aws

Console Administration Guide

AWS re:Post Private

AWS re:Post Private: Console Administration Guide

Table of Contents

What is AWS re:Post Private?

AWS re:Post Private is a private version of AWS re:Post for enterprises with Enterprise Support or Enterprise On-Ramp Support plans. It provides access to knowledge and experts to accelerate cloud adoption and increase developer productivity. With your organization-specific private re:Post, you can build an organization-specific developer community that drives efficiencies at scale and provides access to valuable knowledge resources. Additionally, re:Post Private centralizes trusted AWS technical content and offers private discussion forums to improve how your teams collaborate internally and with AWS to remove technical obstacles, accelerate innovation, and scale more efficiently in the cloud.

For more information, see AWS re:Post Private.

Access re:Post Private

Administrators use the AWS re:Post Private console to create their organization-specific private re:Post. When administrators create a private re:Post, they can name their private re:Post and define a subdomain under `*.private.repost.aws`. Administrators for an organization's private re:Post can configure user access using AWS IAM Identity Center and specify one of the following identity sources for authentication: Identity Center directory, Active Directory, or an external identity provider. After configuring the users, console administrators can assign a re:Post Private admin role to one or more users. re:Post Private administrators can customize their private re:Post application in line with organizational branding and knowledge needs. The AWS account team members, such as Technical Account Managers, who are familiar with the organization's architecture and workloads are automatically added to the organization's private re:Post for collaboration.

Administrators for the re:Post Private application can customize branding, add tags to classify content, and select topics of interest for their developers to automatically populate training and technical content. They can also invite users to join their private re:Post for increased collaboration. For more information, see AWS re:Post Private Administration Guide.

Non-administrative users use the re:Post Private application to sign in using credentials that are configured by their administrator. After signing in to a private re:Post, users can browse or search existing content, including tailored training and technical content that are scoped to their topics of interest. Users can also search AWS public technical content directly from their private re:Post and create private threads for internal discussions on AWS public content. Users can collaboratively

solve AWS technical problems and get technical guidance from other users of the private re:Post by asking a question, providing a response, or publishing an article. Users can also convert a discussion thread into an AWS Support case. Users can choose to add the responses from AWS Support to the private re:Post. For more information, see AWS re:Post Private User Guide.

Pricing

Only customers with Enterprise Support (ES) and Enterprise On-Ramp (EOP) Support plans can subscribe to the re:Post Private service. You can choose from the two available pricing tiers - Free tier and Standard tier. The Free tier provides you the ability to explore and try out Standard tier capabilities to full extent for six months before you can seamlessly transition to a paid tier. If you use the Standard tier, then you can pay a monthly subscription per user charge to use re:Post Private. For more information, see Pricing.

How to get started

To get started with re:Post Private, see Prerequisites.

Prerequisites

You must meet the following prerequisites before you can create a new private re:Post or manage an existing private re:Post in AWS re:Post Private:

- You must sign up for an Enterprise or Enterprise On-Ramp Support Plan.
- You must enable AWS IAM Identity Center in the same Region where you want to set up your private re:Post.
- You must create an AWS Identity and Access Management role that has the required permissions to create, manage, and resolve AWS Support cases for you. The re:Post Private service uses this role to make API calls to AWS Support. For more information, see Managing access to AWS Support case creation and management in re:Post Private.

Onboard to re:Post Private through IAM Identity Center

re:Post Private integrates with AWS IAM Identity Center to provide identity federation for your workforce. Through IAM Identity Center, users are redirected to their existing company directory to sign in with their existing credentials. Then, they're seamlessly signed in to their private re:Post. This makes sure that security settings such as password policies and two-factor authentication are enforced. Using IAM Identity Center doesn't impact your existing IAM configuration.

If you don't have an existing user directory or prefer not to federate, then IAM Identity Center offers an integrated user directory that you can use to create users and groups for re:Post Private. re:Post Private doesn't support the use of IAM users and roles to assign permissions within a private re:Post. User permissions within a private re:Post are configured by an administrator on their private re:Post application.

For more information about IAM Identity Center, see What is AWS IAM Identity Center (successor to AWS Single Sign-On). For more information about getting started with IAM Identity Center, see Getting started. To use IAM Identity Center, you must also have AWS Organizations activated for the account.

Important

re:Post Private supports only organization instances of IAM Identity Center.

Security in re:Post Private

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

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

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

Topics

Data protection in AWS re:Post Private

The AWS shared responsibility model applies to data protection in AWS re:Post Private. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on

this infrastructure. You are also responsible for the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see the Data Privacy FAQ. For 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 users with AWS IAM Identity Center or AWS Identity and Access Management (IAM). That way, each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We require TLS 1.2 and recommend TLS 1.3.
- Set up API and user activity logging with AWS CloudTrail. For information about using CloudTrail trails to capture AWS activities, see Working with CloudTrail trails in the *AWS CloudTrail User Guide*.
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing sensitive data that is stored in Amazon S3.
- If you require FIPS 140-3 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-3.

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

Protecting data with encryption

Encryption at rest

re:Post Private uses Amazon Simple Storage Service buckets, Amazon DynamoDB databases, Amazon Neptune databases, and Amazon OpenSearch Service domains that are encrypted at rest using either Amazon managed keys or customer managed keys.

Encryption in transit

re:Post Private uses the HTTPS protocol to communicate with your client application. It uses HTTPS and AWS signatures to communicate with other services on your application's behalf.

Key management

re:Post Private is integrated with AWS Key Management Service and supports AWS KMS keys. You can customize the data encryption settings for your private re:Post when you create it. To do so, you can either choose an existing AWS KMS key or create a new AWS KMS key.

How re:Post Private works with IAM

Before you use IAM to manage access to AWS re:Post Private, you must understand which IAM features are available to use with re:Post Private. To get a high-level view of how re:Post Private and other AWS services work with IAM, see AWS services that work with IAM in the *IAM User Guide*.

re:Post Private identity-based policies

With IAM identity-based policies, you can specify allowed or denied actions. re:Post Private supports specific actions. To learn about the elements that you use in a JSON policy, see IAM JSON policy elements reference in the *IAM User Guide*.

Actions

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

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

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

Policy actions in re:Post Private use the following prefix before the action: `repostspace:`. For example, to grant someone permission to run the re:Post Private `CreateSpace` API operation,

you include the `repostspace:CreateSpace` action in their policy. Policy statements must include either an `Action` or `NotAction` element. re:Post Private 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": [
      "repostspace:CreateSpace",
      "repostspace:DeleteSpace"
```

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

```
"Action": "repostspace:Describe*"
```

To see a list of re:Post Private actions, see Actions defined by re:Post Private in the *IAM User Guide*.

Resources

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

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

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

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

Condition keys

re:Post Private doesn't provide any service-specific condition keys, but it supports using global condition keys. To see all AWS global condition keys, see AWS global condition context keys in the *IAM User Guide*.

Examples

To view examples of re:Post Private identity-based policies, see AWS re:Post Private identity-based policy examples.

re:Post Private resource-based policies

Resource-based policies are JSON policy documents that you attach to a resource. Examples of resource-based policies are IAM role trust policies and Amazon S3 bucket policies. In services that support resource-based policies, service administrators can use them to control access to a specific resource. For the resource where the policy is attached, the policy defines what actions a specified principal can perform on that resource and under what conditions. You must specify a principal in a resource-based policy. Principals can include accounts, users, roles, federated users, or AWS services. Resource-based policies are inline policies that are located in that service. You can't use AWS managed policies from IAM in a resource-based policy.

re:Post Private doesn't support resource-based policies.

Authorization based on tags

re:Post Private supports tagging resources or controlling access based on tags. For more information, see Controlling access to AWS resources using tags.

re:Post Private IAM roles

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

Using temporary credentials with re:Post Private

We strongly recommend using temporary credentials to sign in with federation, assume an IAM role, or to assume a cross-account role. You obtain temporary security credentials by calling AWS STS API operations such as AssumeRole or GetFederationToken.

re:Post Private supports using temporary credentials.

Service-linked roles

Service-linked roles allow AWS services to access resources in other services to complete an action for you. 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.

Service roles

This feature allows a service to assume a service role for you. This role allows the service to access resources in other services to complete an action for you. For more information, see Creating a role to delegate permissions to an AWS service. 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.

Using service-linked roles for re:Post Private

AWS re:Post Private uses AWS Identity and Access Management (IAM) service-linked roles. A service-linked role is a unique type of IAM role that's linked directly to re:Post Private. Service-linked roles are predefined by re:Post Private and include all the permissions that the service requires to call other AWS services on your behalf.

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

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

Service-linked role permissions for re:Post Private

re:Post Private uses the service-linked role named **AWSServiceRoleForrePostPrivate**. re:Post Private uses this service-linked role to publish data to CloudWatch.

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

- `repostspace.amazonaws.com`

The role permissions policy named `AWSrePostPrivateCloudWatchAccess` allows re:Post Private to complete the following actions on the specified resources:

- Action on `cloudwatch`: `PutMetricData`

You must configure permissions to allow your users, groups, or roles to create, edit, or delete a service-linked role. For more information, see Service-linked role permissions in the *IAM User Guide*.

For more information, see AWSrePostPrivateCloudWatchAccess.

Creating a service-linked role for re:Post Private

You don't need to manually create a service-linked role. When you create your first private re:Post in the AWS Management Console, the AWS CLI, or the AWS API, re:Post Private creates the service-linked role for you.

> **Important**
>
> This service-linked role can appear in your account if you completed an action in another service that uses the features supported by this role. Also, if you were using the re:Post Private service before December 1, 2023, when it began supporting service-linked roles, then re:Post Private created the `AWSServiceRoleForrePostPrivate` role in your account. To learn more, see A new role appeared in my AWS account.

If you delete this service-linked role, and then need to create it again, you can use the same process to recreate the role in your account. When you create your first private re:Post, re:Post Private creates the service-linked role for you again.

In the AWS CLI or the AWS API, create a service-linked role with the `repostspace.amazonaws.com` service name. For more information, see Creating a service-linked role in the *IAM User Guide*. If you delete this service-linked role, you can use this same process to create the role again.

Editing a service-linked role for re:Post Private

re:Post Private doesn't allow you to edit the `AWSServiceRoleForrePostPrivate` service-linked role. After you create a service-linked role, you can't change the name of the role because various entities might reference the role. However, you can edit the description of the role using IAM. For more information, see Editing a service-linked role in the *IAM User Guide*.

Deleting a service-linked role for re:Post Private

You don't need to manually delete the `AWSServiceRoleForrePostPrivate` role. When you delete your private re:Post in the AWS Management Console, the AWS CLI, or the AWS API, re:Post Private deletes the service-linked role for you.

You can also use the IAM console, the AWS CLI, or the AWS API to manually delete the service-linked role.

To manually delete the service-linked role using IAM

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

Supported Regions for re:Post Private service-linked roles

re:Post Private supports using service-linked roles in the AWS Regions where the service is available.

Region name	Region identity	Support in re:Post Private
US East (N. Virginia)	us-east-1	Yes
US East (Ohio)	us-east-2	No
US West (N. California)	us-west-1	No
US West (Oregon)	us-west-2	Yes
Africa (Cape Town)	af-south-1	No
Asia Pacific (Hong Kong)	ap-east-1	No
Asia Pacific (Jakarta)	ap-southeast-3	No
Asia Pacific (Mumbai)	ap-south-1	No
Asia Pacific (Osaka)	ap-northeast-3	No
Asia Pacific (Seoul)	ap-northeast-2	No

Region name	Region identity	Support in re:Post Private
Asia Pacific (Singapore)	ap-southeast-1	Yes
Asia Pacific (Sydney)	ap-southeast-2	Yes
Asia Pacific (Tokyo)	ap-northeast-1	No
Canada (Central)	ca-central-1	Yes
Europe (Frankfurt)	eu-central-1	Yes
Europe (Ireland)	eu-west-1	Yes
Europe (London)	eu-west-2	No
Europe (Milan)	eu-south-1	No
Europe (Paris)	eu-west-3	No
Europe (Stockholm)	eu-north-1	No
Middle East (Bahrain)	me-south-1	No
Middle East (UAE)	me-central-1	No
South America (São Paulo)	sa-east-1	No

AWS re:Post Private identity-based policy examples

> **Note**
>
> For greater security, create federated users instead of IAM users whenever possible.

By default, AWS Identity and Access Management users and roles don't have permission to create or modify AWS re:Post Private resources. They also can't perform tasks using the AWS Management Console, AWS CLI, or AWS API. An IAM administrator must create IAM policies that grant users and roles permission to perform specific API operations on the specified resources they need.

The administrator must then attach those policies to the IAM users or groups that require those permissions.

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

Topics

Policy best practices

Identity-based policies determine whether someone can create, access, or delete re:Post Private resources in your account. These actions can incur costs for your AWS account. When you create or edit identity-based policies, follow these guidelines and recommendations:

- **Get started with AWS managed policies and move toward least-privilege permissions** – To get started granting permissions to your users and workloads, use the *AWS managed policies* that grant permissions for many common use cases. They are available in your AWS account. We recommend that you reduce permissions further by defining AWS customer managed policies that are specific to your use cases. For more information, see AWS managed policies or AWS managed policies for job functions in the *IAM User Guide*.
- **Apply least-privilege permissions** – When you set permissions with IAM policies, grant only the permissions required to perform a task. You do this by defining the actions that can be taken on specific resources under specific conditions, also known as *least-privilege permissions*. For more information about using IAM to apply permissions, see Policies and permissions in IAM in the *IAM User Guide*.
- **Use conditions in IAM policies to further restrict access** – You can add a condition to your policies to limit access to actions and resources. For example, you can write a policy condition to specify that all requests must be sent using SSL. You can also use conditions to grant access to service actions if they are used through a specific AWS service, such as AWS CloudFormation. For more information, see IAM JSON policy elements: Condition in the *IAM User Guide*.
- **Use IAM Access Analyzer to validate your IAM policies to ensure secure and functional permissions** – IAM Access Analyzer validates new and existing policies so that the policies adhere to the IAM policy language (JSON) and IAM best practices. IAM Access Analyzer provides more than 100 policy checks and actionable recommendations to help you author secure and

functional policies. For more information, see IAM Access Analyzer policy validation in the *IAM User Guide*.

- **Require multi-factor authentication (MFA)** – If you have a scenario that requires IAM users or a root user in your AWS account, turn on MFA for additional security. To require MFA when API operations are called, add MFA conditions to your policies. For more information, see Configuring MFA-protected API access in the *IAM User Guide*.

For more information about best practices in IAM, see Security best practices in IAM in the *IAM User Guide*.

Allow users to view their own permissions

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

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

```
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Inline policies

Inline policies are policies that you create and manage. You can embed inline policies directly into a user, group, or role. The following policy examples show how to assign permissions to perform AWS re:Post Private actions. For general information about inline policies, see Managing IAM policies in the *AWS IAM User Guide*. You can use the AWS Management Console, AWS Command Line Interface (AWS CLI), or the AWS Identity and Access Management API to create and embed inline policies.

Topics

- Read-only access to re:Post Private
- Full access to re:Post Private

Read-only access to re:Post Private

The following policy grants read access to a user for IAM Identity Center and re:Post Private console. This policy allows the user to perform re:Post Private actions that are read only.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "organizations:DescribeOrganization",
                "organizations:DescribeAccount",

                "sso:DescribeRegisteredRegions",
                "sso:ListDirectoryAssociations",
                "sso:GetSSOStatus",
                "sso:GetManagedApplicationInstance",
```

```
                "sso:ListProfiles",
                "sso:GetProfile",
                "sso:ListProfileAssociations",

                "sso-directory:DescribeDirectory",
                "sso-directory:SearchUsers",
                "sso-directory:SearchGroups",

                "repostspace:GetSpace",
                "repostspace:ListSpaces",
                "repostspace:ListTagsForResource"
            ],
            "Resource": "*"
        }
    ]
}
```

Full access to re:Post Private

The following policy grants full access to a user for IAM Identity Center and re:Post Private console. This policy allows the user to perform all re:Post Private actions.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "organizations:DescribeOrganization",
                "organizations:DescribeAccount",

                "sso:DescribeRegisteredRegions",
                "sso:ListDirectoryAssociations",
                "sso:GetSSOStatus",
                "sso:GetManagedApplicationInstance",
                "sso:ListProfiles",
                "sso:GetProfile",
                "sso:ListProfileAssociations",

                "sso:CreateManagedApplicationInstance",
```

```
                "sso:DeleteManagedApplicationInstance",
                "sso:AssociateProfile",
                "sso:DisassociateProfile",

                "sso-directory:DescribeDirectory",
                "sso-directory:SearchUsers",
                "sso-directory:SearchGroups",

                "kms:ListAliases",
                "kms:DescribeKey",
                "kms:CreateGrant",
                "kms:RetireGrant",

                "repostspace:*"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS managed policies for AWS re:Post Private

Using AWS managed policies makes adding permissions to users, groups, and roles easier than writing policies yourself. It takes time and expertise to create IAM customer managed policies that provide your team with only the permissions they need. Use AWS managed policies to get started quickly. These policies cover common use cases and are available in your AWS account. For more information about AWS managed policies, see AWS managed policies in the *IAM User Guide*.

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

Additionally, AWS supports managed policies for job functions that span multiple services. For example, the `ReadOnlyAccess` AWS managed policy provides read-only access to all AWS services and resources. When a service launches a new feature, AWS adds read-only permissions for new operations and resources. For more information, see AWS managed policies in the *IAM User Guide*.

Topics

AWS managed policy: AWSRepostSpaceSupportOperationsPolicy

This policy allows the AWS re:Post Private service to create, manage, and resolve AWS Support cases that are created through the re:Post Private web application.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "RepostSpaceSupportOperations",
   "Effect": "Allow",
   "Action": [
    "support:AddAttachmentsToSet",
    "support:AddCommunicationToCase",
    "support:CreateCase",
    "support:DescribeCases",
    "support:DescribeCommunications",
    "support:ResolveCase"
   ],
   "Resource": "*"
  }
 ]
}
```

AWS managed policy: AWSrePostPrivateCloudWatchAccess

This policy allows the re:Post Private service to publish data to CloudWatch.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "CloudWatchPublishMetrics",
   "Effect": "Allow",
```

```
   "Action": [
    "cloudwatch:PutMetricData"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "cloudwatch:namespace": [
      "AWS/rePostPrivate",
      "AWS/Usage"
     ]
    }
   }
  }
 ]
}
```

AWS re:Post Private updates to AWS managed policies

View details about updates to AWS managed policies for re:Post Private 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 re:Post Private managed policies since November 26, 2023.

Change	Description	Date
New policy - AWSrePost PrivateCloudWatchAccess	New managed policy for publishing data to CloudWatch	November 26, 2023
New policy - AWSRepost SpaceSupportOperationsPolicy	New managed policy for the AWS Support feature in AWS re:Post Private	November 26, 2023
re:Post Private started tracking changes	re:Post Private started tracking changes for its AWS managed policies	November 26, 2023

Troubleshooting AWS re:Post Private identity and access

Use the following information to help you diagnose and fix common issues that you might encounter when working with re:Post Private and IAM.

Topics

I am not authorized to perform an action in re:Post Private

If you receive an error that you're not authorized to perform an action, your policies must be updated to allow you to perform the action.

The following example error occurs when the `mateojackson` IAM user tries to use the console to view details about a fictional *`my-example-widget`* resource but doesn't have the fictional `repostPrivate:`*`GetWidget`* permissions.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 repostPrivate:GetWidget on resource: my-example-widget
```

In this case, the policy for the `mateojackson` user must be updated to allow access to the *`my-example-widget`* resource by using the `repostPrivate:`*`GetWidget`* action.

If you need help, contact your AWS administrator. Your administrator is the person who provided you with your sign-in credentials.

I am 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 re:Post Private.

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

The following example error occurs when an IAM user named `marymajor` tries to use the console to perform an action in re:Post Private. However, the action requires the service to have

permissions that are granted by a service role. Mary does not have permissions to pass the role to the service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In this case, Mary's policies must be updated to allow her to perform the `iam:PassRole` action.

If you need help, contact your AWS administrator. Your administrator is the person who provided you with your sign-in credentials.

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

Compliance validation for AWS re:Post Private

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

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

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

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying baseline environments on AWS that are security and compliance focused.
- Architecting for HIPAA Security and Compliance on Amazon Web Services – This whitepaper describes how companies can use AWS to create HIPAA-eligible applications.

> **Note**
>
> Not all AWS services are HIPAA eligible. For more information, see the HIPAA Eligible Services Reference.

- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- AWS Customer Compliance Guides – Understand the shared responsibility model through the lens of compliance. The guides summarize the best practices for securing AWS services and map the guidance to security controls across multiple frameworks (including National Institute of Standards and Technology (NIST), Payment Card Industry Security Standards Council (PCI), and International Organization for Standardization (ISO)).
- Evaluating Resources with Rules in the *AWS Config Developer Guide* – The AWS Config service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- AWS Security Hub – This AWS service provides a comprehensive view of your security state within AWS. Security Hub uses security controls to evaluate your AWS 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.
- Amazon GuardDuty – This AWS service detects potential threats to your AWS accounts, workloads, containers, and data by monitoring your environment for suspicious and malicious activities. GuardDuty can help you address various compliance requirements, like PCI DSS, by meeting intrusion detection requirements mandated by certain compliance frameworks.

- AWS Audit Manager – This AWS service helps you continuously audit your AWS usage to simplify how you manage risk and compliance with regulations and industry standards.

Resilience in AWS re:Post Private

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

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

Infrastructure Security in AWS re:Post Private

As a managed service, AWS re:Post Private 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 re:Post Private 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 DHE (Ephemeral Diffie-Hellman) or ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). 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 AWS Identity and Access Management principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.

re:Post Private quotas

AWS re:Post Private provides private re:Posts that you can use in your account in a given AWS Region. When you sign up for re:Post Private, AWS sets default quotas (formerly referred to as limits) on the number of private re:Posts that you can create and size of the private re:Posts.

Service quotas

The following are the default quotas for re:Post Private for your AWS account. You can use the Service Quotas console to view the default quota. None of these quotas are adjustable. You can't request a quota increase.

Resource	Default	Description	Adjustable
Number of private re:Posts	3	The maximum number of private re:Posts in this account in the current Region.	No
Free private re:Post size	10	The maximum size (in GB) of a free private re:Post.	No
Standard private re:Post size	100	The maximum size (in GB) of a standard private re:Post.	No

API throttling limits

The following throttling limits apply per account, per Region in re:Post Private. These quotas can't be increased.

Actions	Token refill rate	Rate of requests	
CreateSpace	1	1	

Actions	Token refill rate	Rate of requests
ListSpaces	10	10
GetSpace	10	10
UpdateSpace	10	10
DeleteSpace	1	1
RegisterAdmin	10	100
DeRegisterAdmin	10	100
SendInvites	1	1
TagResource	10	10
UnTagResource	10	10
ListTagsForResource	10	10

Create, configure, and customize your private re:Post

Topics

Create a new private re:Post

To create a new private re:Post, follow these steps:

1. Open the re:Post Private console at https://console.aws.amazon.com/repost-private/.
2. On the console's homepage, choose **Create private re:Post**.
3. If you don't have IAM Identity Center configured for your account yet, then choose **Open Identity Center**. Follow the instructions in Getting started in the *AWS IAM Identity Center User Guide*.
4. On the **Create private re:Post** page, for **Pricing**, select **Free tier** or **Standard tier** based on your use case. If you already used Free Tier for your account, then **Free tier** option isn't available to you.
5. Under **Details**, do the following:

   For **Name**, enter a unique name for your private re:Post.

   (Optional) For **Description**, enter a brief description for your private re:Post.

   For **Custom subdomain**, enter a custom name for your subdomain.
6. (Optional) To customize your data encryption settings, under **Data encryption**, select **Customize encryption settings**. Then, do either of the following actions:

   For **Choose an AWS KMS key**, select an AWS Key Management Service key or an Amazon Resource Name (ARN).

   -or-

Choose **Create an AWS KMS key**. Then, create the AWS KMS key.

7. (Optional) Under **Service access for Support case integration**, select **Enable service access for this re:Post**.

   > **Note**
   >
   > You can also turn on this option after you create the private re:Post.

   For **Please select an existing IAM role below or create a new role in IAM console**, use the search bar to find your existing IAM role.

   -or-

   Choose **create a new role in IAM console**.

   If you choose to create a new role, then follow the instructions in Create an IAM role.

   If you choose to use an existing service role, then in the search bar, enter the ARN of the role that you want to use. Choose the role from the dropdown list.

   For more information, see Managing access to AWS Support case creation and management in re:Post Private.

8. (Optional) Under **Tags**, choose **Add new tag**. Then enter the following information:

   For **Key**, enter your custom tag key.

   For **Value**, enter your custom tag value.

   To add more tags, choose **Add new tag**.

9. Choose **Create this re:Post**.

A confirmation page will let you know that your private re:Post is being created. You can view the status of the private re:Post in the **Status** field. When your private re:Post is created, the **Status** field displays **Creating**.

It takes approximately 30 minutes for the private re:Post to be created. When your private re:Post is ready, the **Status** field displays **Online**. You can use the **AWS generated subdomain** for your

private re:Post that's listed under the **Settings** tab to access your private re:Post. You can view the **Custom subdomain** for your private re:Post under the **Settings** tab after the review is completed.

Managing access to AWS Support case creation and management in re:Post Private

You must create an AWS Identity and Access Management (IAM) role to manage access to AWS Support case creation and management from AWS re:Post Private. This role performs the following AWS Support actions for you:

- CreateCase
- AddCommunicationToCase
- ResolveCase

After you create the IAM role, 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 private re:Post in the re:Post Private console.

Users in your private re:Post have the same permissions that you grant to the IAM role.

> **Important**
>
> If you change the IAM role or the IAM policy, then your changes apply to the private re:Post that you configured.

Follow these procedures to create your IAM role and policy.

Topics

Use an AWS managed policy or create a customer managed policy

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

> **Tip**
>
> If you don't want to create a policy manually, then we recommend that you use an AWS managed policy instead and skip this procedure. Managed policies automatically have the required permissions for AWS Support. You don't need to update the policies manually. For more information, see AWS managed policy: AWSRepostSpaceSupportOperationsPolicy.

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 re:Post Private

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **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 *`rePostPrivateSupportPolicy`*, 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 AWS Support. After you configure a private re:Post with the role, any user in your private re:Post has the same permissions.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "RepostSpaceSupportOperations",
   "Effect": "Allow",
   "Action": [
    "support:AddAttachmentsToSet",
    "support:AddCommunicationToCase",
    "support:CreateCase",
    "support:DescribeCases",
    "support:DescribeCommunications",
    "support:ResolveCase"
   ],
   "Resource": "*"
  }
 ]
}
```

Note

For a list of AWS managed policies for re:Post Private, see AWS managed policies for AWS re:Post Private.

You can update the policy to remove a permission from AWS Support.

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

- Actions, resources, and condition keys for AWS Support
- Actions, resources, and condition keys for Service Quotas
- Actions, resources, and condition keys for AWS Identity and Access Management

Create an IAM role

After you create the policy, you must create an IAM role, and then attach the policy to that role. You choose this role when you create a private re:Post in the re:Post Private console.

To create a role for AWS Support case creation and management

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Roles**, and then choose **Create role**.
3. For **Trusted entity type**, choose **Custom trust policy**.
4. For **Custom trust policy**, enter the following:

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Principal": {
    "Service": "repostspace.amazonaws.com"
   },
   "Action": [
    "sts:AssumeRole",
    "sts:SetSourceIdentity"
   ]
  }
 ]
}
```

5. Choose **Next**.
6. Under **Permissions policies**, in the search bar, enter the AWS managed policy or a customer managed policy that you created, such as *rePostPrivateSupportPolicy*. Select the check box that's next to the permissions policies that you want the service to have.
7. Choose **Next**.
8. On the **Name, review, and create** page, for **Role name**, enter a name, such as *rePostPrivateSupportRole*.
9. (Optional) For **Description**, enter a description for the role.
10. Review the trust policy and permissions.

11. (Optional) You can use tags as key–value pairs to add metadata to the role. For more information about using tags in IAM, see Tagging IAM resources.
12. Choose **Create role**. You can now choose this role when you configure a private re:Post in the re:Post Private console. See Create a new private re:Post.

For more information, see Creating a role for an AWS service (console) in the *IAM User Guide*.

Troubleshooting

See the following topics to manage access to re:Post Private.

Contents

I want to restrict specific users in my private re:Post from specific actions

By default, users in your private re:Post have the same permissions specified in the IAM policy that you attach to the IAM role that you create. This means that anyone in the private re:Post has read or write access to create and manage AWS Support cases, whether or not they have an AWS account or an IAM user.

We recommend the following best practices:

- Use an IAM policy that has the minimum required permissions to the AWS Support. See AWS managed policy: AWSRepostSpaceSupportOperationsPolicy.

When I configure a private re:Post, I don't see the IAM role that I created

If your IAM role doesn't appear in the **IAM role for re:Post Private;** list, this means that the role doesn't have re:Post Private 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 private re:Post needs permissions to perform the actions that you want. For example, if you want your users in the private re:Post to create support cases, the role must have the `support:CreateCase` permission. re:Post Private assumes this role to perform these actions for you.

If you receive an error about a missing permission for AWS Support, verify that the policy attached to your role has the required permission.

See the previous Example IAM policy.

An error says that my IAM role isn't valid

Verify that you chose the correct role for your private re:Post configuration.

Set up and manage user access using AWS IAM Identity Center

re:Post Private integrates with AWS IAM Identity Center to provide identity federation for your organization's workforce. Use IAM Identity Center to create or connect users from your organization and centrally manage their access across all their AWS accounts and applications. For more information about IAM Identity Center, see What is AWS IAM Identity Center (successor to AWS Single Sign-On). For more information about getting started with IAM Identity Center, see Getting started. To use IAM Identity Center, you must also have AWS Organizations activated for the account.

Customize your private re:Post

You can add one or more administrators to your private re:Post after you create it. Administrators use the re:Post Private application to launch the private re:Post and manage users within it. They can customize branding for the private re:Post, add tags to classify content, and select topics of interest for automatic population of content. For more information, see AWS re:Post Private Administration Guide.

Invite users to your private re:Post

You can add one or more users to your private re:Post after you create it. You can invite users to collaborate within your private re:Post. Users use the re:Post Private application to sign in using

credentials that you configured. After signing in to a private re:Post, users can browse or search existing content, including tailored training and technical content that are scoped to their topics of interest. For more information, see AWS re:Post Private User Guide.

Manage your private re:Post in the re:Post Private console

This section explains how you can manage your private re:Post in the AWS re:Post Private console.

Topics

Add users and groups to your private re:Post

If you're an administrator, you can add users and groups to your private re:Post.

Add users to your private re:Post

1. Open the re:Post Private console at https://console.aws.amazon.com/repost-private/.
2. In the navigation pane, choose **All my private re:Posts**.
3. Choose the private re:Post that you want to manage.
4. Choose the **Users** tab.
5. Under **Users**, choose **Add users and groups**.
6. From the list, select the users that you want to add to your private re:Post. Then, choose **Assign**.

The selected users are added to your private re:Post and listed under the **Users** tab.

Add groups to your private re:Post

1. Open the re:Post Private console at https://console.aws.amazon.com/repost-private/.
2. In the navigation pane, choose **All my private re:Posts**.

3. Choose the private re:Post that you want to manage.
4. Choose the **Groups** tab.
5. Choose **Add users and groups**.
6. From the list, select the groups that you want to add to your private re:Post. Then, choose **Assign**.

The selected groups are added to your private re:Post and listed under the **Groups** tab.

Add users to a group in your private re:Post

Use IAM Identity Center to add new users to an existing group in your private re:Post. For more information, see Add users to groups in the *AWS IAM Identity Center User Guide*.

Invite users and groups to your private re:Post

Follow these steps to invite users and groups to your private re:Post in AWS re:Post Private:

1. Open the re:Post Private console at https://console.aws.amazon.com/repost-private/.
2. In the navigation pane, choose **All my private re:Posts**.
3. Choose the private re:Post that you want to manage.
4. To invite users to your private re:Post, choose the **Users** tab.

   From the list, select the users that you want to invite to your private re:Post. Then, choose **Onboard users to re:Post**.
5. In the **Onboard users to this private re:Post** dialog box, enter the following information:

   For **Subject**, enter the subject for the email message that you're sending.

   For **Body**, enter a welcome message for your private re:Post.

   Choose **Send onboarding email**.
6. To invite groups to your private re:Post, choose the **Groups** tab.

   From the list, select the groups that you want to invite to your private re:Post. Then, choose **Onboard groups to re:Post**.
7. In the **Onboard groups to this private re:Post** dialog box, enter the following information:

For **Subject**, enter the subject for the email message that you're sending.

For **Body**, enter a welcome message for your private re:Post.

Choose **Send onboarding email**.

The welcome message is sent to all selected users and groups with information on how sign in to your private re:Post.

Promote a user in your private re:Post to administrator

To promote a private re:Post user to administrator, follow these steps:

1. Open the re:Post Private console at https://console.aws.amazon.com/repost-private/.
2. In the navigation pane, choose **All my private re:Posts**.
3. Choose the private re:Post that you want to manage.
4. Choose the **Users** tab.
5. Select one or more users that you want to promote to administrator.
6. Choose **Edit role**, and then choose **Make admin**.

The selected users are promoted to administrators. Under the **Users** tab, the **Role** for these users is updated to **Administrator**.

Remove users or groups from your private re:Post

If you're an administrator, then you can remove users or groups from your private re:Post.

Remove users from your private re:Post

1. Open the re:Post Private console at https://console.aws.amazon.com/repost-private/.
2. In the navigation pane, choose **All my private re:Posts**.
3. Choose the private re:Post that you want to manage.
4. Under **Users**, from the list, select the users that you want to remove from your private re:Post. Then, choose **Remove**.

The selected users are removed from your private re:Post. Information about the removed users no longer appears under the **Users** tab.

Remove groups from your private re:Post

1. Open the re:Post Private console at https://console.aws.amazon.com/repost-private/.
2. In the navigation pane, choose **All my private re:Posts**.
3. Choose the private re:Post that you want to manage.
4. Choose the **Groups** tab.
5. From the list, select the groups that you want to remove from your private re:Post. Then, choose **Remove**.

The selected groups are removed from your private re:Post. Information about the removed groups no longer appears under the **Groups** tab.

Add or remove an AWS employee from your private re:Post

If you have an Enterprise or Enterprise On-Ramp Support Plan, then you can add or remove an AWS employee from your private re:Post. Contact Concierge Support or your Technical Account Manager (TAM) for more information.

Delete a private re:Post from re:Post Private

To delete a private re:Post in AWS re:Post Private, follow these steps:

1. Open the re:Post Private console at https://console.aws.amazon.com/repost-private/.
2. In the navigation pane, choose **All my private re:Posts**.
3. Choose the private re:Post that you want to manage, and then choose **Delete**.
4. Select all options to acknowledge and confirm that you want to permanently delete the private re:Post and data that's associated with it.

> **Important**
>
> When you delete the private re:Post, all the configuration information that's related to the private re:Post will be deleted. After the private re:Post is deleted, you can't restore any content from it.

5. Enter the name of your private re:Post when prompted for additional written consent. Then, choose **Delete**.

It takes approximately 30 minutes for your private re:Post to be deleted.

Monitoring AWS re:Post Private

Monitoring is an important part of maintaining the reliability, availability, and performance of AWS re:Post Private and your other AWS solutions. AWS provides the following monitoring tools to watch re:Post Private, report when something is wrong, and take automatic actions when appropriate:

- *Amazon CloudWatch* monitors your AWS resources and the applications you run on AWS in real time. You can collect and track metrics, create customized dashboards, and set alarms that notify you or take actions when a specified metric reaches a threshold that you specify. For example, you can have CloudWatch track CPU usage or other metrics of your Amazon EC2 instances and automatically launch new instances when needed. For more information, see the Amazon CloudWatch User Guide.
- *AWS CloudTrail* captures API calls and related events made by or for your AWS account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see the AWS CloudTrail User Guide.

Monitoring AWS re:Post Private with Amazon CloudWatch

You can monitor AWS re:Post Private using Amazon CloudWatch, which collects raw data and processes it into readable, near real-time metrics. These statistics are kept for 15 months so that you can access historical information and gain a better perspective on how your web application or service is performing. You can also set alarms that watch for certain thresholds, and send notifications or take actions when those thresholds are met. For more information, see the Amazon CloudWatch User Guide.

The re:Post Private service reports the following metrics in the `AWS/rePostPrivate` namespace.

Metric	Description
`NumberOfSpaces`	The number of private re:Posts in the current account. Units: Count
`NumberOfUsers`	The number of users in a private re:Post. This metric uses spaceId as a dimension.

Metric	Description
	Units: Count
`ContentSize`	The amount of content in a private re:Post. This metric uses spaceId as a dimension. Units: Bytes

The following dimensions are supported for the re:Post Private metrics.

Dimension	Description
`spaceId`	The unique identifier for the private re:Post.

Logging AWS re:Post Private API calls using AWS CloudTrail

AWS re:Post Private is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in re:Post Private. CloudTrail captures all API calls for re:Post Private as events. The calls captured include calls from the re:Post Private console and code calls to the re:Post Private API operations. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for re:Post Private. 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 re:Post Private, the IP address from which the request was made, who made the request, when it was made, and additional details.

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

re:Post Private information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in re:Post Private, 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 Working with CloudTrail Event history.

For an ongoing record of events in your AWS account, including events for re:Post Private, 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:

- Creating a trail for your AWS account
- 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 re:Post Private actions are logged by CloudTrail and are documented in the AWS re:Post Private API Reference. re:Post Private supports logging the following actions as events in CloudTrail log files:

- CreateSpace
- DeleteSpace
- DeregisterAdmin
- GetSpace
- ListSpaces
- ListTagsForResource
- RegisterAdmin
- SendInvites
- TagResource
- UntagResource
- UpdateSpace

re:Post Private supports logging the following AWS Support actions as events in the CloudTrail log files:

- CreateCase
- AddCommunicationToCase
- ResolveCase

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 re:Post Private log file entries

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

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


```
{
   "eventVersion": "1.08",
   "userIdentity": {
       "type": "AssumedRole",
       "principalId": "AROAQM47QIR7WLEXAMPLE:user",
       "arn": "arn:aws:sts::123456789012:assumed-role/User/user",
       "accountId": "123456789012",
       "accessKeyId": "EXAMPLE_KEY_ID",
       "sessionContext": {
           "sessionIssuer": {
               "type": "Role",
               "principalId": "AROAQM47QIR7WLEXAMPLE",
               "arn": "arn:aws:iam::123456789012:role/User",
               "accountId": "123456789012",
               "userName": "User"
           },
           "webIdFederationData": {},
           "attributes": {
               "creationDate": "2023-11-06T19:24:39Z",
               "mfaAuthenticated": "false"
```

```
            }
        }
    },
    "eventTime": "2023-11-06T21:37:44Z",
    "eventSource": "repostspace.amazonaws.com",
    "eventName": "CreateSpace",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.251.233.176",
    "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36",
    "requestParameters": {
        "spaceName": "Test space name",
        "spaceSubdomain": "customsubdomain",
        "tagSet": {},
        "tier": "2000",
        "roleArn": "",
        "spaceDescription": "Test space description"
    },
    "responseElements": {
        "spaceId": "SPLPWvQmv9SIWYF30EXAMPLE",
        "Access-Control-Expose-Headers": "x-amzn-errortype, x-amzn-requestid, x-amzn-
errormessage, x-amzn-trace-id, x-amz-apigw-id, date"
    },
    "requestID": "71d815e0-6632-4ec9-9fac-92af3e4a86dc",
    "eventID": "30a6c3da-ce2e-4931-ba5d-b3cc7cf16ec8",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

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

```
 {
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAQM47QIR7WLEXAMPLE:user",
        "arn": "arn:aws:sts::123456789012:assumed-role/User/user",
```

```
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAQM47QIR7WLEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/User",
                "accountId": "123456789012",
                "userName": "User"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-11-07T21:17:19Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-11-07T21:24:23Z",
    "eventSource": "repostspace.amazonaws.com",
    "eventName": "RegisterAdmin",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.251.233.183",
    "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36",
    "requestParameters": {
        "adminId": "08612310-a0f1-7063-3e54-fb2960444dd1",
        "spaceId": "SPlYNZE-ylQEmAXpmEXAMPLE"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-errortype, x-amzn-requestid, x-amzn-
errormessage, x-amzn-trace-id, x-amz-apigw-id, date"
    },
    "requestID": "9939ebbe-8599-4f9a-827b-4995e3006001",
    "eventID": "e1873b18-f80c-4934-9ff2-bf5b35c78031",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```


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

```
{
   "eventVersion": "1.08",
   "userIdentity": {
       "type": "AssumedRole",
       "principalId": "AROAQM47QIR7WLEXAMPLE:user",
       "arn": "arn:aws:sts::123456789012:assumed-role/User/user",
       "accountId": "123456789012",
       "accessKeyId": "EXAMPLE_KEY_ID",
       "sessionContext": {
           "sessionIssuer": {
               "type": "Role",
               "principalId": "AROAQM47QIR7WLEXAMPLE",
               "arn": "arn:aws:iam::123456789012:role/User",
               "accountId": "123456789012",
               "userName": "User"
           },
           "webIdFederationData": {},
           "attributes": {
               "creationDate": "2023-11-09T22:28:23Z",
               "mfaAuthenticated": "false"
           }
       }
   },
   "eventTime": "2023-11-09T22:38:34Z",
   "eventSource": "repostspace.amazonaws.com",
   "eventName": "ListSpaces",
   "awsRegion": "us-west-2",
   "sourceIPAddress": "205.251.233.176",
   "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36",
   "requestParameters": null,
   "responseElements": null,
   "requestID": "95be587b-c04f-4eb0-9269-12fee33ae2e3",
   "eventID": "9777da32-545f-44c4-af0b-1d9109b8cbc3",
   "readOnly": true,
   "eventType": "AwsApiCall",
   "managementEvent": true,
   "recipientAccountId": "123456789012",
   "eventCategory": "Management"
}
```

The following example shows a CloudTrail log entry that demonstrates the `ResolveCase` action. You can use the `sourceIdentity` element in this log entry to identify the user that resolved the case.


```
 {
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAQM47QIR76DQZ7N5WX:create-support-case-
Uk1iHNTWQEOLmR2BR1FDJQ",
        "arn": "arn:aws:sts::123456789012:assumed-role/AWSRepostSpaceRole/create-
support-case-Uk1iHNTWQEOLmR2BR1FDJQ",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAQM47QIR76DQZ7N5WX",
                "arn": "arn:aws:iam::123456789012:role/AWSRepostSpaceRole",
                "accountId": "123456789012",
                "userName": "AWSRepostSpaceRole"
            },
            "attributes": {
                "creationDate": "2023-11-17T21:46:42Z",
                "mfaAuthenticated": "false"
            },
            "sourceIdentity": "28e17330-10f1-705d-7cba-3a62a6b10e2e"
        }
    },
    "eventTime": "2023-11-17T21:46:44Z",
    "eventSource": "support.amazonaws.com",
    "eventName": "ResolveCase",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "54.68.27.29",
    "userAgent": "aws-sdk-nodejs/2.1363.0 linux/v16.20.2 exec-env/AWS_ECS_FARGATE
 promise",
    "requestParameters": {
        "caseId": "case-123456789012-muen-2023-75d2c35481b96357"
    },
    "responseElements": {
        "initialCaseStatus": "unassigned",
        "finalCaseStatus": "resolved"
    },
```

```
    "requestID": "594b91c6-df1c-47e4-a834-d67d67f34b9d",
    "eventID": "7fc9cbe4-c8d5-4d61-a016-e076de272fff",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111111111111",
    "eventCategory": "Management",
    "tlsDetails": {
        "clientProvidedHostHeader": "support.us-west-2.amazonaws.com"
    }
}
```

Troubleshooting re:Post Private

The following information can help you troubleshoot issues with AWS re:Post Private.

Topics

Can't set up my private re:Post in a specific AWS Region

re:Post Private is available only in US East (N. Virginia), US West (Oregon), Europe (Frankfurt), Asia Pacific (Singapore), Asia Pacific (Sydney), Canada (Central), and Europe (Ireland) Regions. Make sure that you're creating your private re:Post in one of these Regions.

Can't set up private re:Post in my account

Make sure that you enabled AWS IAM Identity Center for your account and set up IAM Identity Center in the same Region where you want to create the private re:Post. For more information, see Prerequisites.

Can't manage users or groups in a private re:Post

Be sure that you have the required permissions to edit a private re:Post and manage users and groups within the private re:Post. For more information, see AWS re:Post Private identity-based policy examples.

Document history

The following table describes the documentation releases for AWS re:Post Private:

Change	Description	Date
Update	Added US East (N. Virginia), Asia Pacific (Sydney), Canada (Central), and Europe (Ireland) to supported Regions	May 10, 2024
Update	Added Asia Pacific (Singapore) to supported Regions	March 6, 2024
New resources	Added documentation for AWS managed policies for AWS re:Post Private	November 26, 2023
Initial release	Initial release of the re:Post Private Console Administration Guide	November 26, 2023