# Table of Contents

What is AWS RAM? .......................................................................................................................... 1
Benefits ........................................................................................................................................ 1
How resource sharing works ........................................................................................................ 1
Sharing your resources ............................................................................................................... 1
Using shared resources ............................................................................................................ 1
Service limits ............................................................................................................................ 2
Accessing AWS RAM .................................................................................................................. 2
Pricing ....................................................................................................................................... 2
Shareable resources .................................................................................................................... 3
  AWS App Mesh .......................................................................................................................... 3
  Amazon Aurora ....................................................................................................................... 3
  AWS Certificate Manager Private Certificate Authority .......................................................... 4
  AWS CodeBuild ..................................................................................................................... 4
  Amazon EC2 ............................................................................................................................ 4
  EC2 Image Builder ................................................................................................................ 5
  AWS Glue ................................................................................................................................. 5
  AWS License Manager ........................................................................................................... 6
  AWS Network Firewall .......................................................................................................... 6
  AWS Outposts ........................................................................................................................ 7
  AWS Resource Groups .......................................................................................................... 7
  Amazon Route 53 .................................................................................................................. 8
  Amazon VPC ........................................................................................................................ 8
Getting started ............................................................................................................................ 10
Sharing your resources ............................................................................................................. 10
  Enable sharing with AWS Organizations .................................................................................. 10
  Create a resource share .......................................................................................................... 11
Using shared resources ............................................................................................................. 12
  Respond to the resource share invitation .............................................................................. 12
  Use the resources that are shared with you ............................................................................ 13
Working with shared resources ................................................................................................ 14
Owned by you .............................................................................................................................. 14
  Creating a resource share ........................................................................................................ 14
  Updating a resource share ....................................................................................................... 15
  Viewing a resource share ........................................................................................................ 15
  Viewing your shared resources .............................................................................................. 16
  Viewing principals .................................................................................................................. 16
  Deleting a resource share ........................................................................................................ 17
  Supported actions on shared resources ................................................................................ 17
Shared with you ........................................................................................................................ 17
  Accepting and rejecting invitations ....................................................................................... 17
  Viewing resource shares ........................................................................................................ 18
  Viewing shared resources ....................................................................................................... 19
  Viewing principals sharing with you ..................................................................................... 19
  Leaving a resource share ........................................................................................................ 19
AZ IDs ..................................................................................................................................... 20
Security ..................................................................................................................................... 21
Data protection .......................................................................................................................... 21
Identity and access management .............................................................................................. 22
  How AWS RAM works with IAM ............................................................................................ 22
  Example IAM policies ............................................................................................................ 24
  Disable sharing with Organizations ....................................................................................... 26
AWS RAM permissions ............................................................................................................. 26
  How AWS RAM permissions work ........................................................................................ 26
  AWS-managed permissions .................................................................................................. 27
What is AWS RAM?

AWS Resource Access Manager (AWS RAM) lets you share your resources with any AWS account or through AWS Organizations. If you have multiple AWS accounts, you can create resources centrally and use AWS RAM to share those resources with other accounts.

Contents
- Benefits (p. 1)
- How resource sharing works (p. 1)
- Service limits (p. 2)
- Accessing AWS RAM (p. 2)
- Pricing (p. 2)
- Shareable AWS resources (p. 3)

Benefits

AWS RAM offers the following benefits:

- **Reduces operational overhead**—Create resources centrally and use AWS RAM to share those resources with other accounts. This eliminates the need to provision duplicate resources in every account, which reduces operational overhead.
- **Provides security and consistency**—Govern consumption of shared resources using existing policies and permissions, to achieve security and control. AWS RAM offers a consistent experience for sharing different types of AWS resources.
- **Provides visibility and auditability**—View usage details for shared resources through integration with Amazon CloudWatch and AWS CloudTrail. AWS RAM provides comprehensive visibility into shared resources and accounts.

How resource sharing works

When you share a resource with another account, then that account is granted access to the resource. Any policies and permissions that apply to the account with which you have shared the resource apply to the shared resource.

Sharing your resources

You can share resources that you own by creating a resource share. When you create a resource share, you specify a name, the resources to share, and the principals with whom to share. Principals can be AWS accounts, organizational units, or an entire organization from AWS Organizations. Your account retains full ownership of the resources that you share.

Using shared resources

When the owner of a resource shares it with your account, you can access the shared resource just as you would if it was owned by your account. You can access the resource using the respective service's console, AWS CLI, and API. The actions that users are allowed to perform vary depending on the resource type.
IAM policies and service control policies configured in your account apply, which enables you to leverage your existing investments in security and governance controls.

Service limits

Your AWS account has the following limits related to AWS RAM. You can request an increase for some of these limits. To request a limit increase, contact AWS Support.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Default limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of resource shares per account</td>
<td>5,000</td>
</tr>
<tr>
<td>Maximum number of shared principals per account</td>
<td>5,000</td>
</tr>
<tr>
<td>Maximum number of shared resources per account</td>
<td>5,000</td>
</tr>
<tr>
<td>Maximum number of pending invitations per account</td>
<td>20</td>
</tr>
</tbody>
</table>

Accessing AWS RAM

You can work with AWS RAM in any of the following ways:

AWS RAM Console

AWS RAM provides a web-based user interface, the AWS RAM console. If you’ve signed up for an AWS account, you can access the AWS RAM console by signing into the AWS Management Console and selecting AWS RAM from the console home page.

AWS Command Line Interface (AWS CLI)

The AWS CLI provides direct access to the AWS RAM public API operations. It is supported on Windows, macOS, and Linux. For more information about getting started, see the AWS Command Line Interface User Guide. For more information about the commands for AWS RAM, see the AWS CLI Command Reference.

AWS Tools for Windows PowerShell

AWS provides commands for a broad set of AWS products for those who script in the PowerShell environment. For more information about getting started, see the AWS Tools for Windows PowerShell User Guide. For more information about the cmdlets for AWS RAM, see the AWS Tools for Windows PowerShell Cmdlet Reference.

Query API

The AWS RAM HTTPS Query API gives you programmatic access to AWS RAM and AWS. The AWS RAM API lets you issue HTTPS requests directly to the service. When you use the AWS RAM API, you must include code to digitally sign requests using your credentials. For more information, see the AWS RAM API Reference.

Pricing

There are no additional charges for creating resource shares and sharing your resources across accounts. Resource usage charges vary depending on the resource type. For more information about how shareable resources are billed, refer to the respective service's documentation.
Shareable AWS resources

AWS RAM enables you to share resources that are provisioned and managed in other AWS services. AWS RAM does not let you manage resources, but it does provide the features that let you make resources available across AWS accounts.

The following sections list the services that integrate with AWS RAM, and the resources that support sharing.

Services
- AWS App Mesh (p. 3)
- Amazon Aurora (p. 3)
- AWS Certificate Manager Private Certificate Authority (p. 4)
- AWS CodeBuild (p. 4)
- Amazon EC2 (p. 4)
- EC2 Image Builder (p. 5)
- AWS Glue (p. 5)
- AWS License Manager (p. 6)
- AWS Network Firewall (p. 6)
- AWS Outposts (p. 7)
- AWS Resource Groups (p. 7)
- Amazon Route 53 (p. 8)
- Amazon VPC (p. 8)

AWS App Mesh

You can share the following AWS App Mesh resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mesh</td>
<td>Create and manage a mesh centrally, and share it with other AWS accounts. A shared mesh allows resources created by different AWS accounts to communicate with each other in the same mesh. For more information, see Working with shared meshes in the AWS App Mesh User Guide.</td>
</tr>
</tbody>
</table>

Amazon Aurora

You can share the following Amazon Aurora resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB clusters</td>
<td>Create and manage a DB cluster centrally, and share it with other AWS accounts. This lets multiple AWS accounts clone a shared, centrally-managed DB cluster. For more information, see Cross-Account Aurora DB Cluster Cloning in the Amazon Aurora User Guide.</td>
</tr>
</tbody>
</table>
### AWS Certificate Manager Private Certificate Authority

You can share the following ACM Private CA resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Private certificate authority (CA)</td>
<td>Create and manage private certificate authorities (CAs) for your organization's internal PKI, and share them with other AWS accounts. This lets AWS Certificate Manager users in other accounts issue X.509 certificates signed by your shared CA. For more information, see <a href="#">Enabling access to a private CA in the AWS Certificate Manager Private Certificate Authority User Guide</a>.</td>
</tr>
</tbody>
</table>

### AWS CodeBuild

You can share the following AWS CodeBuild resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Projects</td>
<td>Create a project and use it to run builds. Share the project with other AWS accounts or users. This lets multiple AWS accounts and users view information about a project and analyze its builds. For more information, see <a href="#">Working with shared projects in the AWS CodeBuild User Guide</a>.</td>
</tr>
<tr>
<td>Report groups</td>
<td>Create a report group and use it to create reports when you build a project. Share the report group with other AWS accounts or users. This lets multiple AWS accounts and users view the report group and its reports, and the test case results for each report. A report can be viewed for 30 days after it is created, and then it expires and is no longer available to view. For more information, see <a href="#">Working with shared report groups in the AWS CodeBuild User Guide</a>.</td>
</tr>
</tbody>
</table>

### Amazon EC2

You can share the following Amazon EC2 resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity Reservations</td>
<td>Create and manage Capacity Reservations centrally, and share the reserved capacity with other AWS accounts. This lets multiple AWS accounts launch their Amazon EC2 instances into centrally-managed reserved capacity. For more information, see <a href="#">Working with shared Capacity</a>.</td>
</tr>
</tbody>
</table>
### Dedicated Hosts

Allocate and manage Amazon EC2 Dedicated Hosts centrally, and share the host's instance capacity with other AWS accounts. This lets multiple AWS accounts launch their Amazon EC2 instances onto centrally-managed Dedicated Hosts. For more information, see Working with shared Dedicated Hosts in the Amazon EC2 User Guide for Linux Instances.

### EC2 Image Builder

You can share the following EC2 Image Builder resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Components</td>
<td>Create and manage components centrally, and share them with other AWS accounts or your organization. Manage who can use predefined build and test components in their image recipes. For more information, see Resource Sharing in EC2 Image Builder in the EC2 Image Builder User Guide.</td>
</tr>
<tr>
<td>Images</td>
<td>Create and manage your golden images centrally, and share them with other AWS accounts and your organization. Manage who can use images created with EC2 Image Builder across your organization.. For more information, see Resource Sharing in EC2 Image Builder in the EC2 Image Builder User Guide.</td>
</tr>
<tr>
<td>Image recipes</td>
<td>Create and manage your image recipes centrally, and share them with other AWS accounts and your organization. This allows you to manage who can use predefined documents to automate repeatable image pipelines for a desired configuration. For more information, see Resource Sharing in EC2 Image Builder in the EC2 Image Builder User Guide.</td>
</tr>
</tbody>
</table>

### AWS Glue

You can share the following AWS Glue resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data catalogs</td>
<td>Manage a central data catalog and share metadata about databases and tables with AWS</td>
</tr>
</tbody>
</table>
AWS License Manager

You can share the following AWS License Manager resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>License configurations</td>
<td>Create and manage license configurations centrally, and share them with other AWS accounts. This lets you enforce centrally-managed licensing rules that are based on the terms of your enterprise agreements across multiple AWS accounts. For more information, see Using license configurations in the License Manager User Guide.</td>
</tr>
</tbody>
</table>

AWS Network Firewall

You can share the following AWS Network Firewall resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firewall policies</td>
<td>Create and manage firewall policies centrally, and share them within your AWS organization. This enables multiple accounts in an AWS organization.</td>
</tr>
</tbody>
</table>
### AWS Outposts

You can share the following AWS Outposts resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outposts</td>
<td>Create and manage Outposts centrally, and share them within your AWS organization. This enables multiple accounts to create subnets and EBS volumes on your shared, centrally-managed Outposts. For more information, see Working with shared AWS Outposts resources in the AWS Outposts User Guide.</td>
</tr>
<tr>
<td>Local gateway route tables</td>
<td>Create and manage local gateway route tables on Outpost centrally, and share them within your AWS organization. This enables multiple accounts to create VPC associations to a local gateway, and view configurations of local gateway route tables and virtual interfaces on your Outpost. For more information, see Working with shared AWS Outposts resources in the AWS Outposts User Guide.</td>
</tr>
<tr>
<td>Subnets</td>
<td>Create and manage subnets on Outpost centrally, and share them within your AWS organization. This enables multiple accounts to launch and run EC2 instances in shared subnets on your Outpost. For more information, see Working with shared AWS Outposts resources in the AWS Outposts User Guide.</td>
</tr>
</tbody>
</table>

### AWS Resource Groups

You can share the following AWS Resource Groups resources using AWS RAM.
<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource groups</td>
<td>Create and manage a host resource group centrally, and share it with other AWS accounts. This lets multiple AWS accounts share a group of Amazon EC2 Dedicated Hosts created using AWS License Manager. For more information, see Host resource groups in AWS License Manager in the AWS License Manager User Guide.</td>
</tr>
</tbody>
</table>

**Amazon Route 53**

You can share the following Amazon Route 53 resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forwarding rules</td>
<td>Create and manage forwarding rules centrally, and share them with other AWS accounts. This lets multiple AWS accounts forward DNS queries from their VPCs to the target IP addresses defined in shared, centrally-managed resolver rules. For more information, see Sharing forwarding rules with other AWS accounts and using shared rules in the Amazon Route 53 Developer Guide.</td>
</tr>
<tr>
<td>Query logs</td>
<td>Create and manage query logs centrally, and share them with other AWS accounts. This enables multiple AWS accounts to log DNS queries that originate in their VPCs to a centrally-managed query log. For more information, see Sharing Resolver query logging configurations with other AWS accounts in the Amazon Route 53 Developer Guide.</td>
</tr>
</tbody>
</table>

**Amazon VPC**

You can share the following Amazon VPC resources using AWS RAM.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer-owned IPv4 addresses</td>
<td>During the AWS Outposts installation process, AWS creates an address pool, known as a customer-owned IP address pool, based on information that you provide about your on-premises network. Customer-owned IP addresses provide local, or external connectivity to resources in your Outpost subnets through your on-premises network. You can assign these addresses to resources on your Outpost, such as EC2 instances, using Elastic IP addresses or using the subnet setting</td>
</tr>
<tr>
<td>Resource</td>
<td>Use case</td>
</tr>
<tr>
<td>------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Prefix lists</td>
<td>Create and manage prefix lists centrally, and share them with other AWS accounts. This lets multiple AWS accounts reference prefix lists in their resources, such as VPC security groups and subnet route tables. For more information, see <a href="https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-properties-iam-policy-statement-resourceprefixlist.html">Working with Shared Prefix Lists</a> in the Amazon VPC User Guide.</td>
</tr>
<tr>
<td>Subnets</td>
<td>Create and manage subnets centrally, and share them with other accounts or organizational units that are in the same organization from AWS Organizations. This lets multiple AWS accounts launch their application resources into centrally-managed VPCs. These resources include Amazon EC2 instances, Amazon Relational Database Service (RDS) databases, Amazon Redshift clusters, and AWS Lambda functions. For more information, see <a href="https://docs.aws.amazon.com/vpc/latest/userguide/vpc-sharing-overview.html">Working with VPC sharing</a> in the Amazon VPC User Guide.</td>
</tr>
<tr>
<td>Traffic mirror targets</td>
<td>Create and manage traffic mirror targets centrally, and share them with other AWS accounts. This lets multiple AWS accounts send mirrored network traffic from traffic mirror sources in their accounts to a shared, centrally-managed traffic mirror target. For more information, see <a href="https://docs.aws.amazon.com/traffic-mirroring/latest/guide/accounts_cross-account_traffic-mirroring_targets.html">Cross-Account Traffic Mirroring Targets</a> in the Traffic Mirroring Guide.</td>
</tr>
<tr>
<td>Transit gateways</td>
<td>Create and manage transit gateways centrally, and share them with other AWS accounts. This lets multiple AWS accounts route traffic between their VPCs and on-premises networks through a shared, centrally-managed transit gateway. For more information, see <a href="https://docs.aws.amazon.com/vpc/latest/userguide/amazon-vpc-transit-gateways.html">Sharing a transit gateway</a> in the Amazon VPC Transit Gateways.</td>
</tr>
<tr>
<td>Transit gateway multicast domains</td>
<td>Create and manage transit gateway multicast domains centrally, and share them with other AWS accounts. This lets multiple AWS accounts register and deregister group members or group sources in the multicast domain. For more information, see <a href="https://docs.aws.amazon.com/traffic-mirroring/latest/guide/accounts_overview.html">Working with shared multicast domains</a> in the Traffic Mirroring Guide.</td>
</tr>
</tbody>
</table>
Getting started with AWS RAM

With AWS RAM, you can share resources that you own with individual AWS accounts or through AWS Organizations, and you can use resources that were shared with you by other AWS accounts or through AWS Organizations.

Contents
- Sharing your AWS resources (p. 10)
- Using shared AWS resources (p. 12)

Sharing your AWS resources

To start sharing a resource that you own using AWS RAM, do the following:

- Enable sharing with AWS Organizations (p. 10)
- Create a resource share (p. 11)

Note
Some resources have special considerations and prerequisites for sharing. For more information, see Shareable AWS resources (p. 3).

Enable sharing with AWS Organizations

If you would like to share resources with your organization or organizational units, then you must use the AWS RAM console or CLI command to enable sharing with AWS Organizations. When you share resources within your organization, AWS RAM does not send invitations to principals. Principals in your organization get access to shared resources without exchanging invitations.

If you no longer need to share resources with your entire organization or organizational units, you can disable sharing. For more information, see Disabling resource sharing with AWS Organizations (p. 26).

Requirements
- Only the management account can enable sharing with AWS Organizations.
- The organization must be enabled for all features. For more information, see Enabling All Features in Your Organization in the AWS Organizations User Guide.

Important
- If you do not enable sharing with AWS Organizations, you cannot share resources with your organization or organizational units within your organization. However, you can still share resources with individual AWS accounts in your organization. In this case, the accounts are treated as external principals. They receive an invitation to join the resource share, and they must accept the invitation to get access to the shared resources.
- You must enable sharing with AWS Organizations using the AWS RAM console or the enable-sharing-with-aws-organization AWS CLI command. This ensures that the AWSServiceRoleForResourceAccessManager service-linked role is created. If you enable
trusted access with AWS Organizations using the AWS Organizations console or the `enable-aws-service-access` AWS CLI command, the `AWSServiceRoleForResourceAccessManager` service-linked role is not created, and you will not be able to share resources within your organization.

**To enable sharing with AWS Organizations using the console**

1. Open the **Settings** page of AWS RAM console at [https://console.aws.amazon.com/ram/home#Settings](https://console.aws.amazon.com/ram/home#Settings).
2. Choose **Enable sharing with AWS Organizations**.

**To enable sharing with AWS Organizations using the AWS CLI**

Use the `enable-sharing-with-aws-organization` command.

This command can be used in any region, and it enables sharing with AWS Organizations in all regions in which AWS RAM is supported.

## Create a resource share

To share resources that you own, create a resource share, add the resources to share, and specify the principals with whom they are to be shared.

**Considerations**

- You can share a resource only if you own it. You can't share a resource that is shared with you.
- If you are part of an organization in AWS Organizations and sharing within your organization is enabled, principals in your organization are automatically granted access to the shared resources. Otherwise, principals receive an invitation to join the resource share and are granted access to the shared resources after accepting the invitation.
- After you add an organization to a resource share, changes to the OU or organization affect the resource share. For example, if you add a new account to the organization, it has access to the shared resources.
- You can't add the following to a resource share as principals: IAM users, IAM roles, or OUs or organizations outside your organization in AWS Organizations.

**To create a resource share using the console**

2. If you are new to AWS RAM, choose **Create a resource share** from the home page. Otherwise, choose **Create resource share** from the **Resource shares** page.
3. Under **Description**, for **Name**, type a descriptive name for the resource share.
4. (Optional) Under **Resources**, select resources to add to the resource share as follows:
   a. For **Select resource type**, select the type of resource. This filters the list of shareable resources to resources of the selected type.
   b. Select the check boxes next to the resources. The selected resources are moved under **Selected resources**.

   If you are sharing zonal resources, using the Availability Zone ID (AZ ID) helps you determine the relative location of these resources across accounts. For more information, see [AZ IDs for your AWS resources](p. 20).
5. (Optional) Under **Principals**, do the following:
a. By default, you can share resources with any AWS account. To restrict resource sharing to your organization in AWS Organizations, clear **Allow external accounts**.

b. For each principal, specify its ID and choose **Add**:
   - To add an AWS account, type the 12-digit account ID. For example, 123456789012.
   - To add an OU, type the ID of the OU. For example, ou-abcd1234-mnop5678qrst9098uv76.
   - To add your entire organization, type the ID of the organization. For example, o-abcd1234efgh5678.

6. (Optional) Under **Tags**, type a tag key and tag value. To add another tag, choose **Add tag** and type a tag key and tag value pair. These tags are not applied to the resources included in the resource share.

7. Choose **Create resource share**.

   It can take a few minutes for the resource and principal associations to complete. Allow this process to complete before attempting to use the resource share.

8. You can add and remove resources and principals or apply custom tags to your resource share at any time. You can delete your resource share when you no longer want to share the resources. For more information, see Sharing AWS resources owned by you (p. 14).

**To create a resource share using the AWS CLI**

Use the `create-resource-share` command.

---

**Using shared AWS resources**

To start using shared resources, do the following:

- **Respond to the resource share invitation** (p. 12)
- **Use the resources that are shared with you** (p. 13)

**Respond to the resource share invitation**

If you receive an invitation to join a resource share, you must accept it to gain access to the shared resources. If you are part of an organization in AWS Organizations and sharing within your organization is enabled, principals in your organization are automatically granted access to the shared resources and do not receive these invitations.

**To respond to invitations**

2. In the navigation pane, choose **Shared with me, Resource shares**.
3. Review the list of resource shares to which you have been added.

   The **Status** column indicates your current participation status for the resource share. The **Pending** status indicates that you have been added to a resource share, but you have not yet accepted or rejected the invitation.

4. To respond to the resource share invitation, select the resource share ID and choose **Accept resource share** to accept the invitation, or **Reject resource share** to decline the invitation. If you reject the invitation, you do not get access to the resources. If you accept the invitation, you gain access to the resources.
Use the resources that are shared with you

After you accept the invitation to join a resource share, you gain the ability to perform specific actions on the shared resources. These actions vary by resource type. For more information, see Shareable AWS resources (p. 3).
Working with shared AWS resources

You can share AWS resources that you own and access AWS resources that are shared with you.

Contents

- Sharing AWS resources owned by you (p. 14)
  - Creating a resource share (p. 14)
  - Updating a resource share (p. 15)
  - Viewing a resource share (p. 15)
  - Viewing your shared resources (p. 16)
  - Viewing the principals with whom you're sharing (p. 16)
  - Deleting a resource share (p. 17)
  - Supported actions on shared resources (p. 17)
- Accessing AWS resources shared with you (p. 17)
  - Accepting and rejecting invitations (p. 17)
  - Viewing resource shares (p. 18)
  - Viewing shared resources (p. 19)
  - Viewing principals sharing with you (p. 19)
  - Leaving a resource share (p. 19)
- AZ IDs for your AWS resources (p. 20)

Sharing AWS resources owned by you

AWS RAM enables you to share the resources that you specify with the principals that you specify. At any time, you can modify resource shares that you have created and delete them when they are no longer needed.

Contents

- Creating a resource share (p. 14)
- Updating a resource share (p. 15)
- Viewing a resource share (p. 15)
- Viewing your shared resources (p. 16)
- Viewing the principals with whom you're sharing (p. 16)
- Deleting a resource share (p. 17)
- Supported actions on shared resources (p. 17)

Creating a resource share

To share resources that you own, create a resource share, add the resources to share, and specify the principals with whom they are to be shared.

To create a resource share, follow the directions in Sharing your AWS resources (p. 10).
Updating a resource share

You can update a resource share at any time. You can add principals, resources, or tags to a resource share that you created. You can revoke access to shared resources by removing principals or resources from a resource share. If you revoke access, principals no longer have access to the shared resources.

To update a resource share using the console

2. In the navigation pane, choose Shared by me, Resource shares.
3. Select the resource share and choose Modify.
4. (Optional) To change the name of the resource share, edit Name.
5. (Optional) To add a resource to the resource share, under Resources, select the type of resource and select the check box next to the resource.
6. (Optional) To remove a resource, locate the resource in the Selected resources panel and choose X.
7. (Optional) To add a principal, type the ID of the AWS account OU, or organization and choose Add.
8. (Optional) To remove a principal, locate it in the Selected principals panel and choose X.
9. (Optional) To add a tag to the resource share, under Tags, choose Add tag and type a tag key and tag value pair.
10. To remove a tag from the resource share, locate it and choose Remove tag.
11. Choose Save changes.

To update a resource share using the AWS CLI

Use the following commands:

- associate-resource-share
- disassociate-resource-share
- tag-resource
- update-resource-share

Viewing a resource share

You can view a list of all the resource shares that you have created. You can see which resources you are sharing and the principals with whom they are shared.

To view your resource shares using the console

2. In the navigation pane, choose Shared by me, Resource shares.
3. Apply a filter to find specific resource shares. You can apply multiple filters to narrow your search.
4. Choose the resource share to review. The following information is available:

- **Summary**—Lists information about the resource share, such as its name, ID, owner, Amazon Resource Name (ARN), creation date, and current status.
- **Shared resources**—Lists the resources that are included in the resource share. Choose the ID of a resource to view it in its service console.
- **Shared principals**—Lists the principals with whom the resources are shared.
- **Tags**—Lists the tag key-value pairs for the resource share.

**To view your resource shares using the AWS CLI**

Use the `get-resource-shares` command.

**Viewing your shared resources**

You can view the resources that are shared by your account, across all resource shares. This enables you to determine which resources you are currently sharing, the number of resource shares they are included in, and the number of principals that have access to them.

**To view the resources that you're sharing using the console**

2. In the navigation pane, choose **Shared by me, Shared resources**.
3. For each shared resource, the following information is available:
   - **Resource ID**—The ID of the resource. Choose the ID of a resource to view it in its service console.
   - **Resource type**—The type of resource.
   - **Last share date**—The date on which the resource was last shared.
   - **Resource shares**—The number of resource shares in which the resource is included. Choose the value to list the resource shares.
   - **Principals**—The number of principals with whom the resource is shared. Choose the value to view the principals.

**To view the resources that you're sharing using the AWS CLI**

Use the `list-resources` command.

**Viewing the principals with whom you're sharing**

You can view the principals with whom you are sharing your resources, across all resource shares. Viewing the principals with whom you are sharing enables you to determine who has access to your shared resources.

**To view the principals with whom you're sharing using the console**

2. In the navigation pane, choose **Shared by me, Principals**.
3. For each principal, the following information is available:
   - **Principal ID**—The ID of the principal.
   - **Resource shares**—The number of resource shares you shared with the principal. Choose the value to view the resource shares.
   - **Resources**—The number of resources you shared with the principal. Choose the value to view the shared resources.

**To view the principals with whom you're sharing using the AWS CLI**

Use the `list-principals` command.
Deleting a resource share

You can delete a resource share at any time. When you delete a resource share, all principals that were associated with the resource share lose access to the shared resources. Deleting a resource share does not delete the shared resources.

The deleted resource share remains visible in the console for a short period after deletion, but its status changes to Deleted.

To delete a resource share using the console

2. In the navigation pane, choose Shared by me, Resource shares.
3. Select the resource share. Be sure to select the correct resource share. You can't recover a resource share after you delete it.
4. Choose Delete, type the confirmation message, and choose Delete.

To delete a resource share using the AWS CLI

Use the delete-resource-share command.

Supported actions on shared resources

You can use the AWS CLI to view the actions that principals can perform on shared resources. For more information, see the get-resource-policies command.

Accessing AWS resources shared with you

AWS RAM enables you to view the resource shares to which you have been added, the shared resources that you can access, and the accounts that have shared resources with you. You can also leave a resource share when you no longer require access to the shared resources.

Contents

- Accepting and rejecting invitations (p. 17)
- Viewing resource shares (p. 18)
- Viewing shared resources (p. 19)
- Viewing principals sharing with you (p. 19)
- Leaving a resource share (p. 19)

Accepting and rejecting invitations

To access shared resources, a principal must add you to a resource share.

If you were added to the resource share by an account in your organization in AWS Organizations, and sharing within your organization is enabled, you are automatically get access to the shared resources.

If you were added to a resource share by one of the following, you receive an invitation to join the resource share:

- An account outside of your organization in AWS Organizations
• An account inside your organization, if sharing with AWS Organizations is not enabled

If you receive an invitation to join a resource share, you must accept it to access the shared resources. If you decline the invitation, you cannot access the shared resources.

You have seven days to accept an invitation to join a resource share. If you do not accept the invitation within seven days, it is automatically declined.

**To respond to invitations**

2. In the navigation pane, choose **Shared with me, Resource shares**.
3. Review the list of resource shares to which you have been added.
   
   The **Status** column indicates your current participation status for the resource share. The **Pending** status indicates that you have been added to a resource share, but you have not yet accepted or rejected the invitation.

4. To respond to the resource share invitation, select the resource share ID and choose **Accept resource share** to accept the invitation, or **Reject resource share** to decline the invitation. If you reject the invitation, you do not get access to the resources. If you accept the invitation, you gain access to the resources.

**To respond to an invitation using the AWS CLI**

Use the following commands:

- `accept-resource-share-invitation`
- `reject-resource-share-invitation`

**Viewing resource shares**

You can view the resource shares to which you have been added. You can see which principals are sharing resources with you and which resources they are sharing.

**To view the resource shares using the console**

2. In the navigation pane, choose **Shared with me, Resource shares**.
3. Apply a filter to find specific resource shares. You can apply multiple filters to narrow your search.
4. The following information is available:
   
   - **Name**—The name of the resource share.
   - **ID**—The ID of the resource share. Choose the ID to view the resource share.
   - **Owner**—The ID of the AWS account that created the resource share.
   - **Status**—The current status of the resource share. Possible values include:
     - **Active**—The resource share is active and available for use.
     - **Deleted**—The resource share has been deleted and is no longer available for use.
     - **Pending**—An invitation to join the resource share is pending.

**To view the resource shares using the AWS CLI**

Use the `get-resource-shares` command.
Viewing shared resources

You can view the shared resources that you can access. You can see which principals are sharing resources and in which resource shares they are included.

To view shared resources using the console
2. In the navigation pane, choose Shared with me, Shared resources.
3. Apply a filter to find specific shared resources. You can apply multiple filters to narrow your search.
4. The following information is available:
   - **Resource ID**—The ID of the resource. Choose the ID of the resource to view it in its service console.
   - **Resource type**—The type of resource.
   - **Last share date**—The date on which the resource was shared with you.
   - **Resource shares**—The number of resource shares in which the resource is included. Choose the value to view the resource shares.
   - **Owner ID**—The ID of the principal who owns the resource.

To view shared resources using the AWS CLI
Use the `list-resources` command.

Viewing principals sharing with you

You can view a list of all the principals that are sharing resources with you. You can see which resources and resource shares they have shared with you.

To view the principals that are sharing resources with you using the console
2. In the navigation pane, choose Shared with me, Principals.
3. Apply a filter to find specific principals. You can apply multiple filters to narrow your search.
4. The following information is available:
   - **Principal ID**—The ID of the principal who is sharing with you.
   - **Resource shares**—The number of resource shares to which the principal has added you. Choose the value to view the resource shares.
   - **Resources**—The number of resources the principal is sharing with you. Choose the value to view the resources.

To view the principals that are sharing resources with you using the AWS CLI
Use the `list-principals` command.

Leaving a resource share

If you no longer need access to resources shared with you, you can leave a resource share at any time. When you leave a resource share, you lose access to the shared resources.

You cannot leave a resource share if you were added to it by an account inside your organization and sharing with AWS Organizations is enabled.
To leave a resource share using the console

2. In the navigation pane, choose Shared with me, Resource shares.
3. Select the resource share.
4. Choose Leave resource share, type the confirmation text, and choose Leave resource share.

To leave a resource share using the AWS CLI

Use the disassociate-resource-share command.

AZ IDs for your AWS resources

To ensure that resources are distributed across the Availability Zones for a Region, we independently map Availability Zones to names for each account. For example, the Availability Zone us-east-1a for your AWS account might not have the same location as us-east-1a for another AWS account. For more information, see Regions and Availability Zones in the Amazon EC2 User Guide.

To identify the location of your resources relative to your accounts, you must use the AZ ID, which is a unique and consistent identifier for an Availability Zone. For example, use1-az1 is an AZ ID for the us-east-1 Region and it is the same location in every AWS account.

To view the AZ IDs for the Availability Zones in your account

2. In the navigation pane, choose Resource Access Manager.
3. The AZ IDs for the current Region are under Your AZ ID.

Viewing AZ IDs enables you to determine the location of resources in one account relative to the resources in another account. For example, if you share a subnet in the Availability Zone with the AZ ID use-az2 with another account, this subnet is available to that account in the Availability Zone whose AZ ID is also use-az2. The AZ ID for each subnet is displayed in the Amazon VPC console.

To view AZ IDs using the AWS CLI

- describe-availability-zones
- DescribeAvailabilityZones
Security in AWS Resource Access Manager

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

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

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS compliance programs. To learn about the compliance programs that apply to AWS Resource Access Manager, see AWS Services in Scope by Compliance Program.

- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company’s requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when using AWS RAM. The following topics show you how to configure AWS RAM to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your AWS RAM resources.

**Topics**

- Data protection in AWS RAM (p. 21)
- Identity and access management for AWS RAM (p. 22)
- AWS RAM permissions (p. 26)
- Logging and monitoring in AWS RAM (p. 33)
- Resilience in AWS RAM (p. 36)
- Infrastructure security in AWS RAM (p. 36)

**Data protection in AWS RAM**

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

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

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

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

Identity and access management for AWS RAM

AWS Identity and Access Management (IAM) is an AWS service that helps an administrator securely control access to AWS resources. IAM administrators control who can be authenticated (signed in) and authorized (have permissions) to use AWS resources. IAM enables you to create users and groups under your AWS account. You control the permissions that users have to perform tasks using AWS resources. You can use IAM for no additional charge. For more information about managing and creating custom IAM policies, see Managing IAM policies.

Topics
• How AWS RAM works with IAM (p. 22)
• Example IAM policies for AWS RAM (p. 24)
• Disabling resource sharing with AWS Organizations (p. 26)

How AWS RAM works with IAM

By default, IAM users don’t have permission to create or modify AWS RAM resources. To allow IAM users to create or modify resources and perform tasks, you must create IAM policies that grant permission to use specific resources and API actions. You then attach those policies to the IAM users or groups that require those permissions.

Contents
• Policy structure (p. 22)
  • Effect (p. 23)
  • Action (p. 23)
  • Resource (p. 23)
  • Condition (p. 23)

Policy structure

An IAM policy is a JSON document that includes the following statements: Effect, Action, Resource, and Condition. An IAM policy typically takes the following form:

```json
{
  
}```
"Statement": [{
    "Effect": "effect",
    "Action": "action",
    "Resource": "arn",
    "Condition": {
        "condition": {
            "key": "value"
        }
    }
}]

Effect

The Effect statement indicates whether the policy allows or denies a user permission to perform an action. The possible values include: Allow and Deny.

Action

The Action statement specifies the AWS RAM API actions for which the policy is allowing or denying permission. For a complete list of the allowed actions, see Actions defined by AWS Resource Access Manager in the IAM User Guide.

Resource

The Resource statement specifies the AWS RAM resources that are affected by the policy. To specify a resource in the statement, you need to use its unique Amazon Resource Name (ARN). For a complete list of the allowed resources, see Resources defined by AWS Resource Access Manager in the IAM User Guide.

Condition

Condition statements are optional. They can be used to further refine the conditions under which the policy applies. AWS RAM supports the following condition keys:

- aws:RequestTag/${TagKey} — Specifies a tag key and value pair that must be used when creating or tagging a resource share.
- aws:ResourceTag/${TagKey} — Indicates that the action can be performed only on resources that have the specified tag key and value pair.
- aws:TagKeys — Specifies the tag keys that can be used when creating or tagging a resource share.
- ram:AllowsExternalPrincipals — Indicates that the action can be performed only on resource shares that allow or deny sharing with external principals. An external principal is an AWS account outside of your AWS organization.
- ram:Principal — Indicates that the action can be performed only on the specified principal.
- ram:RequestedResourceType — Indicates that the action can be performed only on the specified resource type. Resource types must be specified in the following format:
  - AWS App Mesh
    - appmesh:Mesh
  - Amazon Aurora
    - rds:Cluster
  - AWS Certificate Manager Private Certificate Authority
    - acm-pca:CertificateAuthority
  - AWS CodeBuild
    - codebuild:Project
    - codebuild:ReportGroup
  - Amazon EC2
Example IAM policies for AWS RAM

Examples

- Example 1: Allow sharing of specific resources (p. 24)
- Example 2: Allow sharing of specific resource types (p. 25)
- Example 3: Restrict sharing with external AWS accounts (p. 25)

Example 1: Allow sharing of specific resources

You can use an IAM policy to restrict principals to associating only specific resources with resource shares.
For example, the following policy limits principals to sharing only the resolver rule with the specified Amazon Resource Name (ARN).

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                }
            }
        }
    ]
}
```

**Example 2: Allow sharing of specific resource types**

You can use an IAM policy to limit principals to associating only specific resource types with resource shares.

For example, the following policy limits principals to sharing only resolver rules.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ram:RequestedResourceType": "route53resolver:ResolverRule"
                }
            }
        }
    ]
}
```

**Example 3: Restrict sharing with external AWS accounts**

You can use an IAM policy to prevent principals from sharing resources with AWS accounts that are outside of its AWS organization.

For example, the following IAM policy prevents principals from adding external AWS accounts to resource shares.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ram:CreateResourceShare",
            "Resource": "*",
            "Condition": {
                "Bool": {
                    "ram:RequestedAllowsExternalPrincipals": "false"
                }
            }
        }
    ]
}
```
Disabling resource sharing with AWS Organizations

If you previously enabled sharing with AWS Organizations and you no longer need to share resources with your entire organization or organizational units, you can disable sharing. When you disable sharing with AWS Organizations, all organizations or organizational units are removed from the resource shares that you have created and they lose access to the shared resources.

To disable sharing with AWS Organizations

1. Disable trusted access to AWS Organizations using the AWS Organizations `disable-aws-service-access` AWS CLI command.

   ```bash
   # aws organizations disable-aws-service-access --service-principal ram.amazonaws.com
   ```

   **Important**
   When you disable trusted access to AWS Organizations, principals within your organizations are removed from all resource shares and lose access to those shared resources.

2. Use the IAM console, the IAM AWS CLI, or the IAM API to delete the `AWSServiceRoleForResourceAccessManager` service-linked role. For more information, see Deleting a service-linked role in the [IAM User Guide](https://docs.aws.amazon.com/iam/latest/userguide/iam-service-linked-roles.html).

AWS RAM permissions

AWS RAM permissions are policy fragments used by AWS RAM. They control which actions principals are allowed to perform on resources that are shared with them. AWS RAM permissions are used to generate the resource-based policies that are attached to shared resources.

AWS RAM includes default AWS-managed permissions for each supported shareable resource type. These managed permissions are created and managed by AWS, and they define the allowed actions for each shareable resource type. For more information about the default AWS-managed permissions, see [AWS-managed permissions](https://docs.aws.amazon.com/ram/latest/userguide/ram-permissions-managed.html) (p. 27).

**Topics**
- How AWS RAM permissions work (p. 26)
- AWS-managed permissions (p. 27)

How AWS RAM permissions work

When you create a resource share, AWS RAM automatically attaches the default permission for each associated resource type to the resource share. For example, if you create a resource share and associate a subnet and a Capacity Reservation, AWS RAM automatically attaches the subnet and Capacity Reservation permissions to the resource share.

After the resource share has been created, the permissions are provided to the respective resource-owning services. The resource-owning service uses the provided permissions to create resource-based policies for each of the resources included in the resource share. The resulting resource-based policies created by the resource-owning service include the following elements:

- **Resource**—The resource included in the resource share.
- **Effect**—The effect of the AWS RAM permission. Always `allow`.
- **Principal**—The ARNs of the principals associated with the resource share.
- **Action**—The standard actions defined in the AWS RAM permission.
The resource-based policies are attached to the shared resources. They allow the specified principals to perform the allowed actions on the resource.

### AWS-managed permissions

AWS RAM provides the following default AWS-managed permissions:

**Contents**

- AWS App Mesh (p. 27)
- Amazon Aurora (p. 28)
- AWS Certificate Manager Private Certificate Authority (p. 28)
- AWS CodeBuild (p. 28)
- Amazon EC2 (p. 29)
- Amazon EC2 Image Builder (p. 29)
- AWS Glue (p. 30)
- AWS License Manager (p. 30)
- AWS Network Firewall (p. 31)
- AWS Outposts (p. 31)
- AWS Resource Groups (p. 32)
- Amazon Route 53 (p. 32)
- Amazon VPC (p. 32)

---

### AWS App Mesh

AWS RAM provides the following default AWS-managed permissions for shareable AWS App Mesh resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>appmesh:Mesh</td>
<td>Name: AWSRAMDefaultPermissionAppMesh</td>
<td>Allow</td>
<td>• appmesh:CreateVirtualNode</td>
</tr>
<tr>
<td></td>
<td>ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionAppMesh</td>
<td></td>
<td>• appmesh:CreateVirtualRouter</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:CreateRoute</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:CreateVirtualService</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:UpdateVirtualNode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:UpdateVirtualRouter</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:UpdateRoute</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:DeleteVirtualService</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:ListVirtualNodes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:ListVirtualRouters</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:ListRoutes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:ListVirtualServices</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:DescribeVirtualNode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:DescribeVirtualRouter</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:DescribeRoute</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:DescribeVirtualService</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:DeleteVirtualNode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:DeleteVirtualRouter</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• appmesh:DeleteRoute</td>
</tr>
</tbody>
</table>
Amazon Aurora

AWS RAM provides the following default AWS-managed permissions for shareable Amazon Aurora resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>rds:Cluster</td>
<td>Name: AWSRAMDefaultPermissionRDSCluster&lt;br&gt;ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionRDSCluster</td>
<td>Allow</td>
<td>• rds:RestoreDbClusterToPointInTime&lt;br&gt;• rds:DescribeDbClusters</td>
</tr>
</tbody>
</table>

AWS Certificate Manager Private Certificate Authority

AWS RAM provides the following default AWS-managed permissions for shareable ACM Private CA resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
</table>

AWS CodeBuild

AWS RAM provides the following default AWS-managed permissions for shareable AWS CodeBuild resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>codebuild:Project</td>
<td>Name: AWSRAMDefaultPermissionCodeBuildProject&lt;br&gt;ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionCodeBuildProject</td>
<td>Allow</td>
<td>• codebuild:BatchGetBuilds&lt;br&gt;• codebuild:BatchGetProjects&lt;br&gt;• codebuild:ListBuildsForProject</td>
</tr>
</tbody>
</table>
### AWS-managed permissions

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>- codebuild:ListReportsForReportGroup</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- codebuild:DescribeTestCases</td>
</tr>
</tbody>
</table>

### Amazon EC2

AWS RAM provides the following default AWS-managed permissions for shareable Amazon EC2 resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>ec2:CapacityReservation</td>
<td>Name: AWSRAMDefaultPermissionCapacityReservation</td>
<td>Allow</td>
<td>- ec2:RunInstance</td>
</tr>
<tr>
<td></td>
<td>ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionCapacityReservation</td>
<td></td>
<td>- ec2:DescribeCapacityReservations</td>
</tr>
<tr>
<td>ec2:DedicatedHost</td>
<td>Name: AWSRAMDefaultPermissionDedicatedHost</td>
<td>Allow</td>
<td>- ec2:RunInstances</td>
</tr>
<tr>
<td></td>
<td>ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionDedicatedHost</td>
<td></td>
<td>- ec2:StartInstances</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- ec2:DescribeHosts</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- ec2:ModifyInstancePlacement</td>
</tr>
</tbody>
</table>

### Amazon EC2 Image Builder

AWS RAM provides the following default AWS-managed permissions for shareable Amazon EC2 Image Builder resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>imagebuilder:Component</td>
<td>Name: AWSRAMDefaultPermissionImageBuilderComponent</td>
<td>Allow</td>
<td>- imagebuilder:GetComponent</td>
</tr>
<tr>
<td></td>
<td>ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionImageBuilderComponent</td>
<td></td>
<td>- imagebuilder:ListComponents</td>
</tr>
<tr>
<td>imagebuilder:Image</td>
<td>Name: AWSRAMDefaultPermissionImageBuilderImage</td>
<td>Allow</td>
<td>- imagebuilder:GetImage</td>
</tr>
<tr>
<td></td>
<td>ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionImageBuilderImage</td>
<td></td>
<td>- imagebuilder:ListImages</td>
</tr>
<tr>
<td>imagebuilder:ImageRecipe</td>
<td>Name: AWSRAMDefaultPermissionImageBuilderImageRecipe</td>
<td>Allow</td>
<td>- imagebuilder:GetImageRecipe</td>
</tr>
</tbody>
</table>
## AWS Glue

AWS RAM provides the following default AWS-managed permissions for shareable AWS Glue resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>glue:Catalog</td>
<td>Name:</td>
<td>Allow</td>
<td>• glue:GetTable</td>
</tr>
<tr>
<td></td>
<td>AWSRAMDefaultPermissionGlueCatalog</td>
<td></td>
<td>• glue:GetTableVersion</td>
</tr>
<tr>
<td></td>
<td>ARN:</td>
<td></td>
<td>• glue:GetTableVersions</td>
</tr>
<tr>
<td></td>
<td>arn:aws:ram::aws:permission/AWSRAMDefaultPermissionGlueCatalog</td>
<td></td>
<td>• glue:GetPartition</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetPartitions</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:BatchGetPartition</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetDatabase</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetTables</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetDatabases</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:SearchTables</td>
</tr>
<tr>
<td>glue:Database</td>
<td>Name:</td>
<td>Allow</td>
<td>• glue:GetTable</td>
</tr>
<tr>
<td></td>
<td>AWSRAMDefaultPermissionGlueDatabase</td>
<td></td>
<td>• glue:GetTableVersion</td>
</tr>
<tr>
<td></td>
<td>ARN:</td>
<td></td>
<td>• glue:GetTableVersions</td>
</tr>
<tr>
<td></td>
<td>arn:aws:ram::aws:permission/AWSRAMDefaultPermissionGlueDatabase</td>
<td></td>
<td>• glue:GetPartition</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetPartitions</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:BatchGetPartition</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetDatabase</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetTables</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetDatabases</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:SearchTables</td>
</tr>
<tr>
<td>glue:Table</td>
<td>Name:</td>
<td>Allow</td>
<td>• glue:GetTable</td>
</tr>
<tr>
<td></td>
<td>AWSRAMDefaultPermissionGlueTable</td>
<td></td>
<td>• glue:GetTableVersion</td>
</tr>
<tr>
<td></td>
<td>ARN:</td>
<td></td>
<td>• glue:GetTableVersions</td>
</tr>
<tr>
<td></td>
<td>arn:aws:ram::aws:permission/AWSRAMDefaultPermissionGlueTable</td>
<td></td>
<td>• glue:GetPartition</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetPartitions</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:BatchGetPartition</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetDatabase</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetTables</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:GetDatabases</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• glue:SearchTables</td>
</tr>
</tbody>
</table>

## AWS License Manager

AWS RAM provides the following default AWS-managed permissions for shareable AWS License Manager resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>license-manager:LicenseConfiguration</td>
<td>Name:</td>
<td>Allow</td>
<td>• license-manager:GetLicenseConfiguration</td>
</tr>
<tr>
<td></td>
<td>AWSRAMDefaultPermissionLicenseConfiguration</td>
<td></td>
<td>• license-manager:ListLicenseConfigurations</td>
</tr>
<tr>
<td></td>
<td>ARN:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>arn:aws:ram::aws:permission/AWSRAMDefaultPermissionLicenseConfiguration</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
AWS Network Firewall

AWS RAM provides the following default AWS-managed permissions for shareable AWS Network Firewall resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>network-firewall:FirewallPolicy</td>
<td>Name: AWSRAMDefaultPermissionNetworkFirewallPolicy ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionNetworkFirewallPolicy</td>
<td>Allow</td>
<td>• network-firewall:CreateFirewall&lt;br&gt;• network-firewall:UpdateFirewall&lt;br&gt;• network-firewall:AssociateFirewallPolicy&lt;br&gt;• network-firewall:ListFirewallPolicies</td>
</tr>
</tbody>
</table>

AWS Outposts

AWS RAM provides the following default AWS-managed permissions for shareable AWS Outposts resources.

**Note**

For the default AWS-managed permissions for shared subnets and local gateway route tables on Outposts, see [Subnets](#) and [local gateway route tables](#).

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>outposts:Outpost</td>
<td>Name: AWSRAMDefaultPermissionOutpostsOutpost</td>
<td>Allow</td>
<td>• outposts:GetOutpost&lt;br&gt;• outposts:GetOutpostInstanceTypes&lt;br&gt;• outposts:ListOutposts</td>
</tr>
</tbody>
</table>
AWS Resource Access Manager User Guide

AWS-managed permissions

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARN:</td>
<td>arn:aws:ram::aws:permission/ AWSRAMDefaultPermissionOutpostsOutpost</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

AWS Resource Groups

AWS RAM provides the following default AWS-managed permissions for shareable AWS Resource Groups resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>resource-groups:Group</td>
<td>Name: AWSRAMDefaultPermissionResourceGroup ARN: arn:aws:ram::aws:permission/ AWSRAMDefaultPermissionResourceGroup</td>
<td>Allow</td>
<td>• resource-groups:GetGroup • resource-groups:GetGroupConfiguration • resource-groups:ListGroupResources</td>
</tr>
</tbody>
</table>

Amazon Route 53

AWS RAM provides the following default AWS-managed permissions for shareable Amazon Route 53 resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
</table>

Amazon VPC

AWS RAM provides the following default AWS-managed permissions for shareable Amazon VPC resources.

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Permission name and ARN</th>
<th>Effect</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>ec2:PrefixList</td>
<td>Name: AWSRAMDefaultPermissionPrefixList</td>
<td>Allow</td>
<td>• ec2:DescribeManagedPrefixLists • ec2:GetManagedPrefixListEntries</td>
</tr>
<tr>
<td>Resource type</td>
<td>Permission name and ARN</td>
<td>Effect</td>
<td>Actions</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------</td>
<td>--------</td>
<td>---------</td>
</tr>
</tbody>
</table>
| ec2:Subnet    | Name: AWSRAMDefaultPermissionSubnet ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionSubnet | Allow | • ec2:RunInstances  
• ec2:CreateNetworkInterface  
• ec2:DescribeSubnets |
• ec2:CreateTrafficMirrorSession  
• ec2:DeleteTrafficMirrorSession  
• ec2:DescribeTrafficMirrorSessions |
| ec2:TransitGateway | Name: AWSRAMDefaultPermissionTransitGateway ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionTransitGateway | Allow | • ec2:DescribeTransitGateways  
• ec2:CreateTransitGatewayVpcAttachment  
• ec2:ModifyTransitGatewayVpcAttachment  
• ec2:DeleteTransitGatewayVpcAttachment |
| ec2:LocalGateway | RouteTable Name: AWSRAMDefaultPermissionLocalGateway ARN: arn:aws:ram::aws:permission/AWSRAMDefaultPermissionLocalGateway | Allow | • ec2:CreateLocalGatewayRouteTableVpcAttachment  
• ec2:DeleteLocalGatewayRouteTableVpcAttachment  
• ec2:DescribeLocalGatewayRouteTableVpcAttachment  
• ec2:DescribeLocalGatewayRouteTable  
• ec2:DescribeLocalGatewayRouteTableVirtualInterfaceGroup  
• ec2:DescribeLocalGatewayRouteTableVirtualInterface  
• ec2:DescribeLocalGatewayVirtualInterfaceGroup  
• ec2:DescribeLocalGatewayVirtualInterface  
• ec2:DescribeLocalGateways  
• ec2:SearchTransitGatewayRoutes |

Logging and monitoring in AWS RAM

Monitoring is an important part of maintaining the reliability, availability, and performance of AWS RAM and your AWS solutions. You should collect monitoring data from all parts of your AWS solution so that you can more easily debug a multi-point failure if one occurs. AWS provides several tools for monitoring your AWS RAM resources and responding to potential incidents:
Amazon CloudWatch Events

Delivers a near-real-time stream of system events that describe changes in AWS resources. CloudWatch Events enables automated event-driven computing, as you can write rules that watch for certain events and trigger automated actions in other AWS services when these events happen. For more information, see Monitoring AWS RAM using CloudWatch Events (p. 34).

AWS CloudTrail

Captures API calls and related events made by or on behalf of your AWS account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see Logging AWS RAM API calls with AWS CloudTrail (p. 34).

Monitoring AWS RAM using CloudWatch Events

Using Amazon CloudWatch Events, you can set up automatic notifications for specific events in AWS RAM. Events from AWS RAM are delivered to CloudWatch Events in near-real time. You can configure CloudWatch Events to monitor events and invoke targets in response to events that indicate changes to your resource shares. Changes to a resource share trigger events for both the owner of the resource share and the principals that were granted access to the resource share.

When you create an event pattern, the source is `aws.ram`.

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

Logging AWS RAM API calls with AWS CloudTrail

AWS RAM is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in AWS RAM. CloudTrail captures all API calls for AWS RAM as events. The calls captured include calls from the AWS RAM console and code calls to the AWS RAM API operations. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for AWS RAM. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in Event history. Use the information collected by CloudTrail to determine the request that was made to AWS RAM, the requesting IP address, the requester, when it was made, and additional details.

For more information about CloudTrail, see the AWS CloudTrail User Guide.

AWS RAM information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in AWS RAM, that activity is recorded in a CloudTrail event along with other AWS service events in Event history. You can view, search, and download recent events in your AWS account. For more information, see Viewing Events with CloudTrail Event History.

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

- Overview for Creating a Trail
- CloudTrail Supported Services and Integrations
- Configuring Amazon SNS Notifications for CloudTrail
• Receiving CloudTrail Log Files from Multiple Regions and Receiving CloudTrail Log Files from Multiple Accounts

All AWS RAM actions are logged by CloudTrail and are documented in the AWS RAM API Reference. For example, calls to the CreateResourceShare, AssociateResourceShare, and EnableSharingWithAwsOrganization actions generate entries in the CloudTrail log files.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

• Whether the request was made with root or AWS Identity and Access Management (IAM) user credentials.
• Whether the request was made with temporary security credentials for a role or federated user.
• Whether the request was made by another AWS service.

For more information, see the CloudTrail userIdentity Element.

Understanding AWS RAM log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files aren't an ordered stack trace of the public API calls, so they don't appear in any specific order.

The following example shows a CloudTrail log entry for the CreateResourceShare action.

```json
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "NOPIOSFODNN7EXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/admin",
    "accountId": "111122223333",
    "accessKeyId": "BCDIOSFODNN7EXAMPLE",
    "userName": "admin"
  },
  "eventTime": "2018-11-03T04:23:19Z",
  "eventSource": "ram.amazonaws.com",
  "eventName": "CreateResourceShare",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "192.0.1.0",
  "userAgent": "aws-cli/1.16.2 Python/2.7.10 Darwin/16.7.0 botocore/1.11.2",
  "requestParameters": {
    "name": "foo"
  },
  "responseElements": {
    "resourceShare": {
      "allowExternalPrincipals": true,
      "name": "foo",
      "owningAccountId": "111122223333",
      "status": "ACTIVE"
    }
  },
  "requestID": "EXAMPLE0-abcd-1234-mnop-987654567876",
  "eventID": "EXAMPLE0-1234-abcd-hijk-543234565434",
  "readOnly": false,
  "eventType": "AwsApiCall"
}
```
Resilience in AWS RAM

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

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

Infrastructure security in AWS RAM

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

You use AWS published API calls to access AWS RAM through the network. Clients must support Transport Layer Security (TLS) 1.0 or later. We recommend TLS 1.2 or later. Clients must also support cipher suites with perfect forward secrecy (PFS) such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems such as Java 7 and later support these modes.

Additionally, requests must be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.
# Document history for the AWS RAM User Guide

The following table describes the documentation updates for AWS RAM.

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Support for sharing AWS Transit Gateway resources</td>
<td>Use AWS RAM to share transit gateway multicast domains. For more information, see Amazon VPC (p. 8).</td>
<td>December 10, 2020</td>
</tr>
<tr>
<td>Support for sharing AWS Network Firewall resources</td>
<td>Use AWS RAM to share AWS Network Firewall and rule groups. For more information, see AWS Network Firewall (p. 6).</td>
<td>November 17, 2020</td>
</tr>
<tr>
<td>Support for sharing Outposts and local gateway route tables</td>
<td>Use AWS RAM to share Outposts and local gateway route tables. For more information, see AWS Outposts (p. 7) and Amazon VPC (p. 8).</td>
<td>October 15, 2020</td>
</tr>
<tr>
<td>Support for sharing query logs</td>
<td>Use AWS RAM to share Route 53 query logs. For more information, see Amazon Route 53 (p. 8).</td>
<td>September 7, 2020</td>
</tr>
<tr>
<td>Support for sharing ACM Private CA private certificate authorities (CAs)</td>
<td>Use AWS RAM to share ACM Private CA private CAs. For more information, see AWS Certificate Manager Private Certificate Authority (p. 4).</td>
<td>August 17, 2020</td>
</tr>
<tr>
<td>Support for sharing AWS Glue data catalogs, databases, and tables</td>
<td>Use AWS RAM to share AWS Glue data catalogs, databases, and tables. For more information, see AWS Glue (p. 5).</td>
<td>July 07, 2020</td>
</tr>
<tr>
<td>Support for sharing managed prefix lists</td>
<td>Use AWS RAM to share managed prefix lists. For more information, see Amazon EC2 (p. 4).</td>
<td>June 29, 2020</td>
</tr>
<tr>
<td>Support for sharing AWS Outposts customer-owned IPv4 addresses</td>
<td>Use AWS RAM to share AWS Outposts customer-owned IPv4 addresses. For more information, see Amazon EC2 (p. 4).</td>
<td>April 22, 2020</td>
</tr>
<tr>
<td>Support for sharing AWS App Mesh meshes</td>
<td>Use AWS RAM to share meshes. For more information, see AWS App Mesh (p. 3).</td>
<td>January 17, 2020</td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Date</td>
</tr>
<tr>
<td>-------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Support for sharing AWS CodeBuild projects and report groups</td>
<td>Use AWS RAM to share AWS CodeBuild projects and report groups. For more information, see AWS CodeBuild (p. 4).</td>
<td>December 13, 2019</td>
</tr>
<tr>
<td>Support for sharing additional resources</td>
<td>Use AWS RAM to share Amazon EC2 Dedicated Hosts, AWS Resource Groups resource groups, and Amazon EC2 Image Builder components, images, and image recipes. For more information, see Shareable AWS resources (p. 3).</td>
<td>December 02, 2019</td>
</tr>
<tr>
<td>Support for sharing On-Demand Capacity Reservations</td>
<td>Use AWS RAM to share On-Demand Capacity Reservations. For more information, see Amazon EC2 (p. 4).</td>
<td>July 29, 2019</td>
</tr>
<tr>
<td>Support for sharing Aurora DB clusters</td>
<td>Use AWS RAM to share Aurora DB clusters. For more information, see Amazon Aurora (p. 3).</td>
<td>July 02, 2019</td>
</tr>
<tr>
<td>Support for sharing Traffic Mirroring targets</td>
<td>Use AWS RAM to share Traffic Mirroring targets. For more information, see Amazon EC2 (p. 4).</td>
<td>June 25, 2019</td>
</tr>
<tr>
<td>Support for sharing license configurations</td>
<td>Use AWS RAM to share AWS License Manager license configurations. For more information, see AWS License Manager (p. 6).</td>
<td>December 05, 2018</td>
</tr>
<tr>
<td>Support for sharing subnets</td>
<td>Use AWS RAM to share Amazon VPC subnets. For more information, see Amazon EC2 (p. 4).</td>
<td>November 27, 2018</td>
</tr>
<tr>
<td>Support for sharing transit gateways</td>
<td>Use AWS RAM to share Amazon VPC transit gateways. For more information, see AWS License Manager (p. 6).</td>
<td>November 26, 2018</td>
</tr>
<tr>
<td>Support for sharing forwarding rules</td>
<td>Use AWS RAM to share Route 53 forwarding rules. For more information, see Amazon Route 53 (p. 8).</td>
<td>November 20, 2018</td>
</tr>
<tr>
<td>Initial release</td>
<td>This release introduces AWS Resource Access Manager.</td>
<td>November 20, 2018</td>
</tr>
</tbody>
</table>