon Route 53 Hosted Zone has CNAME records pointing to mismatching S3 bucket hostnames.

Green: No mismatching CNAME records found in your Amazon Route 53 Hosted Zone.

Recommended Action

When pointing CNAME records to S3 bucket hostnames, you must make sure that a matching bucket exists for any CNAME or alias record you configure. By doing this, you avoid the risk

of your CNAME records being spoofed. You also prevent any unauthorized Amazon user from hosting faulty or malicous web content with your domain.

To avoid pointing CNAME records directly to S3 bucket hostnames, consider using origin access control (OAC) to access your S3 bucket web assets through Amazon CloudFront.

For more information about associating CNAME with an Amazon S3 bucket hostname, see Customizing Amazon S3 URLs with CNAME records.

Additional Resources

- How to associate a hostname with an Amazon S3 bucket
- Restricting access to an Amazon S3 origin with CloudFront

Report columns

- Status
- Hosted Zone ID
- Hosted Zone ARN
- Matching CNAME Records
- Mismatching CNAME Records
- Last Updated Time

Amazon S3 Bucket Permissions

Description

Checks buckets in Amazon Simple Storage Service (Amazon S3) that have open access permissions, or that allow access to any authenticated Amazon user.

This check examines explicit bucket permissions, as well as bucket policies that might override those permissions. Granting list access permissions to all users for an Amazon S3 bucket is not recommended. These permissions can lead to unintended users listing objects in the bucket at high frequency, which can result in higher than expected charges. Permissions that grant upload and delete access to everyone can lead to security vulnerabilities in your bucket.

Check ID

`Pfx0RwqBli`

Alert Criteria

- Yellow: The bucket ACL allows List access for **Everyone** or **Any Authenticated Amazon User**.

- Yellow: A bucket policy allows any kind of open access.
- Yellow: Bucket policy has statements that grant public access. The **Block public and cross-account access to buckets that have public policies** setting is turned on and has restricted access to only authorized users of that account until public statements are removed.
- Yellow: Trusted Advisor does not have permission to check the policy, or the policy could not be evaluated for other reasons.
- Red: The bucket ACL allows upload and delete access for **Everyone** or **Any Authenticated Amazon User**.

Recommended Action

If a bucket allows open access, determine if open access is truly needed. If not, update the bucket permissions to restrict access to the owner or specific users. Use Amazon S3 Block Public Access to control the settings that allow public access to your data. See Setting Bucket and Object Access Permissions.

Additional Resources

Managing Access Permissions to Your Amazon S3 Resources

Report columns

- Status
- Region Name
- Region API Parameter
- Bucket Name
- ACL Allows List
- ACL Allows Upload/Delete
- Policy Allows Access

ELB Listener Security

Description

Checks for load balancers with listeners that do not use recommended security configurations for encrypted communication. Amazon recommends using a secure protocol (HTTPS or SSL), up-to-date security policies, as well as ciphers and protocols that are secure.

When you use a secure protocol for a front-end connection (client to load balancer), the requests are encrypted between your clients and the load balancer, which create a more secure

environment. Elastic Load Balancing provides predefined security policies with ciphers and protocols that adhere to Amazon security best practices. New versions of predefined policies are released as new configurations become available.

Check ID

`a2sEc6ILx`

Alert Criteria

- Yellow: A load balancer has no listener that uses a secure protocol (HTTPS or SSL).
- Yellow: A load balancer listener uses an outdated predefined SSL security policy.
- Yellow: A load balancer listener uses a cipher or protocol that is not recommended.
- Red: A load balancer listener uses an insecure cipher or protocol.

Recommended Action

If the traffic to your load balancer must be secure, use either the HTTPS or the SSL protocol for the front-end connection.

Upgrade your load balancer to the latest version of the predefined SSL security policy.

Use only the recommended ciphers and protocols.

For more information, see Listener Configurations for Elastic Load Balancing.

Additional Resources

- Listener Configurations Quick Reference
- Update SSL Negotiation Configuration of Your Load Balancer
- SSL Negotiation Configurations for Elastic Load Balancing
- SSL Security Policy Table

Report columns

- Status
- Region
- Load Balancer Name
- Load Balancer Port
- Reason

ELB Security Groups

Description

Checks for load balancers configured with a missing security group, or a security group that allows access to ports that are not configured for the load balancer.

If a security group associated with a load balancer is deleted, the load balancer will not work as expected. If a security group allows access to ports that are not configured for the load balancer, the risk of loss of data or malicious attacks increases.

Check ID

`xSqX82fQu`

Alert Criteria

- Yellow: The inbound rules of an Amazon VPC security group associated with a load balancer allow access to ports that are not defined in the load balancer's listener configuration.
- Red: A security group associated with a load balancer does not exist.

Recommended Action

Configure the security group rules to restrict access to only those ports and protocols that are defined in the load balancer listener configuration, plus the ICMP protocol to support Path MTU Discovery. See Listeners for Your Classic Load Balancer and Security Groups for Load Balancers in a VPC.

If a security group is missing, apply a new security group to the load balancer. Create security group rules that restrict access to only those ports and protocols that are defined in the load balancer listener configuration. See Security Groups for Load Balancers in a VPC.

Additional Resources

- Elastic Load Balancing User Guide
- Configure Your Classic Load Balancer

Report columns

- Status
- Region
- Load Balancer Name
- Security Group IDs
- 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 Amazon 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

- Yellow: A password policy is enabled, but at least one content requirement is not enabled.
- Red: No password policy is 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.

Additional Resources

Managing Passwords

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 Amazon Directory Service are flagged as red or yellow, but they don't pose a security risk and can be safely ignored or excluded. For more information, see the Trusted Advisor FAQ.

> **Note**
>
> This check does not include the use case when a customer managed prefix list grants access to 0.0.0.0/0 and is used as a **source** with a security group.

Check ID

`HCP4007jGY`

Alert Criteria

- Green: Access to port 80, 25, 443, or 465 is unrestricted.
- Red: Access to port 20, 21, 1433, 1434, 3306, 3389, 4333, 5432, or 5500 is unrestricted.
- Yellow: Access to any other port is unrestricted.

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.

Additional Resources

- Amazon EC2 Security Groups

  List of TCP and UDP port numbers

- Classless Inter-Domain Routing

Report columns

- Status
- Region
- Security Group Name
- Security Group ID
- Protocol
- From Port
- To Port

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

> **Note**
>
> This check does not include the use case when a customer managed prefix list grants access to 0.0.0.0/0 and is used as a **source** with a security group.

Check ID

`1iG5NDGVre`

Alert Criteria

Red: A security group rule has a source IP address with a /0 suffix for ports other than 25, 80, or 443.

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.

Additional Resources

- Amazon EC2 Security Groups
- Classless Inter-Domain Routing

Report columns

- Status
- Region
- Security Group Name
- Security Group ID
- Protocol
- From Port
- To Port
- IP Range

Fault tolerance

You can use the following checks for the fault tolerance category.

Check names

- Amazon DocumentDB Single AZ Clusters
- Amazon EBS Snapshots
- Amazon ECS AmazonLogs driver in blocking mode
- Amazon ElastiCache Multi-AZ clusters
- Amazon MemoryDB Multi-AZ clusters
- Amazon RDS Backups
- Amazon S3 Bucket Logging

- Auto Scaling Group Health Check
- Auto Scaling Group Resources
- Amazon Direct Connect Location Resiliency
- ELB Connection Draining
- Load Balancer Optimization

Amazon DocumentDB Single AZ Clusters

Description

Checks if there are Amazon DocumentDB clusters configured as Single-AZ.

Running Amazon DocumentDB workloads in a Single-AZ architecture is not sufficient for highly critical workloads and it can take up to 10 minutes to recover from a component failure. Customers should deploy replica instances in additional availability zones to ensure availability during maintenance, instance failures, component failures, or availability zone failures.

> **Note**
>
> Results for this check are automatically refreshed one or more times each day, and refresh requests are not allowed. It might take a few hours for changes to appear. Currently, you can't exclude resources from this check.

Check ID

c15vnddn2x

Alert Criteria

Yellow: Amazon DocumentDB cluster has instances in less than three availability zones.

Green: Amazon DocumentDB cluster has instances in three availability zones.

Recommended Action

If your application requires high availability, modify your DB instance to enable Multi-AZ using replica instances. See Amazon DocumentDB High Availability and Replication

Additional Resources

Understanding Amazon DocumentDB Cluster Fault Tolerance

Regions and Availability Zones

Report columns

- Status
- Region
- Availability Zone
- DB Cluster Identifier
- DB Cluster ARN
- Last Updated Time

Amazon EBS Snapshots

Description

Checks the age of the snapshots for your Amazon Elastic Block Store (Amazon EBS) volumes (either available or in-use).

Even though Amazon EBS volumes are replicated, failures can occur. Snapshots are persisted to Amazon Simple Storage Service (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.

Additional Resources

Amazon Elastic Block Store (Amazon EBS)

Report columns

- Status

- Region
- Volume ID
- Volume Name
- Snapshot ID
- Snapshot Name
- Snapshot Age
- Volume Attachment
- Reason

Amazon ECS AmazonLogs driver in blocking mode

Description

Checks for Amazon ECS task definitions configured with the AmazonLogs logging driver in blocking mode. A driver configured in the blocking mode risks system availability.

> **Note**
>
> Results for this check are automatically refreshed one or more times each day, and refresh requests are not allowed. It might take a few hours for changes to appear. Currently, you can't exclude resources from this check.

Check ID

`c1dvkm4z6b`

Alert Criteria

Yellow: The awslogs driver logging configuration parameter mode is set to blocking or missing. A missing mode parameter indicates a default blocking configurations.

Green: Amazon ECS task definition is not using the awslogs driver or the awslogs driver is configured in non-blocking mode.

Recommended Action

To mitigate the availability risk, consider changing the task definition AmazonLogs driver configuration from blocking to non-blocking. With non-blocking mode, you will have to set a

value for the max-buffer-size parameter. For more information and guidance on configuration parameters, see . See Preventing log loss with non-blocking mode in the AmazonLogs container log driver

Additional Resources

Using the Amazon logs log driver

Choosing container logging options to avoid backpressure

Preventing log loss with non-blocking mode in the AmazonLogs container log driver

Report columns

- Status
- Region
- Task Definition ARN
- Container Definition Names
- Last Updated Time

Amazon ElastiCache Multi-AZ clusters

Description

Checks for ElastiCache clusters that deploy in a single Availability Zone (AZ). This check alerts you if Multi-AZ is inactive in a cluster.

Deployments in multiple AZs enhance ElastiCache cluster availability by asynchronously replicating to read-only replicas in a different AZ. When planned cluster maintenance occurs, or a primary node is unavailable, ElastiCache automatically promotes a replica to primary. This failover allows cluster write operations to resume, and doesn't require an administrator to intervene.

> **Note**
>
> Results for this check are automatically refreshed several times daily, and refresh requests are not allowed. It might take a few hours for changes to appear. Currently, you can’t exclude resources from this check.

Check ID

`ECHdfsQ402`

Alert Criteria

- Green: Multi-AZ is active in the cluster.
- Yellow: Multi-AZ is inactive in the cluster.

Recommended Action

Create at least one replica per shard, in an AZ that is different than the primary.

Additional Resources

For more information, see Minimizing downtime in ElastiCache for Redis with Multi-AZ.

Report columns

- Status
- Region
- Cluster Name
- Last Updated Time

Amazon MemoryDB Multi-AZ clusters

Description

Checks for MemoryDB clusters that deploy in a single Availability Zone (AZ). This check alerts you if Multi-AZ is inactive in a cluster.

Deployments in multiple AZs enhance MemoryDB cluster availability by asynchronously replicating to read-only replicas in a different AZ. When planned cluster maintenance occurs, or a primary node is unavailable, MemoryDB automatically promotes a replica to primary. This failover allows cluster write operations to resume, and doesn't require an administrator to intervene.

> **Note**
>
> Results for this check are automatically refreshed several times daily, and refresh requests are not allowed. It might take a few hours for changes to appear. Currently, you can’t exclude resources from this check.

Check ID

`MDBdfsQ401`

Alert Criteria

- Green: Multi-AZ is active in the cluster.
- Yellow: Multi-AZ is inactive in the cluster.

Recommended Action

Create at least one replica per shard, in an AZ that is different than the primary.

Additional Resources

For more information, see Minimizing downtime in MemoryDB with Multi-AZ.

Report columns

- Status
- Region
- Cluster Name
- Last Updated Time

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.

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 S3 Bucket Logging

Description

Checks the logging configuration of Amazon Simple Storage Service (Amazon S3) buckets.

When server access logging is enabled, detailed access logs are delivered hourly to a bucket that you choose. An access log record contains details about each request, such as the request type, the resources specified in the request, and the time and date the request was processed. By default, bucket logging is not enabled. You should enable logging if you want to perform security audits or learn more about users and usage patterns.

When logging is initially enabled, the configuration is automatically validated. However, future modifications can result in logging failures. This check examines explicit Amazon S3 bucket permissions, but it does not examine associated bucket policies that might override the bucket permissions.

Check ID

`BueAdJ7NrP`

Alert Criteria

- Yellow: The bucket does not have server access logging enabled.
- Yellow: The target bucket permissions do not include the root account, so Trusted Advisor cannot check it.
- Red: The target bucket does not exist.
- Red: The target bucket and the source bucket have different owners.
- Red: The log deliverer does not have write permissions for the target bucket.

Recommended Action

Enable bucket logging for most buckets. See Enabling Logging Using the Console and Enabling Logging Programmatically.

If the target bucket permissions do not include the root account and you want Trusted Advisor to check the logging status, add the root account as a grantee. See Editing Bucket Permissions.

If the target bucket does not exist, select an existing bucket as a target or create a new one and select it. See Managing Bucket Logging.

If the target and source have different owners, change the target bucket to one that has the same owner as the source bucket. See Managing Bucket Logging.

If the log deliverer does not have write permissions for the target (write not enabled), grant Upload/Delete permissions to the Log Delivery group. See Editing Bucket Permissions.

Additional Resources

- Working with Buckets
- Server Access Logging
- Server Access Log Format
- Deleting Log Files

Report columns

- Status
- Region
- Bucket Name
- Target Name
- Target Exists
- Same Owner
- Write Enabled
- Reason

Auto Scaling Group Health Check

Description

Examines the health check configuration for Auto Scaling groups.

If Elastic Load Balancing is being used for an Auto Scaling group, the recommended configuration is to enable an Elastic Load Balancing health check. If an Elastic Load Balancing health check is not used, Auto Scaling can only act upon the health of the Amazon Elastic Compute Cloud (Amazon EC2) instance. Auto Scaling will not act on the application running on the instance.

Check ID

`CLOG40CD08`

Alert Criteria

- Yellow: An Auto Scaling group has an associated load balancer, but the Elastic Load Balancing health check is not enabled.
- Yellow: An Auto Scaling group does not have an associated load balancer, but the Elastic Load Balancing health check is enabled.

Recommended Action

If the Auto Scaling group has an associated load balancer, but the Elastic Load Balancing health check is not enabled, see Add an Elastic Load Balancing Health Check to your Auto Scaling Group.

If the Elastic Load Balancing health check is enabled, but no load balancer is associated with the Auto Scaling group, see Set Up an Auto-Scaled and Load-Balanced Application.

Additional Resources

Amazon EC2 Auto Scaling User Guide

Report columns

- Status
- Region
- Auto Scaling Group Name
- Load Balancer Associated
- Health Check

Auto Scaling Group Resources

Description

Checks the availability of resources associated with launch configurations and your Auto Scaling groups.

Auto Scaling groups that point to unavailable resources cannot launch new Amazon Elastic Compute Cloud (Amazon EC2) instances. When properly configured, Auto Scaling causes the number of Amazon EC2 instances to increase seamlessly during demand spikes, and decrease automatically during demand lulls. Auto Scaling groups and launch configurations that point to unavailable resources do not operate as intended.

Check ID

`8CNsSllI5v`

Alert Criteria

- Red: An Auto Scaling group is associated with a deleted load balancer.
- Red: A launch configuration is associated with a deleted Amazon Machine Image (AMI).

Recommended Action

If the load balancer has been deleted, either create a new load balancer or target group then associate it to the Auto Scaling group, or create a new Auto Scaling group without the load balancer. For information about creating a new Auto Scaling group with a new load balancer, see Set Up an Auto-Scaled and Load-Balanced Application. For information about creating a new Auto Scaling group without a load balancer, see Create Auto Scaling Group in Getting Started With Auto Scaling Using the Console.

If the AMI has been deleted, create a new launch template or launch template version using a valid AMI and associate it with an Auto Scaling group. See Create Launch Configuration in Getting Started With Auto Scaling Using the Console.

Additional Resources

- Troubleshooting Auto Scaling: Amazon EC2 AMIs
- Troubleshooting Auto Scaling: Load Balancer Configuration
- Amazon EC2 Auto Scaling User Guide

Report columns

- Status

- Region
- Auto Scaling Group Name
- Launch Type
- Resource Type
- Resource Name

Amazon Direct Connect Location Resiliency

Description

Checks the Amazon Direct Connect location resiliency associated with each of your virtual private gateways or transit gateways.

This check alerts you if any of your virtual private gateways or Direct Connect gateways aren't configured to use at least two Direct Connect locations. Lack of location resiliency can result in unexpected downtime and a poor connectivity experience.

> **Note**
>
> Results for this check are automatically refreshed several times daily, and refresh requests are not allowed. It might take a few hours for changes to appear.

Check ID

`c1dfpnchv2`

Alert Criteria

Red: The virtual private gateway or Direct Connect gateway doesn't have virtual interfaces configured to connect to devices across multiple Direct Connect locations.

Yellow: The virtual private gateway or Direct Connect gateway is set up with multiple virtual interfaces to connect to different devices within the same Direct Connect location. But it isn't configured to connect to devices across multiple Direct Connect locations.

Green: The virtual private gateway or Direct Connect gateway is configured to utilize at least two Direct Connect locations.

Recommended Action

To build Direct Connect location resiliency, you can configure the virtual private gateway or Direct Connect gateway to connect to at least two distinct Direct Connect locations. For more information, see Amazon Direct Connect Resiliency Recommendation.

Additional Resources

Amazon Direct Connect Resiliency Recommendations

Amazon Direct Connect Failover Test

Report columns

- Status
- Region
- Last Updated Time
- Resiliency Status
- Location
- Connection ID
- Gateway ID

ELB Connection Draining

Description

Checks for load balancers that do not have connection draining enabled.

When connection draining is not enabled and you deregister an Amazon EC2 instance from a load balancer, the load balancer stops routing traffic to that instance and closes the connection. When connection draining is enabled, the load balancer stops sending new requests to the deregistered instance but keeps the connection open to serve active requests.

Check ID

`7qGXsKIUw`

Alert Criteria

Yellow: Connection draining is not enabled for a load balancer.

Recommended Action

Enable connection draining for the load balancer. For more information, see Connection Draining and Enable or Disable Connection Draining for Your Load Balancer.

Additional Resources

Elastic Load Balancing Concepts

Report columns

- Status
- Region
- Load Balancer Name
- Reason

Load Balancer Optimization

Description

Checks your load balancer configuration.

To help increase the level of fault tolerance in Amazon Elastic Compute Cloud (Amazon EC2) when using Elastic Load Balancing , we recommend running an equal number of instances across multiple Availability Zones in a Region. A load balancer that is configured accrues charges, so this is a cost-optimization check as well.

Check ID

`iqdCTZKCUp`

Alert Criteria

- Yellow: A load balancer is enabled for a single Availability Zone.
- Yellow: A load balancer is enabled for an Availability Zone that has no active instances.
- Yellow: The Amazon EC2 instances that are registered with a load balancer are unevenly distributed across Availability Zones. (The difference between the highest and lowest instance counts in utilized Availability Zones is more than 1, and the difference is more than 20% of the highest count.)

Recommended Action

Ensure that your load balancer points to active and healthy instances in at least two Availability Zones. For more information, see Add Availability Zone.

If your load balancer is configured for an Availability Zone with no healthy instances, or if there is an imbalance of instances across the Availability Zones, determine if all the Availability Zones are necessary. Omit any unnecessary Availability Zones and ensure there is a balanced distribution of instances across the remaining Availability Zones. For more information, see Remove Availability Zone.

Additional Resources

- Availability Zones and Regions
- Managing Load Balancers
- Best Practices in Evaluating Elastic Load Balancing

Report columns

- Status
- Region
- Load Balancer Name
- # of Zones
- Zone a Instances
- Zone b Instances
- Zone c Instances
- Zone d Instances
- Zone e Instances
- Zone f Instances
- 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 Amazon Web Services 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 create 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

- DynamoDB Read Capacity
- DynamoDB Write Capacity
- EBS Active Snapshots
- EBS General Purpose SSD (gp2) Volume Storage
- EBS General Purpose SSD (gp3) Volume Storage
- EBS Magnetic (standard) Volume Storage
- EBS Provisioned IOPS (SSD) Volume Aggregate IOPS
- EBS Provisioned IOPS SSD (io1) Volume Storage
- EC2 Reserved Instance Leases
- EC2-VPC Elastic IP Address
- ELB Classic Load Balancers

- VPC
- VPC Internet Gateways

DynamoDB Read Capacity

Description

Checks for usage that is more than 80% of the DynamoDB provisioned throughput limit for reads per Amazon Web Services 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 Amazon Web Services account.

Check ID

`c5ftjdfkMr`

Additional Resources

DynamoDB quotas

EBS Active Snapshots

Description

Checks for usage that is more than 80% of the EBS active snapshots quota.

Check ID

`eI7KK0l7J9`

Additional Resources

Amazon EBS limits

EBS General Purpose SSD (gp2) Volume Storage

Description

Checks for usage that is more than 80% of the EBS General Purpose SSD (gp2) volume storage quota.

Check ID

dH7RR016J9

Additional Resources

Amazon EBS limits

EBS General Purpose SSD (gp3) Volume Storage

Description

Checks for usage that is more than 80% of the EBS General Purpose SSD (gp3) volume storage quota.

Check ID

dH7RR016J3

Additional Resources

Amazon EBS limits

EBS Magnetic (standard) Volume Storage

Description

Checks for usage that is more than 80% of the EBS Magnetic (standard) volume storage quota.

Check ID

cG7HH017J9

Additional Resources

Amazon EBS limits

EBS Provisioned IOPS (SSD) Volume Aggregate IOPS

Description

Checks for usage that is more than 80% of the EBS Provisioned IOPS (SSD) volume aggregate IOPS quota.

Check ID

`tV7YY0l7J9`

Additional Resources

Amazon EBS limits

EBS Provisioned IOPS SSD (io1) Volume Storage

Description

Checks for usage that is more than 80% of the EBS Provisioned IOPS SSD (io1) volume storage quota.

Check ID

`gI7MM0l7J9`

Additional Resources

Amazon EBS limits

EC2 Reserved Instance Leases

Description

Checks for usage that is more than 80% of the EC2 Reserved Instance leases quota.

Check ID

`iH7PP0l7J9`

Additional Resources

Amazon EC2 quotas

EC2-VPC Elastic IP Address

Description

Checks for usage that is more than 80% of the EC2-VPC Elastic IP address quota.

Check ID

`lN7RR0l7J9`

Additional Resources

VPC Elastic IP quotas

ELB Classic Load Balancers

Description

Checks for usage that is more than 80% of the ELB Classic Load Balancers quota.

Check ID

`iK7O0l7J9`

Additional Resources

Elastic Load Balancing quotas

VPC

Description

Checks for usage that is more than 80% of the VPC quota.

Check ID

`jL7PP0l7J9`

Additional Resources

VPC quotas

VPC Internet Gateways

Description

Checks for usage that is more than 80% of the VPC Internet gateways quota.

Check ID

`kM7QQ017J9`

Additional Resources

VPC quotas

Change log for Amazon Trusted Advisor

See the following topic for recent changes to Trusted Advisor checks.

> **Note**
>
> If you use the Trusted Advisor console or the Amazon Web Services Support API, checks that were removed won't appear in check results. If you use any of the removed checks such as specifying the check ID in an Amazon Web Services Support API operation or your code, you must remove these checks to avoid API call errors.

For more information about the available checks, see the Amazon 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 Amazon Trusted Advisor check reference.

New fault tolerance check

Trusted Advisor added 1 Fault Tolerance check on January 31, 2024:

- Amazon Direct Connect Location Resiliency

For more information, see the Amazon 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 Amazon Trusted Advisor check reference.

Updated security check

Trusted Advisor amended 1 Security check on December 21, 2023:

- Amazon Lambda Functions Using Deprecated Runtimes

For more information, see the Amazon 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 Amazon 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 Amazon 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 AmazonLogs driver in blocking mode

For more information, see the Amazon 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 Amazon Security Hub

Trusted Advisor made the following update on November 17, 2022.

If you disable Security Hub or Amazon Config for an Amazon Web Services Region, Trusted Advisor now removes your control findings for that Amazon Web Services Region within 7-9 days. Previously, the time frame to remove your Security Hub data from Trusted Advisor was 90 days.

For more information, see the following sections in the Troubleshooting topic:

- I turned off Security Hub or Amazon Config in a Region
- My control is archived in Security Hub, 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 Amazon Web Services 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 Amazon Web Services Support API as usual.

For more information, see Get started with Trusted Advisor Recommendations.

Added Security Hub checks to Trusted Advisor

As of June 23, 2022, Trusted Advisor only supports Security Hub controls available through April 7, 2022. This release supports all controls in the Amazon Foundational Security Best Practices security standard except for controls in the Category: Recover > Resilience. For more information, see Viewing Amazon Security Hub controls in Amazon Trusted Advisor.

For a list of supported controls, see Amazon Foundational Security Best Practices controls in the *Amazon Security Hub User Guide*.

Added checks from Amazon 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`

Check name	Check category	Check ID
Amazon Lambda over-provisioned functions for memory size	Cost optimization	COr6dfpM05
Amazon Lambda under-provisioned functions for memory size	Performance	COr6dfpM06

You must opt in your Amazon Web Services account for Compute Optimizer so that these checks can receive data from your Lambda and Amazon EBS resources. For more information, see Opt in Amazon Compute Optimizer for Trusted Advisor checks.

Updated checks for Amazon Direct Connect

Trusted Advisor updated the following checks on March 29, 2022.

Check name	Check category	Check ID
Amazon Direct Connect Connection Redundancy	Fault tolerance	0t121N1Ty3
Amazon Direct Connect Location Redundancy	Fault tolerance	8M012Ph3U5
Amazon Direct Connect Virtual Interface Redundancy	Fault tolerance	4g3Nt5M1Th

- The value for the **Region** column now shows the Amazon Web Services 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.

Check name	Check category	Check ID
Amazon OpenSearch Service Reserved Instance Optimization	Cost optimization	`7ujm6yhn5t`

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 Amazon Trusted Advisor metrics.

Added checks for Amazon Elastic Block Store volume storage

Trusted Advisor added the following checks on June 8, 2021.

Check name	Check category	Check ID
EBS General Purpose SSD (gp3) Volume Storage	Service limits	`dH7RR0l6J3`

Added checks for Amazon Lambda

Trusted Advisor added the following checks on March 8, 2021.

Check name	Check category	Check ID
Amazon Lambda Functions with Excessive Timeouts	Cost optimization	L4dfs2Q3C3
Amazon Lambda Functions with High Error Rates	Cost optimization	L4dfs2Q3C2
Amazon Lambda Functions Using Deprecated Runtimes	Security	L4dfs2Q4C5
Amazon Lambda VPC-enabled Functions without Multi-AZ Redundancy	Fault tolerance	L4dfs2Q4C6

For more information about how to use these checks with Lambda, see Example Amazon Trusted Advisor workflow to view recommendations in the *Amazon Lambda Developer Guide*.

Trusted Advisor check removal

Trusted Advisor removed the following check for the China (Beijing) Region on March 8, 2021.

Check name	Check category	Check ID
EC2 Elastic IP Addresses	Service limits	aW9HH0l8J6

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 Amazon 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	V77iOLlBqz
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
EBS Active Volumes	Service limits	fH7LL0l7J9

Amazon Elastic Block Store no longer has a limit on the number of volumes that you can provision.

You can monitor your Amazon EC2 instances and verify they are up to date by using Amazon 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

Amazon Web Services Support App in Slack

You can use the Amazon Web Services Support App to manage your Amazon Web Services support cases in Slack. You can invite your team members to chat channels, respond to case updates, and chat directly with support agents. The Amazon Web Services Support App helps you manage support cases quickly and directly in Slack.

You can use the Amazon Web Services Support App to do the following:

- Create, update, search for, and resolve support cases in Slack channels
- Attach files to support cases
- Request quota increases from Service Quotas
- 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 Amazon Web Services Support App, the case is also updated in the Amazon 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, service quota increases, and technical support.

Topics

- Prerequisites
- Authorize a Slack workspace
- Configuring a Slack channel
- Creating support cases in a Slack channel
- Replying to support cases in Slack

- Joining a live chat session with Amazon Web Services Support
- Searching for support cases in Slack
- Resolving a support case in Slack
- Reopening a support case in Slack
- Requesting service quota increases
- Deleting a Slack channel configuration from the Amazon Web Services Support App
- Deleting a Slack workspace configuration from the Amazon Web Services Support App
- Amazon Web Services Support App in Slack commands
- View Amazon Web Services Support App correspondences in the Amazon Support Center Console
- Creating Amazon Web Services Support App in Slack resources with Amazon CloudFormation

Prerequisites

You must meet the following requirements to use the Amazon Web Services Support App in Slack:

- You have a Business, Enterprise On-Ramp, or Enterprise Support plan. You can find your support plan from the Amazon Support Center Console or from the Support plans page. For more information, see Compare Amazon Web Services 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 Amazon Web Services account as an Amazon Identity and Access Management (IAM) user or role with the required permissions. For more information, see Managing access to the Amazon Web Services Support App widget.
- You will need to create an IAM role that has the required permissions to perform actions for you. The Amazon Web Services Support App uses this role to make API calls to different services. For more information, see Managing access to the Amazon Web Services Support App.

Topics

- Managing access to the Amazon Web Services Support App widget
- Managing access to the Amazon Web Services Support App

Managing access to the Amazon Web Services Support App widget

You can attach an Amazon Identity and Access Management (IAM) policy to grant an IAM user permission to configure the Amazon Web Services Support App widget in the Amazon 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 Amazon Web Services 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.

```
{
    "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 Amazon Web Services Support App to Slack

The Amazon Web Services 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 Amazon Web Services Support App uses the following API actions to connect to Slack and then lists your *public* Slack channels in the Amazon 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 Amazon CLI and Amazon SDKs.

Managing access to the Amazon Web Services Support App

After you have permissions to the Amazon Web Services Support App widget, you must also create an Amazon Identity and Access Management (IAM) role. This role performs actions from other Amazon Web Services for you, such as the Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services Support App creates for you.

Follow these procedures to create your IAM role and policy.

Topics

- Use an Amazon managed policy or create a customer managed policy
- Create an IAM role
- Troubleshooting

Use an Amazon managed policy or create a customer managed policy

To grant your role permissions, you can use either an Amazon managed policy or a customer managed policy.

Tip

If you don't want to create a policy manually, we recommend that you use an Amazon managed policy instead and skip this procedure. Managed policies automatically have the required permissions for the Amazon Web Services Support App. You don't need to update the policies manually. For more information, see Amazon managed policies for Amazon Web Services 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 Amazon Web Services Support App

1. Sign in to the Amazon Web Services Management Console and open the IAM console at https://console.amazonaws.cn/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 Amazon Web Services 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 Amazon managed policies, see Amazon managed policies for Amazon Web Services Support App in Slack.

You can update the policy to remove a permission from the Amazon Web Services Support App.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "servicequotas:GetRequestedServiceQuotaChange",
                "servicequotas:GetServiceQuota",
                "servicequotas:RequestServiceQuotaIncrease",
                "support:AddAttachmentsToSet",
```

```
                "support:AddCommunicationToCase",
                "support:CreateCase",
                "support:DescribeCases",
                "support:DescribeCommunications",
                "support:DescribeSeverityLevels",
                "support:InitiateChatForCase",
                "support:ResolveCase"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {"iam:AWSServiceName": "servicequotas.amazonaws.com"}
            }
        }
    ]
}
```

For descriptions for each action, see the following topics in the *Service Authorization Reference*:

- Actions, resources, and condition keys for Amazon Web Services Support
- Actions, resources, and condition keys for Service Quotas
- Actions, resources, and condition keys for Amazon 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 Amazon Web Services Support App

1. Sign in to the Amazon Web Services Management Console and open the IAM console at https://console.amazonaws.cn/iam/.
2. In the navigation pane, choose **Roles**, and then choose **Create role**.
3. For **Select trusted entity**, choose **Amazon Web Service**.
4. Choose **Amazon Web Services Support App**.
5. Choose **Next: Permissions**.

6. Enter the policy name. You can choose the Amazon 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 Amazon service in the *IAM User Guide*.

Troubleshooting

See the following topics to manage access to the Amazon Web Services Support App.

Contents

- I want to restrict specific users in my Slack channel from specific actions
- When I configure a Slack channel, I don't see the IAM role that I created
- My IAM role is missing a permission
- A Slack error says that my IAM role isn't valid
- The Amazon Web Services Support App says that I'm missing an IAM role for Service Quotas

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 Amazon Web Services account or an IAM user.

We recommend the following best practices:

- Configure private Slack channels with the Amazon Web Services 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 Amazon Web Services Support App. See Amazon managed policies for Amazon Web Services 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 Amazon Web Services Support App** list, this means that the role doesn't have the Amazon Web Services 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 Amazon Web Services Support App assumes this role to perform these actions for you.

If you receive an error about a missing permission from the Amazon Web Services 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 Amazon Support Center Console at https://console.amazonaws.cn/support/app#/config page.
2. Choose the channel that you configured with the Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services Support App to create one for you.

Authorize a Slack workspace

After you authorize your workspace and give the Amazon Web Services Support App permission to access it, you then need an Amazon Identity and Access Management (IAM) role for your Amazon Web Services account. The Amazon Web Services Support App uses this role to call API operations from Amazon Web Services Support and Service Quotas for you. For example, the Amazon Web Services Support App uses the role to call the `CreateCase` operation to create a support case for you in Slack.

Notes

- The Slack channel inherits permissions from the IAM role. This means that any user in the Slack channel has the same permissions that are specified in the IAM policy that is attached to the role.

  For example, if your IAM policy allows the role to have 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 only have read permissions to 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 Amazon Web Services account. If you have multiple Amazon Web Services 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 Amazon Organizations and you want to authorize multiple accounts, skip to Authorize multiple accounts.

To authorize the Slack workspace for your Amazon Web Services account

1. Sign in to the **Amazon 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 **Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services accounts to use Slack workspaces, you can use Amazon CloudFormation or Terraform to create your Amazon Web Services Support App resources.

Configuring a Slack channel

After you authorize your Slack workspace, you can configure your Slack channels to use the Amazon Web Services Support App.

The channel where you invite and add the Amazon Web Services Support App is where you can create and search for cases, and receive case notifications. This channel shows case updates, such as newly created or resolved cases, added correspondences, and shared case details.

The Slack channel inherits permissions from the IAM role. This means that any user in the Slack channel has the same permissions that are specified in the IAM policy that is attached to the role.

For example, if your IAM policy allows the role to have 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 only have read permissions to your support cases.

You can add up to 20 channels for an account. A Slack channel can have up to 100 Amazon Web Services accounts. This means that only 100 accounts can add the same Slack channel to the Amazon Web Services Support App. We recommend that you only add the accounts that you need to manage support cases for your organization. This can reduce the number of notifications that you receive in the channel so that you and your team have fewer distractions.

Each Amazon Web Services account must configure a Slack channel separately in the Amazon Web Services Support App. This way, the Amazon Web Services Support App can access the support cases in that Amazon Web Services account. If another Amazon Web Services account in your organization already invited the Amazon Web Services 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 Amazon Web Services account can use the Amazon Web Services Support App. The Amazon Web Services 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 Amazon Web Services Support App.
2. Complete the following steps to invite the Amazon Web Services Support App to your channel:
   a. Choose the **+** icon and enter `invite`, and then, when prompted, choose **Add apps to this channel**.
   b. To search for the app, under **Add apps to channelName** enter **Amazon Web Services Support App**.
   c. Choose **Add** next to the **Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services Support App. This name appears only in your Amazon Web Services 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 Amazon Web Services Support App in Slack**, choose a role that you created for the Amazon Web Services Support App. Only roles that have the Amazon Web Services 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 Amazon Web Services 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 a severity.
   - **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 Amazon Web Services account and won't appear in Slack.
    - Choose **Delete** to delete the channel configuration from the Amazon Web Services Support App. See Deleting a Slack channel configuration from the Amazon Web Services 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 **Amazon Web Services 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 Amazon Web 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 Amazon Web Services Support agent, and **Current channel** will

use a thread in the current channel for you to chat with the Amazon Web Services 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 Amazon Web Services Support App will automatically add you and selected members to the new channel. For **Current channel**, the Amazon Web Services Support App will automatically tag you and selected members in the chat thread when the Amazon Web Services 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 Amazon Web Services Support App uses the same chat channel or thread that was used for a previous live chat. The Amazon Web Services 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 Amazon Support Center Console to reply to support agents. For more information, see Updating, resolving, and reopening your case.

- You cannot add correspondences to cases from chat channels created by the Amazon Web Services 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

Joining a live chat session with Amazon Web Services 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 Amazon Web Services 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. We recommend that you only add 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 will also receive notifications when a correspondence is added to the case outside of the live chat session. This will occur 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 where you invited the Amazon Web Services Support App to reply to these correspondences.

To join a live chat session with Amazon Web Services Support in a new channel

1. In the Slack application, navigate to the channel that the Amazon Web Services Support App creates for you. The channel name includes your support case ID, such as *awscase-1234567890*.

   Note

   The Amazon Web Services 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 won'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 Amazon Web Services Support App records the chat.

You can chat with the agent about your support case and upload any file attachments to the channel. The Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services Support in the current channel

1. In the Slack application, navigate to the thread in the current channel that the Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Support Center Console, you can use the Amazon Web Services 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:

   - **Amazon Web Services 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 Amazon Web Services 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 Amazon 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.

Requesting service quota increases

You can request service quota increases for your account from your Slack channel.

To request service quota increases

1. In the Slack channel, enter the following command:

   ```
   /awssupport quota
   ```

2. From the **Increase service quota** dialog box, enter the following information:

   a. Choose the **Amazon Web Services account**.

   b. Choose the **Amazon Web Services Region**.

   c. Choose the **Service name**.

   d. Choose the **Quota name**.

   e. Enter the **Requested value** for the quota increase. You must enter a value greater than the default quota.

3. Choose **Submit**.

Example : Quota increase for Alexa for Business

You can also view your requests from the Service Quotas console. For more information, see Requesting a quota increase in the *Service Quotas User Guide*.

Deleting a Slack channel configuration from the Amazon Web Services Support App

You can delete a channel configuration from the Amazon Web Services Support App if you don't need it. This action only removes the channel from the Amazon Web Services Support App and the Amazon Support Center Console. Your channel isn't deleted from Slack.

You can add up to 20 channels for your Amazon Web Services 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 Amazon Web Services Support App again later.

Deleting a Slack workspace configuration from the Amazon Web Services Support App

You can delete a workspace configuration from the Amazon Web Services Support App if you don't need it. This action only removes the workspace from the Amazon Web Services Support App and the Amazon Support Center Console. Your workspace isn't deleted from Slack.

You can add up to 5 workspaces for your Amazon Web Services 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 Amazon Web Services Support App, you must first delete these channels before you can delete the workspace. See Deleting a Slack channel configuration from the Amazon Web Services Support App.

To delete a Slack workspace configuration

1. Sign in to the **Amazon 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 Amazon Web Services account again later.

Amazon Web Services Support App in Slack commands

Slack channel commands

You can enter the following commands in the Slack channel where you invited the Amazon Web Services Support App. This Slack channel name also appears as a configured channel in the Amazon 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 Amazon Web Services accounts that configured the Amazon Web Services 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 Amazon Web Services Support App creates for you if you choose a new channel for your chat with Amazon Web Services 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 Amazon Web Services Support App correspondences in the Amazon 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 **Amazon 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 Amazon Web Services Support App in Slack resources with Amazon CloudFormation

Amazon Web Services Support App in Slack is integrated with Amazon CloudFormation, a service that helps you to model and set up your Amazon resources so that you can spend less time creating and managing your resources and infrastructure. You create a template that describes all the Amazon resources that you want (such as your AccountAlias and SlackChannelConfiguration), and Amazon CloudFormation provisions and configures those resources for you.

When you use Amazon CloudFormation, you can reuse your template to set up your Amazon Web Services Support App resources consistently and repeatedly. Describe your resources once, and then provision the same resources over and over in multiple Amazon Web Services accounts and Regions.

Amazon Web Services Support App and Amazon CloudFormation templates

To provision and configure resources for Amazon Web Services Support App and related services, you must understand Amazon CloudFormation templates. Templates are formatted text files in JSON or YAML. These templates describe the resources that you want to provision in your Amazon CloudFormation stacks. If you're unfamiliar with JSON or YAML, you can use Amazon CloudFormation Designer to help you get started with Amazon CloudFormation templates. For more information, see What is Amazon CloudFormation Designer? in the *Amazon CloudFormation User Guide*.

Amazon Web Services Support App supports creating your AccountAlias and SlackChannelConfiguration in Amazon CloudFormation. For more information, including examples of JSON and YAML templates for the AccountAlias and SlackChannelConfiguration resources, see the Amazon Web Services Support App resource type reference in the *Amazon CloudFormation User Guide*.

Create Slack configuration resources for your organization

You can use CloudFormation templates to create the resources that you need for the Amazon Web Services 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 Amazon 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services Support App resources in the Amazon Support Center Console.
- Create a support case with Amazon Web Services Support to authorize multiple accounts to use the Amazon Web Services 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 Amazon Web Services 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 Amazon managed policy is attached.

Contents

- Update your CloudFormation templates for Slack
- Create a stack for the management account
- Create a stack set for your organization

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 Amazon Web Services account in the Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services Support App.

To create a stack for the management account

1. Sign in to the Amazon Web Services Management Console as the management account for your organization.
2. Open the Amazon CloudFormation console at https://console.amazonaws.cn/cloudformation.

3. If you haven't already, in the **Region selector**, choose one of the following Amazon Web Services 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 Amazon 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 Amazon Web Services Management Console as the management account for your organization.
2. Open the Amazon CloudFormation console at https://console.amazonaws.cn/cloudformation.
3. If you haven't already, in the **Region selector**, choose the same Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services Region that you chose in step 3.
> - Choosing more than one Amazon Web Services 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 Amazon managed policy. This role has the required permissions to access other services for you. For more information, see Managing access to the Amazon Web Services 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 Amazon Web Services Support App for their Amazon Web Services 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 Amazon Web Services Region that you specified before.

Learn more about CloudFormation

To learn more about CloudFormation, see the following resources:

- Amazon CloudFormation
- Amazon CloudFormation User Guide
- Amazon CloudFormation API Reference
- Amazon CloudFormation Command Line Interface User Guide

Create Amazon Web Services Support App resources by using Terraform

You can also use Terraform to create the Amazon Web Services Support App resources for your Amazon Web Services account. Terraform is an infrastructure-as-code tool that you can use for your cloud applications. You can use Terraform to create Amazon Web Services Support App resources instead of deploying a CloudFormation stack to an account.

After you install Terraform, you can specify the Amazon Web Services 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 Amazon Web Services Support App resources for an OU in your organization.
- You can also find the event history for these updates from Terraform in Amazon CloudTrail. The `eventSource` for these events will be `cloudcontrolapi.amazonaws.com` and `supportapp.amazonaws.com`. For more information, see Logging Amazon Web Services Support App in Slack API calls using Amazon CloudTrail.

Learn more

To learn more about Terraform, see the following topics:

- Terraform installation
- Terraform tutorial: Build infrastructure for Amazon
- `awscc_support_app_account_alias`
- `awscc_supportapp_slack_workspace_configuration`
- `awscc_supportapp_slack_channel_configuration`

Security in Amazon Web Services Support

Cloud security at Amazon is the highest priority. As an Amazon 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 Amazon and you. The shared responsibility model describes this as security *of* the cloud and security *in* the cloud:

- **Security of the cloud** – Amazon is responsible for protecting the infrastructure that runs Amazon services in the Amazon Cloud. Amazon 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 Amazon compliance programs. To learn about the compliance programs that apply to Amazon Web Services Support, see Amazon Web Services in scope by compliance program.
- **Security in the cloud** – Your responsibility is determined by the Amazon service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when using Amazon Web Services Support. The following topics show you how to configure Amazon Web Services 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 Amazon Web Services Support resources.

Topics

- Data protection in Amazon Web Services Support
- Security for your Amazon Web Services Support cases
- Identity and access management for Amazon Web Services Support
- Incident response
- Logging and monitoring in Amazon Web Services Support and Amazon Trusted Advisor
- Compliance validation for Amazon Web Services Support
- Resilience in Amazon Web Services Support
- Infrastructure security in Amazon Web Services Support
- Configuration and vulnerability analysis in Amazon Web Services Support

Data protection in Amazon Web Services Support

The Amazon shared responsibility model applies to data protection in Amazon Web Services Support. As described in this model, Amazon is responsible for protecting the global infrastructure that runs all of the Amazon Web Services 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 Amazon Web Services that you use. For more information about data privacy, see the Data Privacy FAQ.

For data protection purposes, we recommend that you protect Amazon Web Services account credentials and set up individual users with Amazon IAM Identity Center or Amazon 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 Amazon resources. We require TLS 1.2 and recommend TLS 1.3.
- Set up API and user activity logging with Amazon CloudTrail.
- Use Amazon encryption solutions, along with all default security controls within Amazon Web 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 140-2 validated cryptographic modules when accessing Amazon through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-2.

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form text fields such as a **Name** field. This includes when you work with Amazon Web Services Support or other Amazon Web Services using the console, API, Amazon CLI, or Amazon 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.

Security for your Amazon Web Services Support cases

When you create a support case, you own the information that you include in your support case. Amazon doesn't access your Amazon Web Services account data without your permission. Amazon doesn't share your information with third parties.

When you create a support case, note the following:

- Amazon Web Services Support uses the permissions defined in the `AWSServiceRoleForSupport` service-linked role to call other Amazon Web Services that troubleshoot customer issues for you. For more information, see Using service-linked roles for Amazon Web Services Support and Amazon managed policy: AWSSupportServiceRolePolicy.
- You can view API calls to Amazon Web Services Support that occurred in your Amazon Web Services account. For example, you can view log information when someone in your account creates or resolves a support case. For more information, see Logging Amazon Web Services Support API calls with Amazon CloudTrail.
- You can use the Amazon Web Services 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 12 months after the case was created. For more information, see DescribeCases in the *Amazon Web Services Support API Reference*.
- Your support cases follow Compliance validation for Amazon Web Services Support.
- When you create a support case, Amazon 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. Amazon Web Services 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 Amazon Web Services Support FAQs.
- You can change your Amazon Web Services Support plan to get the help that you need for your account. For more information, see Compare Amazon Web Services Support Plans and Changing your Amazon Web Services Support plan.

Identity and access management for Amazon Web Services Support

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

Topics

- Audience
- Authenticating with identities
- Managing access using policies
- How Amazon Web Services Support works with IAM
- Amazon Web Services Support identity-based policy examples
- Using service-linked roles
- Amazon managed policies for Amazon Web Services Support
- Manage access to Amazon Web Services Support Center
- Manage access to Amazon Web Services Support Plans
- Manage access to Amazon Trusted Advisor
- Example Service Control Policies for Amazon Trusted Advisor
- Troubleshooting Amazon Web Services Support identity and access

Audience

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

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

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

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

Authenticating with identities

Authentication is how you sign in to Amazon using your identity credentials. You must be *authenticated* (signed in to Amazon) as the Amazon Web Services account root user, as an IAM user, or by assuming an IAM role.

If you access Amazon programmatically, Amazon provides a software development kit (SDK) and a command line interface (CLI) to cryptographically sign your requests by using your credentials. If you don't use Amazon tools, you must sign requests yourself. For more information about using the recommended method to sign requests yourself, see Signing Amazon API requests in the *IAM User Guide*.

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

Amazon account root user

When you create an Amazon Web Services account, you begin with one sign-in identity that has complete access to all Amazon Web Services and resources in the account. This identity is called the Amazon Web Services account *root user* and is accessed by signing in with the email address and password that you used to create the account. We strongly recommend that you don't use the root user for your everyday tasks. Safeguard your root user credentials and use them to perform the tasks that only the root user can perform. For the complete list of tasks that require you to sign in as the root user, see Tasks that require root user credentials in the *IAM User Guide*.

IAM users and groups

An *IAM user* is an identity within your Amazon Web Services account that has specific permissions for a single person or application. Where possible, we recommend relying on temporary credentials instead of creating IAM users who have long-term credentials such as passwords and access keys. However, if you have specific use cases that require long-term credentials with IAM users, we recommend that you rotate access keys. For more information, see Rotate access keys regularly for use cases that require long-term credentials in the *IAM User Guide*.

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

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

IAM roles

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

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

- **Federated user access** – To assign permissions to a federated identity, you create a role and define permissions for the role. When a federated identity authenticates, the identity is associated with the role and is granted the permissions that are defined by the role. For information about roles for federation, see Creating a role for a third-party Identity Provider in the *IAM User Guide*.
- **Temporary IAM user permissions** – An IAM user or role can assume an IAM role to temporarily take on different permissions for a specific task.
- **Cross-account access** – You can use an IAM role to allow someone (a trusted principal) in a different account to access resources in your account. Roles are the primary way to grant cross-

account access. However, with some Amazon Web Services, you can attach a policy directly to a resource (instead of using a role as a proxy). To learn the difference between roles and resource-based policies for cross-account access, see How IAM roles differ from resource-based policies in the *IAM User Guide*.

- **Cross-service access** – Some Amazon Web Services use features in other Amazon Web Services. For example, when you make a call in a service, it's common for that service to run applications in Amazon EC2 or store objects in Amazon S3. A service might do this using the calling principal's permissions, using a service role, or using a service-linked role.
  - **Forward access sessions (FAS)** – When you use an IAM user or role to perform actions in Amazon, you are considered a principal. When you use some services, you might perform an action that then initiates another action in a different service. FAS uses the permissions of the principal calling an Amazon Web Service, combined with the requesting Amazon Web Service to make requests to downstream services. FAS requests are only made when a service receives a request that requires interactions with other Amazon Web Services or resources to complete. In this case, you must have permissions to perform both actions. For policy details when making FAS requests, see Forward access sessions.
  - **Service role** – A service role is an IAM role that a service assumes to perform actions on your behalf. An IAM administrator can create, modify, and delete a service role from within IAM. For more information, see Creating a role to delegate permissions to an Amazon Web Service in the *IAM User Guide*.
  - **Service-linked role** – A service-linked role is a type of service role that is linked to an Amazon Web Service. The service can assume the role to perform an action on your behalf. Service-linked roles appear in your Amazon Web Services account and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.
- **Applications running on Amazon EC2** – You can use an IAM role to manage temporary credentials for applications that are running on an EC2 instance and making Amazon CLI or Amazon API requests. This is preferable to storing access keys within the EC2 instance. To assign an Amazon role to an EC2 instance and make it available to all of its applications, you create an instance profile that is attached to the instance. An instance profile contains the role and enables programs that are running on the EC2 instance to get temporary credentials. For more information, see Using an IAM role to grant permissions to applications running on Amazon EC2 instances in the *IAM User Guide*.

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

Managing access using policies

You control access in Amazon by creating policies and attaching them to Amazon identities or resources. A policy is an object in Amazon that, when associated with an identity or resource, defines their permissions. Amazon evaluates these policies when a principal (user, root user, or role session) makes a request. Permissions in the policies determine whether the request is allowed or denied. Most policies are stored in Amazon as JSON documents. For more information about the structure and contents of JSON policy documents, see Overview of JSON policies in the *IAM User Guide*.

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

By default, users and roles have no permissions. To grant users permission to perform actions on the resources that they need, an IAM administrator can create IAM policies. The administrator can then add the IAM policies to roles, and users can assume the roles.

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

Identity-based policies

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

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

Other policy types

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

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

Multiple policy types

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

How Amazon Web Services Support works with IAM

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

For information about how to manage access for Amazon Web Services Support using IAM, see Manage access for Amazon Web Services Support.

Topics

- Amazon Web Services Support identity-based policies
- Amazon Web Services Support IAM roles

Amazon Web Services 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. Amazon Web Services 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 Amazon JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

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

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

Policy actions in Amazon Web Services 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. Amazon Web Services 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 Amazon Web Services Support actions, see Actions Defined by Amazon Web Services Support in the *IAM User Guide*.

Examples

To view examples of Amazon Web Services Support identity-based policies, see Amazon Web Services Support identity-based policy examples.

Amazon Web Services Support IAM roles

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

Using temporary credentials with Amazon Web Services 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 Amazon STS API operations such as AssumeRole or GetFederationToken.

Amazon Web Services Support supports using temporary credentials.

Service-linked roles

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

Amazon Web Services Support supports service-linked roles. For details about creating or managing Amazon Web Services Support service-linked roles, see Using service-linked roles for Amazon Web Services 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.

Amazon Web Services Support supports service roles.

Amazon Web Services Support identity-based policy examples

By default, IAM users and roles don't have permission to create or modify Amazon Web Services Support resources. They also can't perform tasks using the Amazon Web Services Management Console, Amazon CLI, or Amazon 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
- Using the Amazon Web Services Support console
- Allow users to view their own permissions

Policy best practices

Identity-based policies are very powerful. They determine whether someone can create, access, or delete Amazon Web Services Support resources in your account. When you create or edit identity-based policies, follow these guidelines and recommendations:

- **Get Started Using Amazon Managed Policies** – To start using Amazon Web Services Support quickly, use Amazon managed policies to give your employees the permissions they need. These policies are already available in your account and are maintained and updated by Amazon. For more information, see Get started using permissions with Amazon 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 Amazon in the *IAM User Guide*.

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

Using the Amazon Web Services Support console

To access the Amazon Web Services Support console, you must have a minimum set of permissions. These permissions must allow you to list and view details about the Amazon Web Services Support resources in your Amazon 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 Amazon Web Services Support console, also attach the following Amazon 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 Amazon CLI or the Amazon 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 Amazon CLI or Amazon API.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
```

```
                "iam:GetUser"
            ],
            "Resource": ["arn:aws-cn: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

Amazon Web Services Support and Amazon Trusted Advisor use Amazon Identity and Access Management (IAM) service-linked roles. A service-linked role is a unique IAM role that is linked directly to Amazon Web Services Support and Trusted Advisor. In each case, the service-linked role is a predefined role. This role includes all the permissions that Amazon Web Services Support or Trusted Advisor require to call other Amazon services on your behalf. The following topics explain what service-linked roles do and how to work with them in Amazon Web Services Support and Trusted Advisor.

Topics

- Using service-linked roles for Amazon Web Services Support
- Using service-linked roles for Trusted Advisor

Using service-linked roles for Amazon Web Services Support

Amazon Web Services Support tools gather information about your Amazon resources through API calls to provide customer service and technical support. To increase the transparency and

auditability of support activities, Amazon Web Services Support uses an Amazon Identity and Access Management (IAM) service-linked role.

The `AWSServiceRoleForSupport` service-linked role is a unique IAM role that is linked directly to Amazon Web Services Support. This service-linked role is predefined, and it includes the permissions that Amazon Web Services Support requires to call other Amazon 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 Amazon Web Services Support access to resource metadata, not customer data. Only Amazon Web Services Support tools can assume this role, which exists within your Amazon account.

We redact fields that could contain customer data. For example, the `Input` and `Output` fields of the GetExecutionHistory for the Amazon Step Functions API call aren't visible to Amazon Web Services Support. We use Amazon KMS keys to encrypt sensitive fields. These fields are redacted in the API response and aren't visible to Amazon Web Services Support agents.

> **Note**
>
> Amazon Trusted Advisor uses a separate IAM service-linked role to access Amazon 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 Amazon Web Services Support API calls to be visible to customers through Amazon CloudTrail. This helps with monitoring and auditing requirements, because it provides a transparent way to understand the actions that Amazon Web Services Support performs on your behalf. For information about CloudTrail, see the Amazon CloudTrail User Guide.

Service-linked role permissions for Amazon Web Services Support

This role uses the `AWSSupportServiceRolePolicy` Amazon 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** – Amazon 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 Amazon account** – Amazon Web Services Support might use the permissions granted by the managed policy to access service attributes and usage data for your Amazon account. This policy allows Amazon Web Services 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** – Amazon Web Services 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**
>
> Amazon Web Services Support automatically updates the `AWSSupportServiceRolePolicy` policy once per month to add permissions for new Amazon services and actions.

For more information, see Amazon managed policies for Amazon Web Services Support.

Creating a service-linked role for Amazon Web Services Support

You don't need to manually create the `AWSServiceRoleForSupport` role. When you create an Amazon account, this role is automatically created and configured for you.

> **Important**
>
> If you used Amazon Web Services Support before it began supporting service-linked roles, then Amazon 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Command Line Interface (Amazon CLI). This protects your Amazon account, because you can't inadvertently remove necessary permissions for administering support services.

For more information about the `AWSServiceRoleForSupport` role or its uses, contact Amazon Web Services Support.

Using service-linked roles for Trusted Advisor

Amazon Trusted Advisor uses the Amazon Identity and Access Management (IAM) service-linked role. A service-linked role is a unique IAM role that is linked directly to Amazon Trusted Advisor. Service-linked roles are predefined by Trusted Advisor, and they include all the permissions that the service requires to call other Amazon services on your behalf. Trusted Advisor uses this role to check your usage across Amazon and to provide recommendations to improve your Amazon 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**
>
> Amazon Web Services 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 Amazon Web Services Support.

For information about other services that support service-linked roles, see Amazon 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 Amazon services on your behalf. The role permissions policy allows Trusted Advisor read-only access for all Amazon resources. This role simplifies getting started with your Amazon account, because you don't have to add the necessary permissions for Trusted Advisor. When you open an Amazon 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 Amazon 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 Amazon 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 Amazon 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 Amazon 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, Amazon 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 Amazon Web Services Management Console, and then navigate to the Trusted Advisor console at https://console.amazonaws.cn/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, Amazon 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 Amazon Web Services Management Console and navigate to the Trusted Advisor console at https://console.amazonaws.cn/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 Amazon 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*.

Amazon managed policies for Amazon Web Services Support

An Amazon managed policy is a standalone policy that is created and administered by Amazon. Amazon 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 Amazon managed policies might not grant least-privilege permissions for your specific use cases because they're available for all Amazon 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 Amazon managed policies. If Amazon updates the permissions defined in an Amazon managed policy, the update affects all principal identities (users, groups, and roles) that the policy is attached to. Amazon is most likely to update an Amazon managed policy when a new Amazon Web Service is launched or new API operations become available for existing services.

For more information, see Amazon managed policies in the *IAM User Guide*.

Topics

- Amazon managed policies for Amazon Web Services Support
- Amazon managed policies for Amazon Web Services Support App in Slack
- Amazon Web Services managed policies for Amazon Trusted Advisor
- Amazon managed policies for Amazon Web Services Support Plans

Amazon managed policies for Amazon Web Services Support

Amazon Web Services Support has the following managed policies.

Contents

- Amazon managed policy: AWSSupportServiceRolePolicy
- Amazon Web Services Support updates to Amazon managed policies

- Permission changes for AWSSupportServiceRolePolicy

Amazon managed policy: AWSSupportServiceRolePolicy

Amazon Web Services Support uses the AWSSupportServiceRolePolicy Amazon 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 Amazon Web Services Support.

For a list of changes to the policy, see Amazon Web Services Support updates to Amazon managed policies and Permission changes for AWSSupportServiceRolePolicy.

Amazon Web Services Support updates to Amazon managed policies

View details about updates to Amazon managed policies for Amazon Web Services 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 Amazon Web Services Support managed policies since February 17, 2022.

Amazon Web Services Support

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: • Amazon Clean Rooms – To troubleshoot issues	Jan 17, 2024

Change	Description	Date
	related to the Amazon Clean Rooms. • Amazon CodeStar Connections – To troubleshoot issues related to Amazon CodeStar Connections. • 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. • Amazon Outposts – To troubleshoot issues related to the Amazon Outposts. • Amazon RDS – To debug issues related to Amazon RDS. • Amazon IAM Identity Center – To troubleshoot issues related to Amazon IAM Identity Center.	

Change	Description	Date
	• Amazon S3 Express – To debug issues related to Amazon S3 Express. • Amazon Trusted Advisor – To troubleshoot issues related to Amazon 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: • Amazon Direct Connect – To troubleshoot issues related to the Amazon Direct Connect service. • Amazon SageMaker – To troubleshoot issues related to Amazon SageMaker service. • Amazon AppStream – To debug issues related to Amazon AppStream. • Amazon Resource Explorer – To debug issues related to the Amazon 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 troubleshoot issues related to the Amazon EC2. • Amazon Elastic Kubernetes Service – To debug issues related to Amazon Elastic Kubernetes Service. • Amazon Elastic Disaster Recovery – To troubleshoot issues related to Amazon Elastic Disaster Recovery. • Amazon AppSync – To debug issues related to Amazon AppSync. • Amazon CloudWatch Logs – To troubleshoot issues related to Amazon CloudWatch Logs. • Amazon Health – To debug issues related to the Amazon Health Service. • Amazon Connect – To debug issues related to the Amazon Connect. • Amazon Snowball – To troubleshoot issues related to Amazon Snowball. • Amazon HealthImaging – To troubleshoot issues related to Amazon HealthImaging.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 163 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administrative, and technical support: • Amazon CloudFront – To troubleshoot issues related to the CloudFront service. • Amazon EC2 – To troubleshoot issues related to Amazon EC2 service. • Amazon AppStream – To debug issues related to Amazon AppStream. • Amazon WAF – To debug issues related to the Amazon Web Application Firewall. • Amazon Connect – To troubleshoot issues related to Amazon Connect. • Amazon IoT – To debug issues related to the Amazon IoT. • Amazon Route 53 – To troubleshoot issues related to Amazon Route 53. • Amazon Verified Access – To troubleshoot issues related to the Amazon Verified Access service.	Oct 27, 2023

Change	Description	Date
	• Amazon Simple Email Service – To debug issues related to Amazon Simple Email Service. • Amazon Elastic Beanstalk – To troubleshoot issues related to Amazon Elastic Beanstalk. • Amazon DynamoDB – To debug issues related to Amazon DynamoDB. • Amazon EC2 Image Builder – To troubleshoot issues related to Amazon EC2 Image Builder. • Amazon Outposts – To debug issues related to the Amazon Outposts Service. • Amazon Glue – To debug issues related to the Amazon Glue. • Amazon Directory Service – To troubleshoot issues related to Amazon Directory Service. • Amazon Elastic Disaster Recovery – To troubleshoot issues related to Amazon Elastic Disaster Recovery. • Amazon Step Functions – To debug issues related to Amazon Step Functions.	

Change	Description	Date
	• Amazon EMR – To troublesh oot issues related to Amazon EMR. • Amazon Relational Database Service – To troubleshoot issues related 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: • Amazon Glue – To troubleshoot issues related to the Amazon Glue service • Amazon EMR – To troubleshoot issues related to Amazon EMR service. • Amazon Security Lake – To debug issues related to Amazon Security Lake. • Amazon Systems Manager – To debug issues related to the Systems Manager service. • Amazon Verified Permissions – To troubleshoot issues related to Amazon Verified Permissions. • Amazon IAM Access Analyzer – To debug issues related to the IAM Access Analyzer service. • Amazon Backup – To troubleshoot issues related to Amazon Backup. • Amazon Database Migration Service – To	Aug 28, 2023

Change	Description	Date
	troubleshoot 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. • Amazon Identity and Access Management – To troubleshoot issues related to Amazon Identity and Access Management. • Amazon Network Firewall – To troubleshoot issues related to Amazon Network Firewall.	

Change	Description	Date
	• Amazon HealthOmics – To debug issues related to Amazon HealthOmics. • Amazon QuickSight – To debug issues related to Amazon QuickSight. • 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 – To debug issues related to Amazon SageMaker.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added 141 new permissions to the following services to perform actions that help troubleshoot customer issues related to billing, administrative, and technical support: • Lambda – To troubleshoot issues related to Lambda service. • Amazon Lex – To troubleshoot issues related to Amazon Lex service. • Amazon Transfer – To debug issues related to Transfer service. • Amazon Amplify – To debug issues related to Amplify service. • Amazon EventBridge Pipes – To troubleshoot permissions 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. • Amazon Systems Manager – To troubleshoot issues	June 26, 2023

Change	Description	Date
	related to Systems Manager. • 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. • Amazon 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. • Amazon 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.	

Change	Description	Date
	• Amazon EventBridge Schemas – To debug issues related to EventBridge Schemas. • Amazon User Notifications – To troubleshoot issues related to User Notificat ions. • 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. • Amazon Compute Optimizer – To troublesh oot 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. • Amazon IoT TwinMaker – To troubleshoot issues related to Amazon 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
	• Amazon OAM Link – To debug issues related to OAM resources. • Amazon Outposts – To troubleshoot issues related to Amazon Outposts. • Amazon RDS – To debug issues related to Amazon RDS. • Amazon 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: • Amazon Backup gateway – To troubleshoot issues related to Backup gateway. • Amazon S3 – To debug issues related to Amazon S3. • Amazon Application Migration Service – To troubleshoot issues related to Application Migration Service. • Amazon Clean Rooms – To debug issues related to Amazon Clean Rooms; • Amazon Systems Manager for SAP – To troubleshoot issues related to Amazon 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 Amazon Web Services 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 Amazon	January 10, 2023

Change	Description	Date
	Web Services Support to query Amazon IVS resources to troubleshoot customer issues. • Amazon FSx – To enable Amazon Web Services Support to develop tools to support importing and exporting for an Amazon FSx data repository. • Amazon GameLift – To troubleshoot issues related to Amazon GameLift. • Amazon Glue– To troublesh oot issues related to Amazon 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. • Amazon Network Manager – To troubleshoot issues	

Change	Description	Date
	related to Network Manager. • 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. • Amazon 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: • Amazon Application Migration Service – To troubleshoot replication and launch issues. • Amazon CloudFormation hooks – To enable Amazon Web Services Support to develop automation tools that can help resolve issues. • Amazon Elastic Kubernete s Service – To troubleshoot issues related to Amazon EKS. • Amazon IoT FleetWise – To troubleshoot issues related to Amazon IoT FleetWise. • Amazon Mainframe Modernization – To debug issues related to Mainframe Modernization. • Amazon Outposts – To help Amazon Web Services Support get a list of dedicated hosts and assets. • Amazon Private 5G – To troubleshoot issues related to Private 5G.	October 4, 2022

Change	Description	Date
	• Amazon 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. • Amazon DataSync – To troubleshoot issues related to DataSync. • Amazon Elastic Disaster Recovery – To troublesh oot replication and launch issues. • Amazon GameSparks – To troubleshoot issues related to GameSparks. • Amazon IoT TwinMaker – To debug issues related to Amazon IoT TwinMaker. • Amazon Lambda – To view the configuration of a function URL to troublesh ooting 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 Amazon Web Services Support can check the DNS resolution behavior of a VPC.	

Change	Description	Date
AWSSupportServiceRolePolicy – Update to an existing policy	Added new permissions to the following services to perform actions that help troublesh oot 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 Amazon Web Services 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: • Amazon 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. • Amazon Backup – To troubleshoot issues related to backup jobs. • Amazon CloudFormation – To perform diagnostics on issues related to IAM, extension, and versioning. • Amazon Kinesis – To troubleshoot issues related to Kinesis. • Amazon 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, administr ative, and technical support:
	• Amazon Elastic Compute Cloud
	  • To troubleshoot issues related to customer and Amazon-managed prefixed lists.
	  • To troubleshoot issues related to Amazon VPC IP Address Manager (IPAM).
	• Amazon Network Manager – To troubleshoot issues related to Network Manager.
	• Savings Plans – To get metadata about outstanding Savings Plan commitments.
	• Amazon Serverless Application Repository – To improve and support response actions as part of researching and resolving support cases.
	• Amazon WorkSpaces Web – To debug and troublesh
		March 14, 2022

Change	Description	Date
	oot issues with WorkSpaces Web services.	

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, administrative, and technical support: • Amazon Application Migration Service – To support agentless replication in the Application Migration Service. • Amazon 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. • Amazon Elastic Disaster Recovery – To troubleshoot replication and launch problems in customer accounts. • Amazon FSx – To view the description of Amazon FSx snapshots. • Amazon Lightsail – To view metadata and configurations 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. • Amazon 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 Amazon Web Services Support managed policies.	February 17, 2022

Permission changes for AWSSupportServiceRolePolicy

Most permissions added to `AWSSupportServiceRolePolicy` allow Amazon Web Services 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:GetAnalyticsConfiguration
	s3.ListBucketAnalyticsConfiguration	
	s3.GetBucketNotificationConfiguration	s3:GetBucketNotification
	s3.GetBucketEncryption	s3:GetEncryptionConfiguration
	s3.GetBucketIntelligentTieringConfiguration	s3:GetIntelligentTieringConfiguration
	s3.ListBucketIntelligentTieringConfiguration	
	s3.GetBucketInventoryConfiguration	s3:GetInventoryConfiguration
	s3.ListBucketInventoryConfiguration	
	s3.GetBucketLifecycleConfiguration	s3:GetLifecycleConfiguration
	s3.GetBucketMetricsConfiguration	s3:GetMetricsConfiguration
	s3.ListBucketMetricsConfiguration	

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

Amazon managed policies for Amazon Web Services Support App in Slack

Note

To access and view support cases in the Amazon Support Center Console, see Manage access to Amazon Web Services Support Center.

Amazon Web Services Support App has the following managed policies.

Contents

- Amazon managed policy: AWSSupportAppFullAccess
- Amazon managed policy: AWSSupportAppReadOnlyAccess
- Amazon Web Services Support App updates to Amazon managed policies

Amazon 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 Amazon Web Services Support App in Slack.

This policy grants permissions that allow the entity to perform Amazon Web Services Support, Service Quotas, and IAM actions for the Amazon Web Services Support App.

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.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "servicequotas:GetRequestedServiceQuotaChange",
                "servicequotas:GetServiceQuota",
                "servicequotas:RequestServiceQuotaIncrease",
                "support:AddAttachmentsToSet",
                "support:AddCommunicationToCase",
                "support:CreateCase",
                "support:DescribeCases",
                "support:DescribeCommunications",
                "support:DescribeSeverityLevels",
                "support:InitiateChatForCase",
                "support:ResolveCase"
            ],
```

```
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {"iam:AWSServiceName": "servicequotas.amazonaws.com"}
            }
        }
    ]
}
```

For more information, see Managing access to the Amazon Web Services Support App.

Amazon managed policy: AWSSupportAppReadOnlyAccess

The AWSSupportAppReadOnlyAccess policy grants permissions that allow the entity to perform read-only Amazon Web Services Support App actions. For more information, see Amazon Web Services Support App in Slack.

Permissions details

This policy includes the following permissions:

- `support` – Describes support case details and communications added to the support cases.

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

Amazon Web Services Support App updates to Amazon managed policies

View details about updates to Amazon managed policies for Amazon Web Services 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 Amazon Web Services Support App managed policies since August 17, 2022.

Amazon Web Services Support App

Change	Description	Date
AWSSupportAppFullAccess and AWSSupportAppReadOnlyAccess New Amazon managed policies for the Amazon Web Services 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 Amazon Web Services Support App.	August 19, 2022
Change log published	Change log for the Amazon Web Services Support App managed policies.	August 19, 2022

Amazon Web Services managed policies for Amazon Trusted Advisor

Trusted Advisor has the following Amazon Web Services managed policies.

Contents

- Amazon managed policy: AWSTrustedAdvisorPriorityFullAccess
- Amazon managed policy: AWSTrustedAdvisorPriorityReadOnlyAccess
- Amazon managed policy: AWSTrustedAdvisorServiceRolePolicy
- Amazon managed policy: AWSTrustedAdvisorReportingServiceRolePolicy

- Trusted Advisor updates to Amazon managed policies

Amazon 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 Amazon 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 Amazon Organizations.

In the second statement, the policy includes the following permissions for `organizations`:

- Describes your Trusted Advisor account and organization.
- Lists the Amazon Web 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.

```
{
 "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"
     ]
    }
   }
  }
 ]
}
```

Amazon 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 Amazon Web Services that you enabled to use Organizations.
- Lists the delegated administrators for Trusted Advisor Priority

```
{
 "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"
      ]
     }
    }
   }
  ]
}
```

Amazon 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 Amazon 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 Amazon Web Services. These permissions allow the checks for Trusted Advisor to evaluate your account.

Permissions details

This policy includes the following permissions.

- `Auto Scaling` – Describes Amazon EC2 Auto Scaling account quotas and resources
- `cloudformation` – Describes Amazon CloudFormation (CloudFormation) account quotas and stacks
- `cloudfront` – Describes Amazon CloudFront distributions
- `cloudtrail` – Describes Amazon CloudTrail (CloudTrail) trails
- dynamodb – Describes Amazon DynamoDB account quotas and resources
- `ec2` – Describes Amazon Elastic Compute Cloud (Amazon EC2) account quotas and resources
- `elasticloadbalancing` – Describes Elastic Load Balancing (ELB) account quotas and resources
- `iam` – Gets IAM resources, such as credentials, password policy, and certificates
- `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 Amazon Outposts resources

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "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",
                "dynamodb:DescribeLimits",
                "dynamodb:DescribeTable",
                "dynamodb:ListTables",
```

```
                "ec2:DescribeAddresses",
                "ec2:DescribeReservedInstances",
                "ec2:DescribeInstances",
                "ec2:DescribeVpcs",
                "ec2:DescribeInternetGateways",
                "ec2:DescribeImages",
                "ec2:DescribeVolumes",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeRegions",
                "ec2:DescribeReservedInstancesOfferings",
                "ec2:DescribeSnapshots",
                "ec2:DescribeVpnConnections",
                "ec2:DescribeVpnGateways",
                "ec2:DescribeLaunchTemplateVersions",
                "ecs:DescribeTaskDefinition",
                "ecs:ListTaskDefinitions"
                "elasticloadbalancing:DescribeAccountLimits",
                "elasticloadbalancing:DescribeInstanceHealth",
                "elasticloadbalancing:DescribeLoadBalancerAttributes",
                "elasticloadbalancing:DescribeLoadBalancerPolicies",
                "elasticloadbalancing:DescribeLoadBalancerPolicyTypes",
                "elasticloadbalancing:DescribeLoadBalancers",
                "elasticloadbalancing:DescribeTargetGroups",
                "iam:GenerateCredentialReport",
                "iam:GetAccountPasswordPolicy",
                "iam:GetAccountSummary",
                "iam:GetCredentialReport",
                "iam:GetServerCertificate",
                "iam:ListServerCertificates",
                "kinesis:DescribeLimits",
                "kafka:ListClustersV2",
                "kafka:ListNodes",
                "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:ListQueues"
            ],
            "Resource": "*"
        }
    ]
}
```

Amazon 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 Amazon 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

```
{
    "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 Amazon managed policies

View details about updates to Amazon managed policies for Amazon Web Services 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 `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` Amazon managed policy to include statement IDs.	December 6, 2023
AWSTrustedAdvisorPriorityReadOnlyAccess Update to an existing policy.	Trusted Advisor updated the `AWSTrustedAdvisorPriorityReadOnlyAccess` Amazon managed policy to include statement IDs.	December 6, 2023

Change	Description	Date
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
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 Amazon 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

Change	Description	Date
AWSTrustedAdvisorPriorityFullAccess and AWSTrustedAdvisorPriorityReadOnlyAccess New Amazon 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
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 Amazon Web Services account.	August 10, 2021
Change log published	Trusted Advisor started tracking changes for its Amazon managed policies.	August 10, 2021

Amazon managed policies for Amazon Web Services Support Plans

Amazon Web Services Support Plans has the following managed policies.

Contents

- Amazon managed policy: AWSSupportPlansFullAccess
- Amazon managed policy: AWSSupportPlansReadOnlyAccess
- Amazon Web Services Support Plans updates to Amazon managed policies

Amazon managed policy: AWSSupportPlansFullAccess

Amazon Web Services Support Plans uses the AWSSupportPlansFullAccess Amazon managed policy. The IAM entity uses this policy to complete the following Support Plans actions for you:

- View your support plan for your Amazon Web Services account
- View details about the status for a request to change your support plan
- Change the support plan for your Amazon Web Services account
- Create support plan schedules for your Amazon Web Services account

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "supportplans:GetSupportPlan",
                "supportplans:GetSupportPlanUpdateStatus",
                "supportplans:StartSupportPlanUpdate",
                "supportplans:CreateSupportPlanSchedule"
            ],
            "Resource": "*"
        }
    ]
}
```

For a list of changes to the policies, see Amazon Web Services Support Plans updates to Amazon managed policies.

Amazon managed policy: AWSSupportPlansReadOnlyAccess

Amazon Web Services Support Plans uses the AWSSupportPlansReadOnlyAccess Amazon 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 Amazon Web Services account
- View details about the status for a request to change your support plan

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "supportplans:GetSupportPlan",
                "supportplans:GetSupportPlanUpdateStatus"
            ],
            "Resource": "*"
        }
    ]
}
```

For a list of changes to the policies, see Amazon Web Services Support Plans updates to Amazon managed policies.

Amazon Web Services Support Plans updates to Amazon managed policies

View details about updates to Amazon 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.

Amazon Web Services Support

Change	Description	Date
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

Manage access to Amazon Web Services 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 the email address and password associated with your Amazon account. This identity is called the Amazon account *root user*.
- Use Amazon Identity and Access Management (IAM).

If you have a Business, Enterprise On-Ramp, or Enterprise Support plan, you can also use the Amazon Web Services Support API to access Amazon Web Services Support and Trusted Advisor operations programmatically. For more information, see the Amazon Web Services 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.

Amazon account

You can sign in to the Amazon Web Services Management Console and access the Support Center by using your Amazon account email address and password. This identity is called the Amazon

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.

Amazon support actions

You can perform the following Amazon Web Services Support actions in the console. You can also specify these Amazon Web Services Support actions in an IAM policy to allow or deny specific actions.

> **Note**
>
> If you deny any of the below actions in your IAM policies, it could result in unintended behaviour in Support Center when creating or interacting with a support case.

Action	Description
`DescribeSupportLevel`	Grants permission to return the support level for an Amazon account identifier. This is used internally by Amazon Web Services Support Center to identify your support level.
`InitiateCallForCase`	Grants permission to initiate a call on Amazon Web Services Support Center. This is used internally by Amazon Web Services Support Center to start a call on your behalf.
`InitiateChatForCase`	Grants permission to initiate a chat on Amazon Web Services Support Center. This is used internally by Amazon Web Services Support Center to start a chat on your behalf.
`RateCaseCommunication`	Grants permission to rate a Amazon Web Services Support case communication.
`DescribeCaseAttributes`	Grants permission to allow secondary services to read Amazon Web Services Support case

Action	Description
	attributes. This is used internally by Amazon Web Services Support Center to get attributes tagged on your case.
`DescribeIssueTypes`	Grants permission to return issue types for Amazon Web Services Support cases. This is used internally by Amazon Web Services Support Center to get available issue types for your account.
`SearchForCases`	Grants permission to return a list of Amazon Web Services Support cases that matches the given inputs. This is used internally by Amazon Web Services Support Center to find searched cases.
`PutCaseAttributes`	Grants permission to allow secondary services to attach attributes to Amazon Web Services Support cases. This is used internally by Amazon Web Services Support Center to add operational tags to your Amazon Web Services Support cases.

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 Amazon Web Services 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 Amazon account and work in the Support Center. IAM users who have Amazon Web Services Support access can see all cases that are created for the account.

For more information, see How IAM users sign in to your Amazon account in the *IAM User Guide*.

The easiest way to grant permissions is to attach the Amazon managed policy AWSSupportAccess to the user, group, or role. Amazon Web Services Support allows action-level permissions to control access to specific Amazon Web Services Support operations. Amazon Web Services 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 Amazon Web Services Support actions

The Amazon managed policy AWSSupportAccess grants an IAM user access to Amazon Web Services Support. An IAM user with this policy can access all Amazon Web Services Support operations and resources.

```
{
    "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 Amazon Web Services Support except resolve a case.

```
{
   "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 Amazon Web Services 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 Amazon Web Services Support.

Access to Amazon Trusted Advisor

In the Amazon Web Services Management Console, a separate `trustedadvisor` IAM namespace controls access to Trusted Advisor. In the Amazon Web Services Support API, the `support` IAM namespace controls access to Trusted Advisor. For more information, see Manage access to Amazon Trusted Advisor.

Manage access to Amazon Web Services Support Plans

Topics

- Permissions for the Support Plans console
- Support Plans actions
- Example IAM policies for Support Plans
- Troubleshooting

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 Amazon Web Services account.

You can create an Amazon 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 Amazon managed policies and attach them to your IAM entities. For more information, see Amazon managed policies for Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services account.
`CreateSupportPlanSchedule`	Grants permission to create support plan schedules for this Amazon Web Services 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.

```
{
```

```
    "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.

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

Deny access to Support Plans

The following policy doesn't allow users access to Support Plans.

```
{
    "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 Amazon managed policy, such as AWSSupportPlansFullAccess or AWSSupportPlansReadOnlyAccess. For more information, see Amazon managed policies for Amazon Web Services Support Plans.

If you don't have access to update your IAM policies, contact your Amazon Web Services 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 Amazon Web Services account is a member account that's part of Amazon 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 Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services account administrator. The management account might need to update the SCP so that all member accounts can access Support Plans.

Notes for Amazon Control Tower

- If your organization uses an SCP with Amazon Control Tower, you can update the **Deny access to Amazon based on the requested Amazon Web Services Region** control (commonly referred to as the Region deny control).
- If you update the SCP for Amazon Control Tower to allow `supportplans`, repairing the drift will remove your update to the SCP. For more information, see Detect and resolve drift in Amazon Control Tower.

Related information

For more information, see the following topics:

- Service control policies (SCPs) in the *Amazon Organizations User Guide.*
- Configure the Region deny control in the *Amazon Control Tower User Guide*
- Deny access to Amazon based on the requested Amazon Web Services Region in the *Amazon Control Tower User Guide*

Manage access to Amazon Trusted Advisor

You can access Amazon Trusted Advisor from the Amazon Web Services Management Console. All Amazon Web Services accounts have access to a select core Trusted Advisor checks. If you have

a Business, Enterprise On-Ramp, or Enterprise Support plan, you can access all checks. for more information, see Amazon Trusted Advisor check reference.

You can use Amazon Identity and Access Management (IAM) to control access to Trusted Advisor.

Topics

- Permissions for the Trusted Advisor console
- Trusted Advisor actions
- IAM policy examples
- See also

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 Amazon Web Services 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 Amazon 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 Amazon Web Services Support API. You must use the `support` namespace for Amazon Web Services Support instead.

> **Note**
>
> If you have permissions to the Amazon Web Services Support API, the Trusted Advisor widget in the Amazon Web Services 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 Amazon Web Services Support plan and various Trusted Advisor preferences.
`DescribeAccountAccess`	Grants permission to view if the Amazon Web Services 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 Amazon account.
`ExcludeCheckItems`	Grants permission to exclude recommend ations for Trusted Advisor checks.
`IncludeCheckItems`	Grants permission to include recommend ations for Trusted Advisor checks.

Action	Description
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.
DescribeCheckStatusHistoryChanges	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 Amazon Trusted Advisor.

Action	Description
DescribeOrganization	Grants permission to view if the Amazon Web Services account meets the requirements to enable the organizational view feature.
DescribeOrganizationAccounts	Grants permission to view the linked Amazon accounts that are in the organization.
DescribeReports	Grants permission to view details for organizational view reports, such as the report name, runtime, date created, status, and format.
DescribeServiceMetadata	Grants permission to view information about organizational view reports, such as the Amazon Web Services Regions, check categories, check names, and resource statuses.

Action	Description
`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 Amazon 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 Amazon 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 Amazon 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.

Trusted Advisor Engage actions

If you have Trusted Advisor Engage 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 Engage .

For more information, see Get started with Amazon Trusted Advisor Engage (Preview).

Action	Description
`CreateEngagement`	Grants permission to create an engagement in Trusted Advisor Engage.
`CreateEngagementAttachment`	Grants permission to create an engagement attachment in Trusted Advisor Engage.
`CreateEngagementCommunication`	Grants permission to create an engagement communication in Trusted Advisor Engage.
`GetEngagement`	Grants permission to view an engagment in Trusted Advisor Engage.
`GetEngagementAttachment`	Grants permission to view an engagment attachment in Trusted Advisor Engage.
`GetEngagementType`	Grants permission to view a specific engagement type in Trusted Advisor Engage.
`ListEngagementCommunications`	Grants permission to view all communications for an engagement in Trusted Advisor Engage.
`ListEngagements`	Grants permission to view all engagements in Trusted Advisor Engage.
`ListEngagementTypes`	Grants permission to view all engagement types in Trusted Advisor Engage.
`UpdateEngagement`	Grants permission to update the details of an engagement in Trusted Advisor Engage.
`UpdateEngagementStatus`	Grants permission to update the status of an engagement in Trusted Advisor Engage.

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 Amazon Web Services Support API operations for Trusted Advisor
- Example IAM policies for Trusted Advisor Priority
- Example IAM policies for Trusted Advisor Engage

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.

```
{
    "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.

```
{
    "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.

```
{
    "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.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "trustedadvisor:*",
            "Resource": "*"
        },
        {
            "Effect": "Deny",
```

```
            "Action": "trustedadvisor:RefreshCheck",
            "Resource": "*"
        }
    ]
}
```

Control access to the Amazon Web Services Support API operations for Trusted Advisor

In the Amazon Web Services 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 Amazon Web Services Support API. Instead, you use the `support` namespace. You must have permissions to the Amazon Web Services 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 Amazon Web Services Support API operations for Trusted Advisor, but not the rest of the Amazon Web Services Support API operations. For example, users can use the API to view and refresh checks. They can't create, view, update, or resolve Amazon Web Services 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.

```
{
    "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 Amazon Web Services Support and Trusted Advisor, see Actions.

Example IAM policies for Trusted Advisor Priority

You can use the following Amazon managed policies to control access to Trusted Advisor Priority. For more information, see Amazon Web Services managed policies for Amazon Trusted Advisor and Get started with Amazon Trusted Advisor Priority.

Example IAM policies for Trusted Advisor Engage

> **Note**
>
> Trusted Advisor Engage is in preview release and does not currently have any Amazon managed policies. You can use one of the following policies to create a *customer managed policy* in the IAM console.

An example policy that grants read and write access in Trusted Advisor Engage:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": [
                "trustedadvisor:CreateEngagement*",
                "trustedadvisor:DescribeAccount*",
                "trustedadvisor:GetEngagement*",
                "trustedadvisor:ListEngagement*",
                "trustedadvisor:UpdateEngagement*"
            ],
            "Resource": "*"
        }
    ]
}
```

An example policy that grants read-only access in Trusted Advisor Engage:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "trustedadvisor:DescribeAccount*",
                "trustedadvisor:GetEngagement*",
                "trustedadvisor:ListEngagement*"
            ],
            "Resource": "*"
        }
    ]
}
```

An example policy that grants read and write access in Trusted Advisor Engage and the ability to enable trusted access to Trusted Advisor:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "organizations:DescribeOrganization",
                "organizations:ListAWSServiceAccessForOrganization",
                "trustedadvisor:CreateEngagement*",
                "trustedadvisor:DescribeAccount*",
```

```
                "trustedadvisor:DescribeOrganization",
                "trustedadvisor:GetEngagement*",
                "trustedadvisor:ListEngagement*",
                "trustedadvisor:SetOrganizationAccess",
                "trustedadvisor:UpdateEngagement*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "organizations:EnableAWSServiceAccess",
                "organizations:DisableAWSServiceAccess"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "organizations:ServicePrincipal": [
                        "reporting.trustedadvisor.amazonaws.com"
                    ]
                }
            }
        },
        {
            "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"
                }
            }
        }
    ]
}
```

See also

For more information about Trusted Advisor permissions, see the following resources:

- Actions defined by Amazon Trusted Advisor in the *IAM User Guide*.
- Controlling Access to the Trusted Advisor Console

Example Service Control Policies for Amazon Trusted Advisor

Amazon 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 Amazon Web Services 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 Amazon Web Services accounts stay within your organization's access control guidelines. For more information, see Service control policies in the *Amazon 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 *Amazon Organizations User Guide*.
- Enable SCPs for use within your organization. For more information, see Enabling and disabling policy types in the *Amazon 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 *Amazon 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.

```
{
  "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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "trustedadvisor:ListEngagement*",
        "trustedadvisor:GetEngagement*",
        "trustedadvisor:CreateEngagement*",
        "trustedadvisor:UpdateEngagement*",
        "trustedadvisor:DescribeRisk*",
        "trustedadvisor:UpdateRisk*",
        "trustedadvisor:DownloadRisk"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

Troubleshooting Amazon Web Services Support identity and access

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

Topics

- I'm not authorized to perform iam:PassRole
- I want to view my access keys
- I'm an administrator and want to allow others to access Amazon Web Services Support
- I want to allow people outside of my Amazon account to access my Amazon Web Services Support resources

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 Amazon Web Services Support.

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

The following example error occurs when an IAM user named `marymajor` tries to use the console to perform an action in Amazon Web Services 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-cn: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 Amazon 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 Amazon Web Services account.

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

I'm an administrator and want to allow others to access Amazon Web Services Support

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

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

I want to allow people outside of my Amazon account to access my Amazon Web Services 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 Amazon Web Services Support supports these features, see How Amazon Web Services Support works with IAM.

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

Incident response

Incident response for Amazon Web Services Support is an Amazon responsibility. Amazon has a formal, documented policy and program that governs incident response. For more information, see the Introducing the Amazon Security Incident Response Whitepaper.

Use the following options to inform yourself about operational issues:

- View Amazon operational issues with broad impact on the Amazon 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 Amazon Health Dashboard. For example, events that affect services or resources in your account. For more information, see Getting started with the Amazon Health Dashboard in the *Amazon Health User Guide*.

Logging and monitoring in Amazon Web Services Support and Amazon Trusted Advisor

Monitoring is an important part of maintaining the reliability, availability, and performance of Amazon Web Services Support and Amazon Trusted Advisor and your other Amazon solutions. Amazon provides the following monitoring tools to watch Amazon Web Services Support and Amazon Trusted Advisor, report when something is wrong, and take actions when appropriate:

- *Amazon CloudWatch* monitors your Amazon resources and the applications that you run on Amazon 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 Amazon resources. EventBridge enables automated event-driven computing, as you can write rules that watch for certain events and trigger automated actions in other Amazon services when these events happen. For more information, see the Amazon EventBridge User Guide.
- *Amazon CloudTrail* captures API calls and related events made by or on behalf of your Amazon 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 Amazon, the source IP address from which the calls were made, and when the calls occurred. For more information, see the Amazon CloudTrail User Guide.

For more information, see Monitoring and logging for Amazon Web Services Support and Monitoring and logging for Amazon Trusted Advisor.

Compliance validation for Amazon Web Services Support

To learn whether an Amazon Web Service is within the scope of specific compliance programs, see Amazon Web Services in Scope by Compliance Program and choose the compliance program that you are interested in. For general information, see Amazon Web Services Compliance Programs.

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

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

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying baseline environments on Amazon that are security and compliance focused.
- Amazon Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- Evaluating Resources with Rules in the *Amazon Config Developer Guide* – The Amazon Config service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.

- Amazon Security Hub – This Amazon Web Service provides a comprehensive view of your security state within Amazon. Security Hub uses security controls to evaluate your Amazon resources and to check your compliance against security industry standards and best practices. For a list of supported services and controls, see Security Hub controls reference.

Resilience in Amazon Web Services Support

The Amazon global infrastructure is built around Amazon Regions and Availability Zones. Amazon 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 Amazon Regions and Availability Zones, see Amazon global infrastructure.

Infrastructure security in Amazon Web Services Support

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

You use Amazon published API calls to access Amazon Web Services 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 Amazon Security Token Service (Amazon STS) to generate temporary security credentials to sign requests.

Configuration and vulnerability analysis in Amazon Web Services Support

For Amazon Trusted Advisor, Amazon 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 Amazon and you, our customer. For more information, see the Amazon shared responsibility model.

Code examples for Amazon Web Services Support using Amazon SDKs

The following code examples show how to use Amazon Web Services Support with an Amazon software development kit (SDK).

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 and cross-service examples.

Scenarios are code examples that show you how to accomplish a specific task by calling multiple functions within the same service.

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Get started

Hello Amazon Web Services Support

The following code examples show how to get started using Amazon Web Services Support.

.NET

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon 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.",
      );
    } else {
      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 *Amazon 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 Amazon 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 { 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 *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.

Code examples

- Actions for Amazon Web Services Support using Amazon SDKs
  - Add an Amazon Web Services Support communication to a case using an Amazon SDK
  - Add an Amazon Web Services Support attachment to a set using an Amazon SDK
  - Create an Amazon Web Services Support case using an Amazon SDK
  - Describe an attachment for an Amazon Web Services Support case using an Amazon SDK
  - Describe Amazon Web Services Support cases using an Amazon SDK
  - Describe Amazon Web Services Support communications for a case using an Amazon SDK

- Describe the available Amazon services for support cases using an Amazon SDK
- Describe Amazon Web Services Support severity levels using an Amazon SDK
- Resolve an Amazon Web Services Support case using an Amazon SDK
- Scenarios for Amazon Web Services Support using Amazon SDKs
  - Get started with Amazon Web Services Support cases using an Amazon SDK

Actions for Amazon Web Services Support using Amazon SDKs

The following code examples demonstrate how to perform individual Amazon Web Services Support actions with Amazon SDKs. These excerpts call the Amazon Web Services Support API and are code excerpts from larger programs that must be run in context. 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 Amazon Web Services Support API Reference.

Examples

- Add an Amazon Web Services Support communication to a case using an Amazon SDK
- Add an Amazon Web Services Support attachment to a set using an Amazon SDK
- Create an Amazon Web Services Support case using an Amazon SDK
- Describe an attachment for an Amazon Web Services Support case using an Amazon SDK
- Describe Amazon Web Services Support cases using an Amazon SDK
- Describe Amazon Web Services Support communications for a case using an Amazon SDK
- Describe the available Amazon services for support cases using an Amazon SDK
- Describe Amazon Web Services Support severity levels using an Amazon SDK
- Resolve an Amazon Web Services Support case using an Amazon SDK

Add an Amazon Web Services Support communication to a case using an Amazon SDK

The following code examples show how to add an Amazon Web Services Support communication with an attachment to a support case.

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:

- Get started with cases

.NET

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 { 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 *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Add an Amazon Web Services Support attachment to a set using an Amazon SDK

The following code examples show how to add an Amazon Web Services Support attachment to an attachment set.

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:

- Get started with cases

.NET

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.addAttachmentsToSet(setRequest)
        return response.attachmentSetId
    }
}
```

- For API details, see AddAttachmentsToSet in *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Create an Amazon Web Services Support case using an Amazon SDK

The following code examples show how to create a new Amazon Web Services Support case.

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:

- Get started with cases

.NET

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon 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 "en" (English) and "ja" (Japanese) 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 *Amazon SDK for .NET API Reference*.

CLI

Amazon CLI

To create a case

The following `create-case` example creates a support case for your Amazon 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 *Amazon Support User Guide*.

- For API details, see CreateCase in *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 { region = "us-west-2" }.use { supportClient ->
        val response = supportClient.createCase(caseRequest)
        return response.caseId
    }
}
```

- For API details, see CreateCase in *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Describe an attachment for an Amazon Web Services Support case using an Amazon SDK

The following code examples show how to describe an attachment for an Amazon Web Services Support case.

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:

- Get started with cases

.NET

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon 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 *Amazon SDK for .NET API Reference*.

CLI

Amazon 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 *Amazon Support User Guide*.

- For API details, see DescribeAttachment in *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon Code Examples Repository.

```
suspend fun describeAttachment(attachId: String?) {
    val attachmentRequest = DescribeAttachmentRequest {
        attachmentId = attachId
    }

    SupportClient { 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 *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Describe Amazon Web Services Support cases using an Amazon SDK

The following code examples show how to describe Amazon Web Services Support cases.

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:

- Get started with cases

.NET

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon 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 "en" (English) and "ja" (Japanese) 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 *Amazon SDK for .NET API Reference*.

CLI

Amazon CLI

To describe a case

The following `describe-cases` example returns information about the specified support case in your Amazon 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 *Amazon Support User Guide*.

- For API details, see DescribeCases in *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 { 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 *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Describe Amazon Web Services Support communications for a case using an Amazon SDK

The following code examples show how to describe Amazon Web Services Support communications for a case.

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:

- Get started with cases

.NET

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon 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 *Amazon SDK for .NET API Reference*.

CLI

Amazon 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 Amazon 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 *Amazon Support User Guide*.

- For API details, see DescribeCommunications in *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon Code Examples Repository.

```
suspend fun listCommunications(caseIdVal: String?): String? {
    val communicationsRequest = DescribeCommunicationsRequest {
        caseId = caseIdVal
        maxResults = 10
    }

    SupportClient { 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 *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Describe the available Amazon services for support cases using an Amazon SDK

The following code examples show how to describe the list of Amazon services.

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:

- Get started with cases

.NET

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon Code Examples Repository.

```
/// <summary>
/// Get the descriptions of AWS services.
```

```
    /// </summary>
    /// <param name="name">Optional language for services.
    /// Currently "en" (English) and "ja" (Japanese) 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 *Amazon SDK for .NET API Reference*.

CLI

Amazon CLI

To list Amazon 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 *Amazon Support User Guide*.

- For API details, see DescribeServices in *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 { 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 *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Describe Amazon Web Services Support severity levels using an Amazon SDK

The following code examples show how to describe Amazon Web Services Support severity levels.

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:

- Get started with cases

.NET

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon Code Examples Repository.

```
/// <summary>
/// Get the descriptions of support severity levels.
/// </summary>
/// <param name="name">Optional language for severity levels.
/// Currently "en" (English) and "ja" (Japanese) 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 *Amazon SDK for .NET API Reference*.

CLI

Amazon 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 *Amazon Support User Guide*.

- For API details, see DescribeSeverityLevels in *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon Code Examples Repository.

```
suspend fun displaySevLevels(): String {
    var levelName = ""
    val severityLevelsRequest = DescribeSeverityLevelsRequest {
        language = "en"
    }

    SupportClient { 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 *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Resolve an Amazon Web Services Support case using an Amazon SDK

The following code examples show how to resolve an Amazon Web Services Support case.

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:

- Get started with cases

.NET

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon 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 *Amazon SDK for .NET API Reference*.

CLI

Amazon CLI

To resolve a support case

The following `resolve-case` example resolves a support case in your Amazon 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 *Amazon Support User Guide*.

- For API details, see ResolveCase in *Amazon 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 Amazon 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 *Amazon 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 Amazon 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 *Amazon 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 Amazon Code Examples Repository.

```
suspend fun resolveSupportCase(caseIdVal: String) {
    val caseRequest = ResolveCaseRequest {
        caseId = caseIdVal
    }
    SupportClient { 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 *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Scenarios for Amazon Web Services Support using Amazon SDKs

The following code examples show you how to implement common scenarios in Amazon Web Services Support with Amazon SDKs. These scenarios show you how to accomplish specific tasks by calling multiple functions within Amazon Web Services Support. Each scenario includes a link to GitHub, where you can find instructions on how to set up and run the code.

Examples

- Get started with Amazon Web Services Support cases using an Amazon SDK

Get started with Amazon Web Services Support cases using an Amazon 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

Amazon SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the Amazon 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 Amazon Web Services 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 "en" (English) and "ja" (Japanese) 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 "en" (English) and "ja" (Japanese) 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 "en" (English) and "ja" (Japanese) 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 "en" (English) and "ja" (Japanese) 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 *Amazon 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 Amazon Code Examples Repository.

Run various Amazon Web Services 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 = 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 *Amazon 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 Amazon 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 inquirer from "inquirer";

// Retry an asynchronous function on failure.
const retry = async ({ intervalInMs = 500, maxRetries = 10 }, fn) => {
  try {
    return await fn();
  } catch (err) {
    console.log(`Function call failed. Retrying.`);
    console.error(err.message);
    if (maxRetries === 0) throw err;
    await new Promise((resolve) => setTimeout(resolve, intervalInMs));
    return retry({ intervalInMs, maxRetries: maxRetries - 1 }, fn);
  }
};

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."
      );
    } else {
      throw err;
    }
  }
};

// Get the list of available services.
export const getService = async () => {
  const { services } = await client.send(new DescribeServicesCommand({}));
  const { selectedService } = await inquirer.prompt({
    name: "selectedService",
    type: "list",
    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;
};

// Get the list of available support case categories for a service.
export const getCategory = async (service) => {
  const { selectedCategory } = await inquirer.prompt({
    name: "selectedCategory",
    type: "list",
    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.prompt({
    name: "selectedSeverityLevel",
    type: "list",
    message: "Select a severity level.",
    choices: severityLevels.map((s) => ({ name: s.name, value: s })),
  });
  return selectedSeverityLevel;
};

// Create a new support case and return the caseId.
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;
};

// Get an attachment set.
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.prompt({
    name: "shouldResolve",
    type: "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.
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 *Amazon 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 Amazon 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 { 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 { 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 { 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 { 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 { 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 { 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 { 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 { 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 { 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 { 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 *Amazon 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 Amazon 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 *Amazon SDK for Python (Boto3) API Reference*.
  - AddAttachmentsToSet
  - AddCommunicationToCase
  - CreateCase
  - DescribeAttachment
  - DescribeCases
  - DescribeCommunications
  - DescribeServices
  - DescribeSeverityLevels

- ResolveCase

For a complete list of Amazon SDK developer guides and code examples, see Using Amazon Web Services Support with an Amazon SDK. This topic also includes information about getting started and details about previous SDK versions.

Monitoring and logging for Amazon Web Services Support

Monitoring is an important part of maintaining the reliability, availability, and performance of Amazon Web Services Support and your other Amazon solutions. Amazon provides the following monitoring tools to watch Amazon Web Services 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 Amazon resources. EventBridge enables automated event-driven computing, as you can write rules that watch for certain events and trigger automated actions in other Amazon services when these events happen. For more information, see the Amazon EventBridge User Guide.
- *Amazon CloudTrail* captures API calls and related events made by or on behalf of your Amazon account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called Amazon, the source IP address from which the calls were made, and when the calls occurred. For more information, see the Amazon CloudTrail User Guide.

Topics

- Monitoring Amazon Web Services Support cases with Amazon EventBridge
- Logging Amazon Web Services Support API calls with Amazon CloudTrail
- Logging Amazon Web Services Support App in Slack API calls using Amazon CloudTrail

Monitoring Amazon Web Services Support cases with Amazon EventBridge

> **Note**
>
> This feature is not available in the China Regions.

You can use Amazon EventBridge to detect and react to changes for your Amazon Web Services Support cases. Then, based on the rules that you create, EventBridge invokes one or more target actions when an event matches the values that you specify in a rule.

Depending on the event, you can send notifications, capture event information, take corrective action, initiate events, or take other actions. For example, you can get notified whenever the following actions occur in your account:

- Create a support case
- Add a case correspondence to an existing support case
- Resolve a support case
- Reopen a support case

> **Note**
>
> Amazon Web Services Support delivers events on a best effort basis. Events are not always guaranteed to be delivered to EventBridge.

Creating an EventBridge rule for Amazon Web Services Support cases

You can create an EventBridge rule to get notified for Amazon Web Services Support case events. The rule will monitor updates for support cases in your account, including actions that you, your IAM users, or support agents perform. Before you create a rule for Amazon Web Services Support case events, 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 to use in your event rule. For example, you can create an Amazon Simple Notification Service (Amazon SNS) topic so that whenever a support case is updated, you will receive a text message or email. For more information, see EventBridge targets.

To create an EventBridge rule for Amazon Web Services Support case events

1. Open the Amazon EventBridge console at https://console.amazonaws.cn/events/.
2. If you haven't already, 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 **Amazon events or EventBridge partner events**.
8. Under **Event pattern**, keep the default value for **Amazon Web Services**.
9. For **Amazon Web Service**, choose **Support**.
10. For **Event type**, choose **Support Case Update**.
11. Choose **Next**.
12. In the **Select target(s)** section, choose the target that you created for this rule, and then configure any additional options that are required for that type. For example, if you choose Amazon SNS, make sure that your SNS topic is configured correctly so that you will be notified by email or SMS.
13. Choose **Next**.
14. (Optional) On the **Configure tags** page, add any tags and then choose **Next**.
15. On the **Review and create** page, review your rule setup and ensure that it meets your event monitoring requirements.
16. Choose **Create rule**. Your rule will now monitor for Amazon Web Services Support case events and then send them to the target that you specified.

Notes

- When you receive an event, you can use the `origin` parameter to determine whether you or an Amazon Web Services Support agent added a case correspondence to a support case. The value for `origin` can be either `CUSTOMER` or `Amazon`.

  Currently, only events for the `AddCommunicationToCase` action will have this value.
- For more information about creating event patterns, see Event patterns in the *Amazon EventBridge User Guide*.
- You can also create another rule for the **Amazon API Call via CloudTrail** event type. This rule will monitor Amazon CloudTrail logs for Amazon Web Services Support API calls in your account.

Example Amazon Web Services Support events

The following events are created when support actions occur in your account.

Example : Create support case

The following event is created when a support case is 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 : Update support case

The following event is created when Amazon Web Services 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 : Resolve support case

The following event is created when a support case is 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 : Reopen support case

The following event is created when a support case is 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": ""
    }
}
```

See also

For more information about how to use EventBridge with Amazon Web Services Support, see the following resources:

- How to automate Amazon Web Services Support API with Amazon EventBridge
- Amazon Web Services Support case activity notifier on GitHub

Logging Amazon Web Services Support API calls with Amazon CloudTrail

Amazon Web Services Support is integrated with Amazon CloudTrail, a service that provides a record of actions taken by a user, role, or an Amazon service in Amazon Web Services Support. CloudTrail captures API calls for Amazon Web Services Support as events. The calls captured include calls from the Amazon Web Services Support console and code calls to the Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon CloudTrail User Guide.

Amazon Web Services Support information in CloudTrail

CloudTrail is enabled on your Amazon account when you create the account. When supported event activity occurs in Amazon Web Services Support, that activity is recorded in a CloudTrail

event along with other Amazon service events in **Event history**. You can view, search, and download recent events in your Amazon account. For more information, see Viewing events with CloudTrail event history.

For an ongoing record of events in your Amazon account, including events for Amazon Web Services 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 Amazon Regions. The trail logs events from all Regions in the Amazon partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other Amazon 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 Amazon Web Services Support API operations are logged by CloudTrail and are documented in the Amazon Web Services 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 Amazon 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 Amazon service.

For more information, see the CloudTrail userIdentity element.

You can also aggregate Amazon Web Services Support log files from multiple Amazon Regions and multiple Amazon accounts into a single Amazon S3 bucket.

Amazon Trusted Advisor information in CloudTrail logging

Trusted Advisor is an Amazon Web Services Support service that you can use to check your Amazon 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 Amazon Web Services 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 Amazon Trusted Advisor console actions with Amazon CloudTrail.

Understanding Amazon Web Services 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 Amazon Web Services Support App in Slack API calls using Amazon CloudTrail

The Amazon Web Services Support App in Slack is integrated with Amazon CloudTrail. CloudTrail provides a record of actions taken by a user, role, or an Amazon Web Service in the Amazon Web Services Support App. To create this record, CloudTrail captures all public API calls for Amazon Web Services Support App as events. These captured calls include calls from the Amazon Web Services Support App console, and code calls to the Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon CloudTrail User Guide.

Amazon Web Services Support App information in CloudTrail

When you create your Amazon Web Services account, this activates CloudTrail on the account. When public API activity occurs in the Amazon Web Services Support App, that activity is recorded

in a CloudTrail event, along with other Amazon service events in **Event history**. You can view, search, and download recent events in your Amazon Web Services account. For more information, see Viewing events with CloudTrail Event history.

For an ongoing record of events in your Amazon Web Services account, including events for Amazon Web Services Support App, create a *trail*. By default, when you create a trail in the console, the trail applies to all Amazon Web Services Regions. The trail logs events from all Regions in the Amazon partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other Amazon Web 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 Amazon Web Services Support App actions. These actions are also documented in the Amazon Web Services 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 Amazon 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 Amazon Web Service.

For more information, see the CloudTrail userIdentity element.

Understanding Amazon Web Services 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 Amazon Web Services Support Plans

Monitoring is an important part of maintaining the reliability, availability, and performance of Support Plans and your other Amazon solutions. Amazon provides the following monitoring tools to watch Support Plans, report when something is wrong, and take automatic actions when appropriate:

- *Amazon CloudTrail* captures API calls and related events made by or on behalf of your Amazon account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called Amazon, the source IP address from which the calls were made, and when the calls occurred. For more information, see the Amazon CloudTrail User Guide.

Topics

- Logging Amazon Web Services Support Plans API calls with Amazon CloudTrail

Logging Amazon Web Services Support Plans API calls with Amazon CloudTrail

Amazon Web Services Support Plans is integrated with Amazon CloudTrail, a service that provides a record of actions taken by a user, role, or an Amazon Web Service. CloudTrail captures API calls for Amazon Web Services Support Plans as events. The calls captured include calls from the Amazon Web Services Support Plans console and code calls to the Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services 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 Amazon CloudTrail User Guide.

Amazon Web Services Support Plans information in CloudTrail

CloudTrail is enabled on your Amazon Web Services account when you create the account. When supported event activity occurs in Amazon Web Services Support Plans, that activity is recorded in a CloudTrail event along with other Amazon Web 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 Amazon Web Services 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 Amazon Web Services Regions. The trail logs events from all Regions in the Amazon partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other Amazon Web 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 Amazon Web Services 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 Amazon 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 Amazon Web Service.

For more information, see the CloudTrail userIdentity element.

You can also aggregate Amazon Web Services Support Plans log files from multiple Amazon Web Services Regions and multiple accounts into a single Amazon S3 bucket.

Understanding Amazon Web Services 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"
}
```

Logging changes to your Amazon Web Services 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 Amazon Web Services 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 Amazon Trusted Advisor

Monitoring is an important part of maintaining the reliability, availability, and performance of Trusted Advisor and your other Amazon solutions. Amazon 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 Amazon resources. EventBridge enables automated event-driven computing, as you can write rules that watch for certain events and trigger automated actions in other Amazon 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 Amazon 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.
- Amazon Trusted Advisor checks identify ways for you to reduce cost, increase performance, and improve security for your Amazon 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.
- *Amazon CloudTrail* captures API calls and related events made by or on behalf of your Amazon account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called Amazon, the source IP address from which the calls were made, and when the calls occurred. For more information, see the Amazon CloudTrail User Guide.

Topics

- Monitoring Amazon Trusted Advisor check results with Amazon EventBridge
- Creating Amazon CloudWatch alarms to monitor Amazon Trusted Advisor metrics
- Logging Amazon Trusted Advisor console actions with Amazon CloudTrail

Monitoring Amazon 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 Amazon 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 Amazon Web Services Support plan to create a rule for Trusted Advisor checks. For more information, see Changing Amazon Web Services 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://console.amazonaws.cn/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 **Amazon events or EventBridge partner events**.
8. Under **Event pattern**, keep the default value for **Amazon Web Services**.
9. For **Amazon Web 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 Amazon 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 Amazon Trusted Advisor metrics

When Amazon 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). For example, you might create an alarm to track status changes for checks in the **Service Limits** category. The alarm will then notify you when you reach or exceed a service quota for your Amazon account.

Follow this procedure to create a CloudWatch alarm for a specific Trusted Advisor metric.

Topics

- Prerequisites
- CloudWatch metrics for Trusted Advisor
- Trusted Advisor metrics and dimensions

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 Amazon Web Services 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

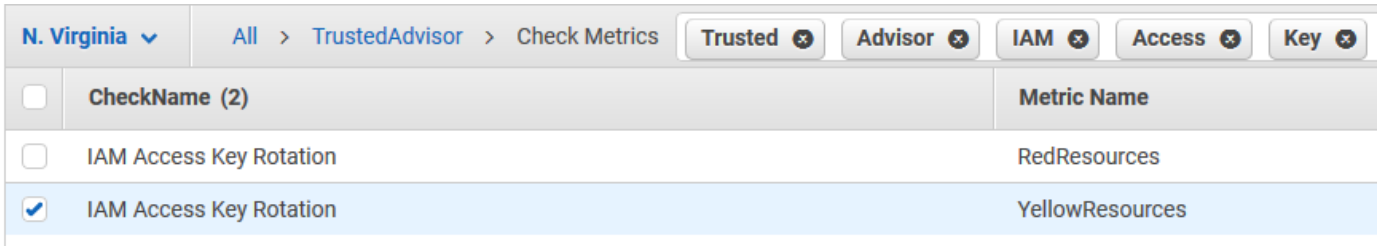
1. Open the CloudWatch console at https://console.amazonaws.cn/cloudwatch/.
2. Use the **Region selector** and choose the **US East (N. Virginia)** Amazon Region.
3. In the navigation pane, choose **Alarms**.
4. Choose **Create alarm**.
5. Choose **Select metric**.
6. For **Metrics**, enter one or more 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**.

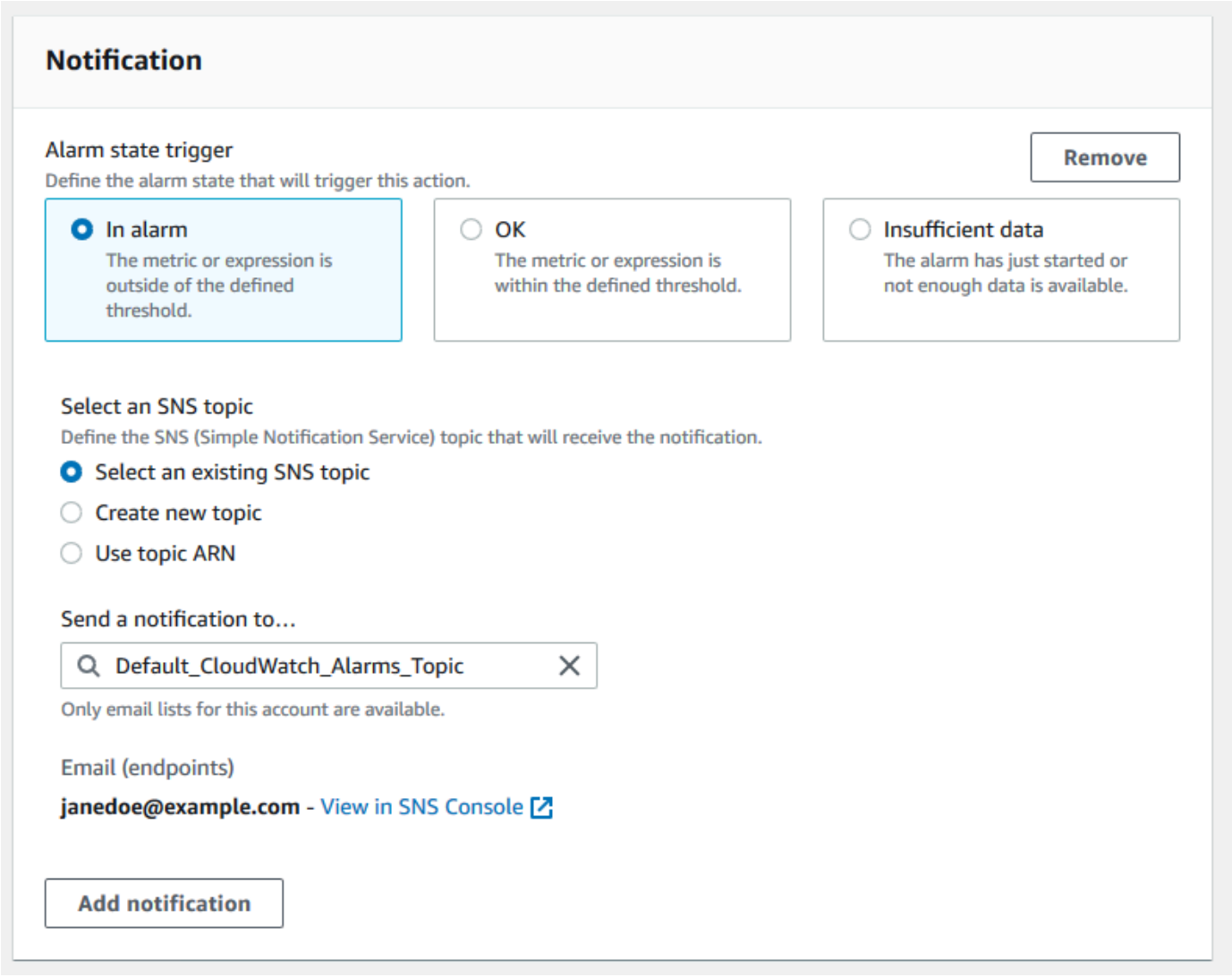


8. Choose **Select metric**.
9. On the **Specify metric and conditions** page, verify that the **Metric name** and **CheckName** that you chose appear on the page.
10. For **Period**, you can specify the time period that you want the alarm to start when the check status changes, such as 5 minutes.
11. 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 Amazon Web Services 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 Amazon access keys in
 my Amazon 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
- Amazon 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 Amazon Command Line Interface (Amazon CLI) to find the metrics available for Trusted Advisor.

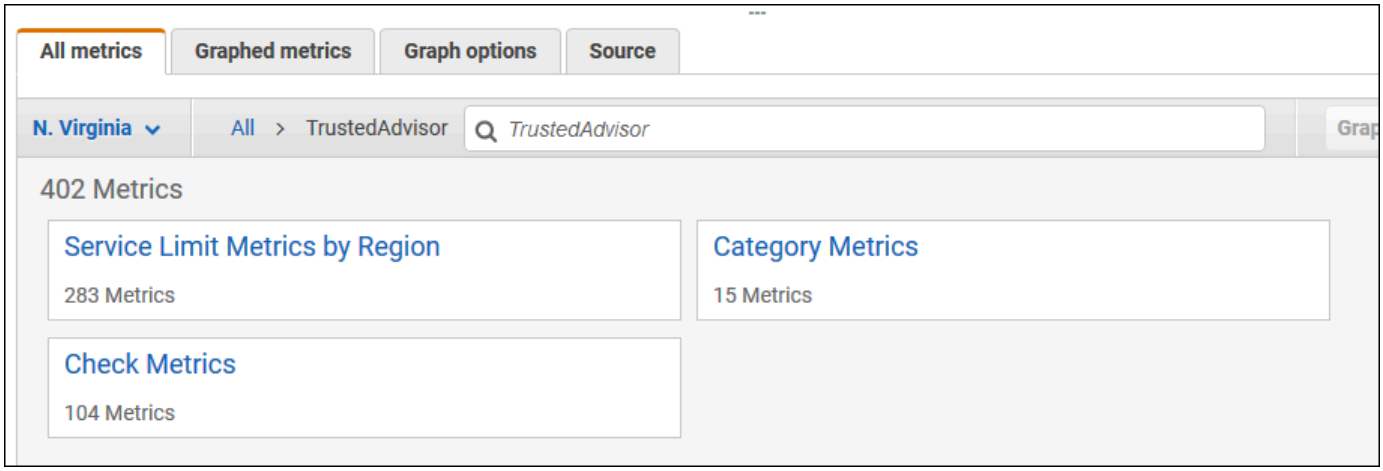
For a list of the namespaces, metrics, and dimensions for all services that publish metrics, see Amazon services that publish CloudWatch metrics in the *Amazon CloudWatch User Guide*.

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://console.amazonaws.cn/cloudwatch/.
2. Use the **Region selector** and choose the **US East (N. Virginia)** Amazon 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**.



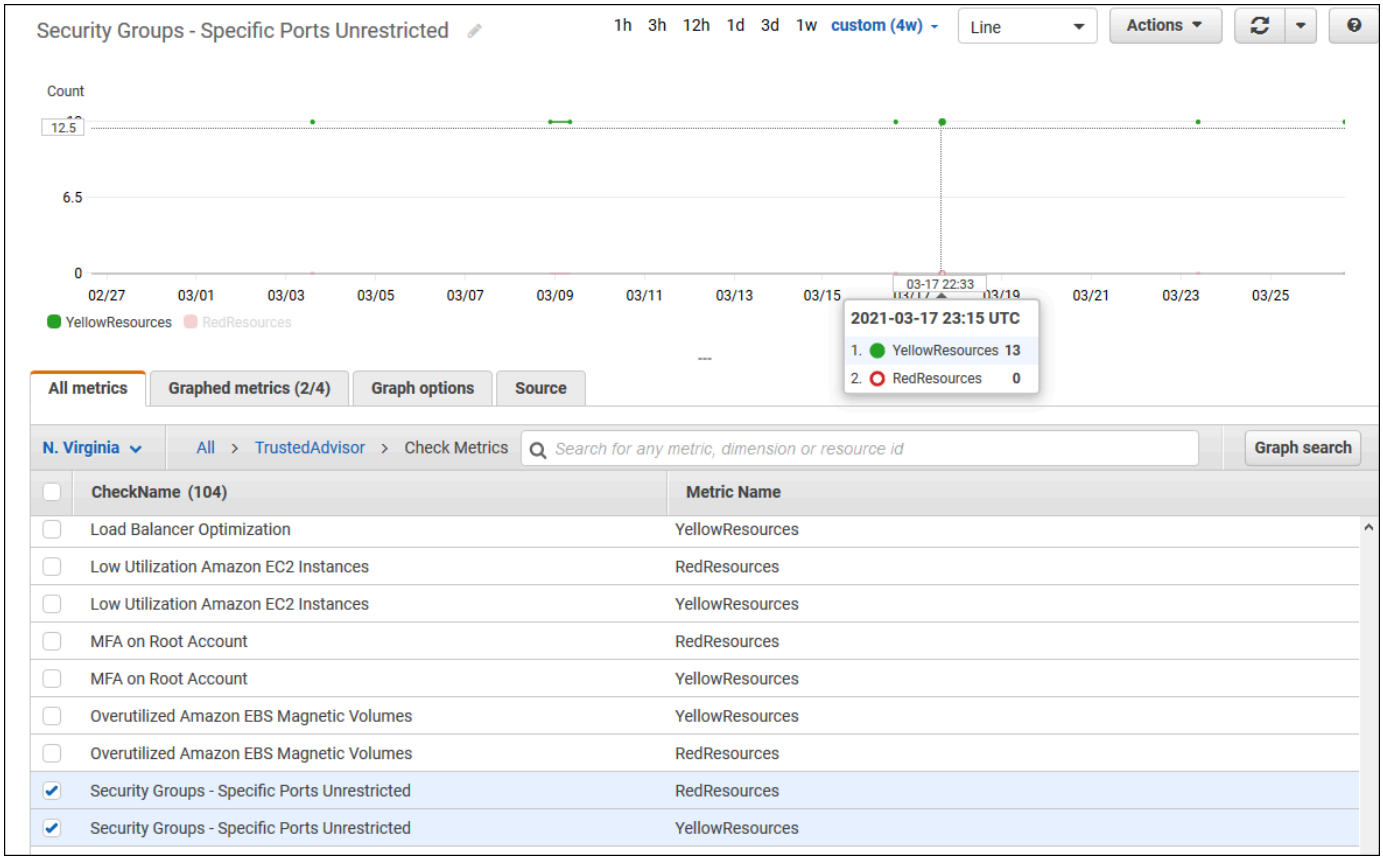
6. The **All metrics** tab shows metrics for that dimension in the namespace. You can do the following:

   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)

You can use the list-metrics Amazon CLI command to view available metrics for Trusted Advisor.

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 Amazon 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 category-level metrics

You can use the following metrics for Trusted Advisor categories.

Metric	Description
`GreenChecks`	The number of Trusted Advisor checks that are in a green state (no issues detected).
`RedChecks`	The number of Trusted Advisor checks that are in a red state (action recommended).
`YellowChecks`	The number of Trusted Advisor checks that are in a yellow state (investigation recommended).

Trusted Advisor service quota-level metrics

You can use the following metrics for Amazon Web 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 Amazon Trusted Advisor check reference.

Dimensions for category-level metrics

You can use the following dimension for Trusted Advisor check categories.

Dimension	Description
`Category`	The name of a Trusted Advisor check category. You can find all check categories in the Trusted Advisor console or the View check categories page.

Dimensions for service quota metrics

You can use the following dimensions for Trusted Advisor service quota metrics.

Dimension	Description
`Region`	The Amazon Web Services Region for a service quota.
`ServiceName`	The name of the Amazon Web Service.
`ServiceLimit`	The name of the service quota. For more information about service quotas, see Amazon Web Service quotas in the *Amazon Web Services General Reference*.

Logging Amazon Trusted Advisor console actions with Amazon CloudTrail

Trusted Advisor is integrated with Amazon CloudTrail, a service that provides a record of actions taken by a user, role, or an Amazon 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 theAmazon CloudTrail User Guide.

Trusted Advisor information in CloudTrail

CloudTrail is enabled on your Amazon 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 Amazon service events in **Event history**. You can view, search, and download recent events in your Amazon account. For more information, seeViewing Events with CloudTrail Event History.

For an ongoing record of events in your Amazon 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 Amazon Regions. The trail logs events from all Regions in the Amazon partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other Amazon 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:

- `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 Amazon Web Services Support API Reference. For more information, seeLogging Amazon Web Services Support API calls with Amazon 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 Amazon 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 Amazon 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 Amazon 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 Windows, Amazon EC2 offers EC2Rescue, which customers can use to examine their Windows instances to help identify common problems, collect log files, and help Amazon Web Services Support to 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 Amazon Web Service documentation contains troubleshooting topics that can get you started before contacting Amazon Web Services 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 Amazon Documentation landing page.

Service	Link
Amazon Web Services	Troubleshooting Amazon 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

Service	Link
Amazon Certificate Manager (ACM)	Troubleshooting
Amazon CloudFormation	Troubleshooting Amazon CloudFormation
Amazon CloudFront	Troubleshooting | Troubleshooting RTMP distributions
Amazon CloudHSM	Troubleshooting
Amazon CloudSearch	Troubleshooting Amazon CloudSearch
Amazon CodeDeploy	Troubleshooting Amazon CodeDeploy
Amazon CloudWatch	Troubleshooting
Amazon Database Migration Service	Troubleshooting migration tasks in Amazon Database Migration Service
Amazon Data Pipeline	Troubleshooting
Amazon Direct Connect	Troubleshooting Amazon Direct Connect
Amazon Directory Service	Troubleshooting Amazon Directory Service administration issues
Amazon DynamoDB	Troubleshooting | Troubleshooting SSL/TLS connection establishment issues
Amazon Elastic Beanstalk	Troubleshooting
Amazon Elastic Compute Cloud (Amazon EC2)	Troubleshooting instances | Troubleshooting Windows instances | Troubleshooting VM Import/Export | Troubleshooting API request errors | Troubleshooting the Amazon management pack | Troubleshooting Amazon Systems Manager for Microsoft SCVMM | Amazon diagnostics for Microsoft Windows server

Service	Link
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
Amazon ElastiCache for Memcached	Troubleshooting applications
Amazon ElastiCache for Redis	Troubleshooting applications
Amazon EMR	Troubleshoot a cluster
Amazon Flow Framework	Troubleshooting and debugging tips
Amazon Glue	Troubleshooting Amazon Glue
Amazon Glue DataBrew	Troubleshooting identity and access in Amazon Glue DataBrew
Amazon GovCloud (US)	Troubleshooting
Amazon 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

Service	Link
Amazon Lambda	Troubleshooting and monitoring Amazon Lambda functions with CloudWatch
Amazon OpenSearch Service	Troubleshooting Amazon OpenSearch Service
Amazon OpsWorks	Debugging and troubleshooting guide
Amazon Personalize	Troubleshooting
Amazon QLDB	Troubleshooting Amazon QLDB
Amazon QuickSight	Troubleshooting Amazon QuickSight | Troubleshooting skipped row errors
Amazon Resource Access Manager (Amazon RAM)	Troubleshooting issues with Amazon 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	Troubleshoot errors | Troubleshooting Amazon SageMaker Studio
Amazon Silk	Troubleshooting
Amazon Simple Email Service (Amazon SES)	Troubleshooting Amazon SES
Amazon Simple Storage Service (Amazon S3)	Troubleshooting

Service	Link
Amazon Simple Workflow Service (Amazon SWF)	Amazon flow framework for Java: Troubleshooting and debugging tips | Amazon flow framework for Ruby: Troubleshooting and debugging workflows
Amazon Storage Gateway	Troubleshooting your gateway
Amazon Systems Manager	Troubleshooting SSM Agent
Amazon Virtual Private Cloud (Amazon VPC)	Troubleshooting
Amazon Virtual Private Network (Amazon VPN)	Troubleshooting your customer gateway device
Amazon WAF	Testing and tuning your Amazon WAF protections
Amazon WorkMail	Troubleshooting the Amazon WorkMail web application
Amazon WorkSpaces	Troubleshooting Amazon WorkSpaces issues | Troubleshooting Amazon WorkSpaces client issues
Amazon WorkSpaces Application Manager (Amazon WAM)	Troubleshooting Amazon WAM application issues

Document history

The following table describes the important changes to the documentation since the last release of the Amazon Web Services Support service.

- **Amazon Web Services Support API version:** 2013-04-15
- **Amazon Web Services Support App API version:** 2021-08-20

The following table describes important updates to the Amazon Web Services Support and Amazon Trusted Advisor documentation, beginning May 10, 2021. You can subscribe to the RSS feed to receive notifications about the updates.

Change	Description	Date
Updated documentation for Amazon Web Services Support plan	Updates to the Features of Amazon Web Services Support Plans. For more information, see Amazon Web Services Support plans.	March 11, 2024
Updated documentation for Trusted Advisor	Added 1 fault tolerance check. For more information, see Change log for Amazon Trusted Advisor checks.	February 29, 2024
Updated documentation for Trusted Advisor	Added 1 fault tolerance check. For more information, see Change log for Amazon Trusted Advisor checks.	January 31, 2024
Updated documentation for AWSTrustedAdvisorServiceRolePolicy	Added new IAM actions `cloudtrail:GetTrail` , `cloudtrail:ListTrails` , `cloudtrail:GetEventSelectors` , `outposts:GetOutpost` , `outposts:ListAssets`	January 18, 2024

	and `outposts:ListOutposts` to onboard new checks. For more information, see Amazon managed policy: AWSTrustedAdvisorServiceRolePolicy.	
Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see Amazon managed policy: AWSSupportServiceRolePolicy.	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 Amazon Trusted Advisor checks.	January 8, 2024
Updated documentation for Trusted Advisor	Updated 1 security check to reflect change in deprecation period. For more information, see Change log for Amazon 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 Amazon Trusted Advisor checks.	December 20, 2023

Updated documentation for Trusted Advisor	Added 1 security check. For more information, see Change log for Amazon 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 optimizat ion check. For more informati on, see Change log for Amazon Trusted Advisor checks.	December 7, 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 Amazon managed policy: AWSSupportServiceRolePolicy .	December 6, 2023

Updated Amazon managed policies for Trusted Advisor	Updated the `AWSTrustedAdvisorPriorityFullAccess` and `AWSTrustedAdvisorPriorityReadOnlyAccess` Amazon managed policies to include statement IDs. For more information, see Amazon managed policies for Amazon Trusted Advisor.	December 6, 2023
Updated documentation for Trusted Advisor	Added 3 new fault tolerance checks. For more information, see Change log for Amazon 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 Amazon 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 Amazon 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 Amazon managed policy: AWSSupportServiceRolePolicy.	October 27, 2023
Updated documentation for Trusted Advisor	Added 64 new checks integrated from Amazon Config. For more information, see Change log for Amazon 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 Amazon 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 Amazon 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 Amazon managed policy: AWSSupportServiceRolePolicy .	August 28, 2023
Updated documentation for Trusted Advisor	Added Added 1 new service limits checks for Lambda. For more information, see the Change log for Amazon 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 Amazon 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 Amazon 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 Amazon 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 Amazon Trusted Advisor checks.	June 1, 2023
Updated documentation for Trusted Advisor	Added two new fault tolerance checks for NAT Gateway. For more informati on, see the Change log for Amazon Trusted Advisor checks.	May 16, 2023
Updated documentation for Amazon Web Services Support Plans	Added a new permission and CloudTrail documentation for the creation of support plan schedules. For more informati on, see Manage access to Amazon Web Services Support Plans, Amazon managed policies for Amazon Web Services Support Plans and Logging Amazon Web Services Support Plans API calls with Amazon CloudTrail.	May 8, 2023

Updated documentation for `AWSSupportServiceRolePolicy`	Added new permissions to provide billing, administrative, and support services for the service-linked role. For more information, see Amazon managed policy: AWSSupportServiceRolePolicy .	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 Amazon Resilience Hub and Incident Manager. For more information, see the Change log for Amazon Trusted Advisor checks.	April 27, 2023

Added documentation for Trusted Advisor Engage	You can use Amazon Trusted Advisor Engage to get the most out of your Amazon Web Services Support Plans by making it easy for you to see, request and track all your proactive engagements, and communicate with your Amazon Web Services account team about ongoing engagements. For more information, see Get started with Amazon 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 Amazon 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 Amazon 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 Amazon Web Services Support	Added .NET, Java, and Kotlin code examples that show how to use Amazon Web Services Support with an Amazon software development kit (SDK). For more informati on, see Code examples for Amazon Web Services Support using Amazon 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 Amazon managed policy: AWSSupportServiceRolePolicy .	January 10, 2023

Updated documentation for Amazon Web Services 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 Amazon Web Services Support App	You can also use Terraform to create your resources for the Amazon Web Services Support App. For more information, see Create Amazon Web Services 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 Amazon CloudHSM. For more information, see the Change log for Amazon Trusted Advisor checks.	December 15, 2022
Updated documentation for the Amazon Web Services 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 Amazon Web Services Support	Added documentation about new endpoints for the Amazon Web Services Support API. For more information, see About the Amazon Web Services Support API.	December 14, 2022
Added documentation for Amazon CloudFormation templates to use for the Amazon Web Services Support App in Slack	You can use CloudFormation templates to create Slack configuration workspaces and channels for Amazon Web Services accounts in Amazon Organizations. For more information, see Creating Amazon Web Services Support App resources with Amazon CloudFormation.	December 5, 2022
Updated documentation for Trusted Advisor	Added two new fault tolerance checks for Amazon Resilience Hub. For more information, see the Change log for Amazon Trusted Advisor checks.	November 17, 2022
Added documentation for your Amazon Security Hub findings in Trusted Advisor	Your findings from Security Hub controls are removed from Trusted Advisor faster. For more information, see the Change log for Amazon Trusted Advisor checks.	November 17, 2022

Updated documentation for Amazon Trusted Advisor	Added documentation for Trusted Advisor Recommendations. For more information, see the Change log for Amazon Trusted Advisor checks.	November 16, 2022
Updated documentation for the Amazon Web Services 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 Amazon Web Services Support Plans	Added troubleshooting information to allow Support Plans access in an organization. For more information, see Troubleshooting.	November 9, 2022
Updated documentation for the Amazon Web Services Support App in Slack	Added documentation for `supportapp` permissions. For more information, see Permissions required for the Amazon Web Services Support App to connect to Slack.	November 1, 2022

Updated documentation for the Amazon Web Services Support App in Slack	You can use the `RegisterSlackWorkspaceForOrganization` API operation to register a Slack workspace for your Amazon Web Services account. To call this API, your account must be part of an organization in Amazon Organizations. For more information, see the Amazon Web Services 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 Amazon managed policy: AWSSupportServiceRolePolicy .	October 4, 2022

Updated documentation for Support Plans	You can now use Amazon Identity and Access Management (IAM) to manage permissions to change the support plan for your Amazon Web Services account. For more informati on, see the following topics: • Managing access for Amazon Web Services Support Plans • Amazon managed policies for Amazon Web Services Support Plans • Changing Amazon Web Services Support Plans • Logging Amazon Web Services Support Plans API calls with Amazon CloudTrail	September 29, 2022
Updated documentation for the Amazon Web Services Support App in Slack	Added documentation on how to configure a public or private channel to use with the Amazon Web Services Support App. For more information, see Configuring a Slack channel.	September 22, 2022
Updated documentation for Amazon Web Services Support	Added a new section about security for your support cases. For more information, see Security for your Amazon Web Services 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 Amazon Trusted Advisor checks.	September 1, 2022
Updated documentation for the Amazon Web Services Support App in Slack	See the following topics: You can use the Amazon Web Services 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 Amazon Web Services Support App in Slack documentation. You can attach Amazon Web Services managed policies to your IAM roles to use the Amazon Web Services Support App. For more information, see Amazon Web Services managed policies for Amazon Web Services Support App in Slack. New API reference for the Amazon Web Services Support App. See the Amazon Web Services 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 Amazon 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 • Amazon Web Services 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 Amazon 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 Amazon Trusted Advisor check reference.	July 7, 2022
Updated documentation for Amazon Web Services 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 Amazon managed policy: AWSSupportServiceRolePolicy .	June 23, 2022

Updated documentation for Trusted Advisor	Trusted Advisor supports additional Amazon Foundational Security Best Practices security standard controls that are sourced from Amazon Security Hub. For more information, see the Change log for Amazon 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 Amazon Web Services 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 Amazon Lambda. For more information, see Opt in Amazon 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 Amazon 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 Amazon Trusted Advisor checks.	April 25, 2022
Updated documentation for Trusted Advisor	The Amazon Direct Connect checks in the fault tolerance category are updated. For more information, see Change log for Amazon 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 Amazon 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 Amazon Trusted Advisor check results with EventBridge.	February 21, 2022
New documentation for using Amazon EventBridge to monitor Amazon Web Services Support cases	You can create an EventBridge rule to monitor and receive notifications about your support cases. For more information, see Monitoring Amazon Web Services 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 Amazon managed policy: AWSSupportServiceRolePolicy.	February 17, 2022
Added documentation for integrating with Amazon Security Hub	In the Trusted Advisor console, you can now view the findings for your Security Hub controls that are part of the Amazon Foundational Security Best Practices security standard. For more information, see Viewing Amazon Security Hub controls in the Amazon 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 Amazon Web Services 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 Amazon 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 Amazon 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 Amazon managed policies for Amazon Web Services Support and Amazon Trusted Advisor.	August 10, 2021
Updated documentation for Trusted Advisor	Updated documentation for the Trusted Advisor console. For more information, see Get started with Amazon Trusted Advisor.	July 16, 2021

Updated documentation for creating Amazon Web Services 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 Amazon 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 Amazon Trusted Advisor metrics topic • Added the Service quotas for the Amazon Web Services 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 Amazon managed policies	Added information about the `AWSSupportServiceRolePolicy` Amazon managed policy. For more information, see Using service-linked roles for Amazon Web Services Support.	March 16, 2021
Added checks for Amazon Lambda	Added four Amazon Trusted Advisor checks for Lambda in the Change log for Amazon Trusted Advisor.	March 8, 2021
Updated service limit checks for Amazon Elastic Block Store	Updated five Amazon Trusted Advisor checks for Amazon EBS in the Change log for Amazon Trusted Advisor.	March 5, 2021
Updated documentation for CloudTrail logging	CloudTrail supports logging for console actions when you change your Amazon Web Services Support plan. For more information, see Logging changes to your Amazon Web Services Support plan.	February 9, 2021
Updated documentation for Trusted Advisor	Updated the Get started with Trusted Advisor Recommendations topic.	January 29, 2021

Change	Description	Date
Updated documentation for Trusted Advisor reports	Added a Troubleshooting section for using Trusted Advisor reports with other Amazon services.	December 4, 2020
Added Amazon Trusted Advisor support for Amazon CloudTrail logging	CloudTrail supports logging for a subset of Trusted Advisor console actions. For more information, see Logging Amazon Trusted Advisor console actions with Amazon CloudTrail.	November 23, 2020
Added a change log topic	View changes to Amazon Trusted Advisor checks and categories in the Change log for Amazon 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
Updated the logging with Amazon CloudTrail topic	Added an example log entry for a Trusted Advisor API operation. See Amazon Trusted Advisor information in CloudTrail logging.	October 22, 2020
Added Amazon Web Services Support quotas	Added information about the current quotas and restrictions for Amazon Web Services Support. See the Amazon Web Services Support endpoints and quotas in the *Amazon Web Services General Reference*.	August 4, 2020
Organizational view for Amazon Trusted Advisor	You can now create reports for Trusted Advisor checks for accounts that are part of Amazon Organizations. See Organizational view for Amazon Trusted Advisor.	July 17, 2020

Change	Description	Date
Security and Amazon Web Services Support	Updated information about security considerations when using Amazon Web Services Support and Trusted Advisor. See Security in Amazon Web Services Support	May 5, 2020
Security and Amazon Web Services Support	Added information about security considerations when using Amazon Web Services 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 Amazon 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
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 Amazon Web Services Support Calls with Amazon CloudTrail	Added new section.	April 21, 2016
Getting Started: Troubleshooting	Added more troubleshooting links.	May 19, 2015

Change	Description	Date
Getting Started: Troubleshooting	Added more troubleshooting links.	November 18, 2014
Getting Started: Case Management	Updated to reflect Service Catalog in the Amazon Web Services Management Console.	October 30, 2014
Programming the Life of an Amazon Web Services 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 Amazon Web Services 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 Amazon Web Services Support service released.	April 30, 2013

Amazon Glossary

For the latest Amazon terminology, see the Amazon glossary in the *Amazon Web Services Glossary Reference*.