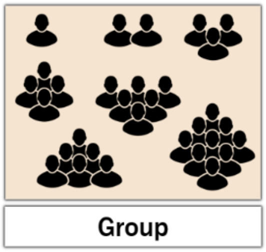
IAM - Groups

What is an IAM Group?



AWS defines an IAM group as;

- Collection of IAM users that let you specify permissions for multiple users, which can make it easier to manage the permissions for those users.
- For example, you could have a group called *Admins* and give that group the types of permissions that administrators typically need.
- Any user in that group automatically has the permissions that are assigned to the group.
- If a new user joins your organization and needs administrator privileges, you can assign the appropriate permissions by adding the user to that group.
- Similarly, if a person changes jobs in your organization, instead of editing that user's permissions, you can remove him or her from the old groups and add him or her to the appropriate new groups.

IAM Groups are not truely real identities, because they can not be identified as a principal in a permission policy. Basically, this is a way for attaching policies to multiple users at any time.

- A group can contain many users, and a user can belong to multiple groups.
- Groups can't be nested; they can contain only users, not other groups.

- There's no default group that automatically includes all users in the AWS account. If you want to have a group like that, you need to create it and assign each new user to it.
- There's a limit to the number of groups you can have, and a limit to how many groups a user can be in. For more information, see IAM and STS Limits.

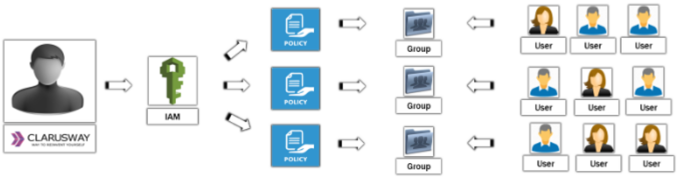
IAM Group Features

An IAM group is essentially a collection of users that makes administration easier over IAM users. Some important features of IAM groups as follows:

- Groups can contain many users, and a user can be in multiple groups.
- Groups have no credentials.
- **Managed IAM** policies can be **attached** to groups.
- **Inline IAM** policies can be **added** to groups.
- Groups can contain only users, but not other groups.
- There's no default group that automatically includes all users in the AWS account.
- The limit of IAM users in a group is equal to the user quota for the account, that is, max 5000.
- An IAM user can be a member of the max. 10 different IAM groups.

Designing IAM Groups

The following diagram shows a simple example of the group designing process.

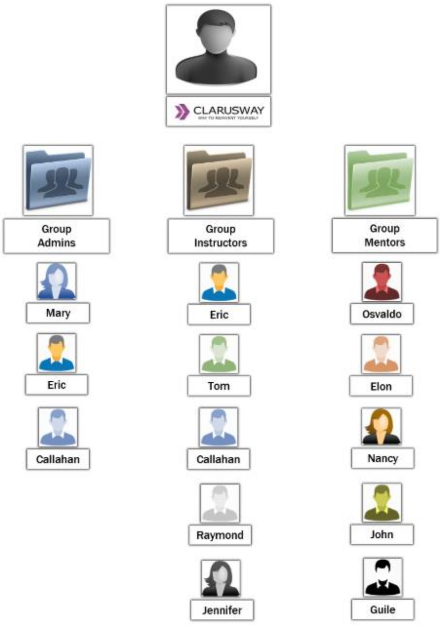


- Create IAM Groups as many as you need (max=300).
- Attach policies to the groups. (One or more managed/inline policies)
- If not, create IAM users for groups.
- Assign users to the groups.

By creating groups in this way, we can assign the needed policies to these groups. Thus, instead of assigning individual policies to each user, we can simplify the management function by putting the users into groups according to the tasks they perform.

IAM Group Example - Clarusway

The following diagram shows a simple example of the Clarusway Company AWS account.



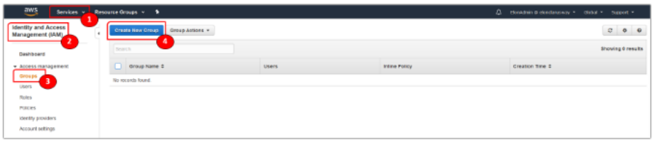
The Clarusway company account owner creates 3 groups of users.

- Admins group for users to create and manage other users as the company grows.
- Instructors group for users that are instructors and access curriculum contents in the Clarusway AWS account.
- Mentors group for users who are mentors and curriculum developers in the Clarusway AWS account.

As you can see in the diagram, some users are in more than one group in terms of their duties.

Creating IAM Groups

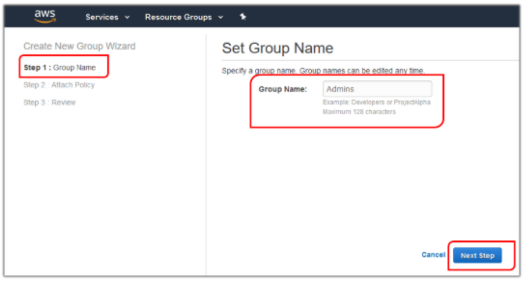
- Sign in to the AWS Management Console.
- Open the **IAM** page using the **Services** tab from the menu bar.
- Click the **Groups** link from the menu on the left.



☐	Group Name	Users	Inline Policy	Creation Time

No records found.

- Let's create our first group by clicking the **"Create New Group"** tab at the top of the page.



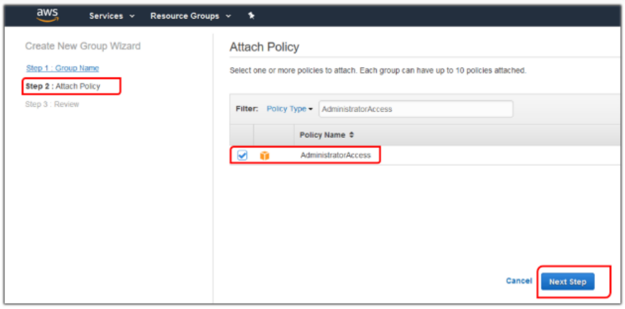
Set Group Name

Specify a group name. Group names can be edited any time.

Group Name: Admins

Example: Developers or ProjectAlpha
Maximum 128 characters

- In the **Group Name** box, type the name of the group and then click **Next Step**.
- Let's write Admins as group name, because we created the **Admins** group.



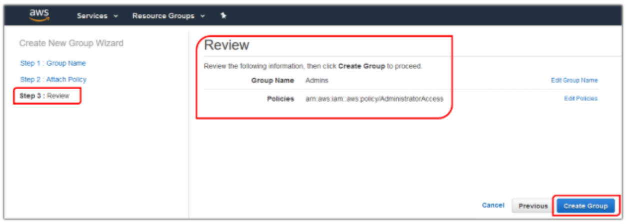
Attach Policy

Select one or more policies to attach. Each group can have up to 10 policies attached.

Filter: Policy Type AdministratorAccess

	Policy Name
☑	AdministratorAccess

- In the list of policies, select the checkbox for the **AdministratorAccess** policy.
- Then click **Next Step**.



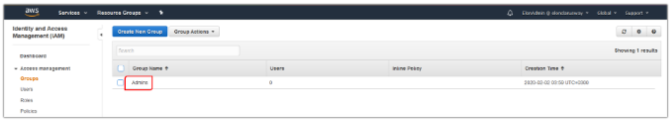
Review

Review the following information, then click **Create Group** to proceed.

Group Name: Admins

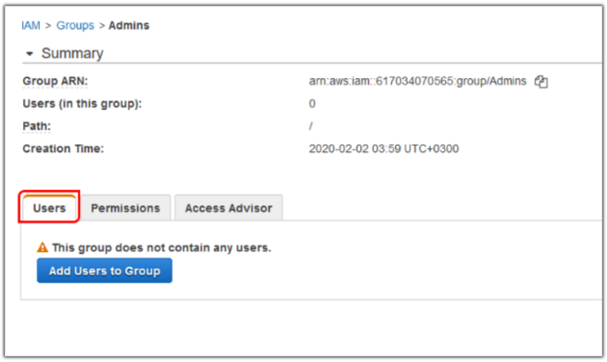
Policies: arn:aws:iam::aws:policy/AdministratorAccess

- Review and click **Create Group** button.
- Congratulations! The Admins group has been created successfully.



☐	Group Name	Users	Inline Policy	Creation Time
☐	Admins	0		2020-02-02 03:59 UTC+0300

- Click the newly created Admins group.



Group ARN: arn:aws:iam::617034070565:group/Admins

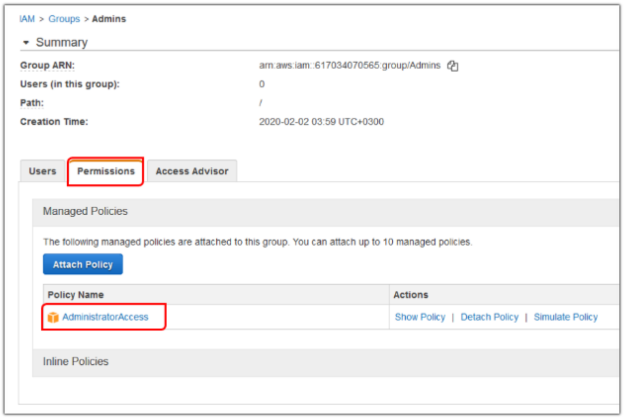
Users (in this group): 0

Path: /

Creation Time: 2020-02-02 03:59 UTC+0300

This group does not contain any users.

- As you can see, there is no user assigned to the group yet.



Group ARN: arn:aws:iam::617034070565:group/Admins

Users (in this group): 0

Path: /

Creation Time: 2020-02-02 03:59 UTC+0300

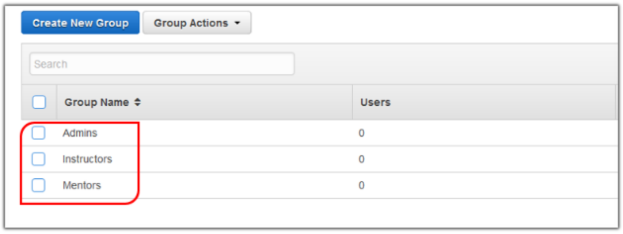
The following managed policies are attached to this group. You can attach up to 10 managed policies.

Policy Name	Actions
AdministratorAccess	Show Policy | Detach Policy | Simulate Policy

- AdministratorAccess policy is attached to the group.

You can create other Instructors and Mentors groups by adding the following policies in a similar way to practice.

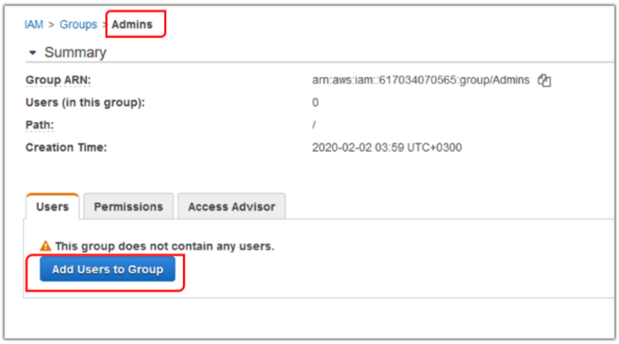
- **Instructors group:** DatabaseAdministrator, AmazonS3FullAccess.
- **Mentors group:** AmazonS3FullAccess, AWSCloud9User.



☐	Group Name	Users
☐	Admins	0
☐	Instructors	0
☐	Mentors	0

Adding Users to IAM Groups

- Select Admins group and then click Add Users to Group tab.



Group ARN: arn:aws:iam::617034070565:group/Admins

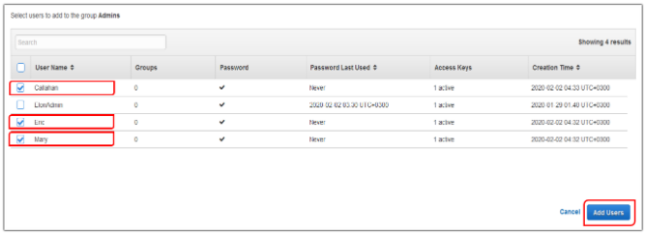
Users (in this group): 0

Path: /

Creation Time: 2020-02-02 03:59 UTC+0300

This group does not contain any users.

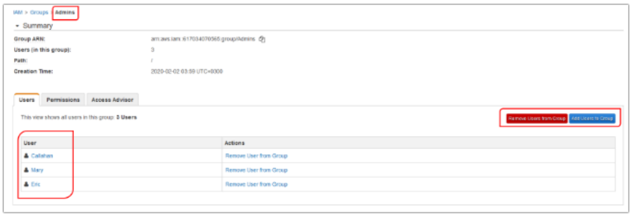
- Select the users you want to add to the Admins group and then click the Add users tab.



Select users to add to the group **Admins**

☐	User Name	Groups	Password	Password Last Used	Access Keys	Creation Time
☑	Callahan	0	✓	Never	1 active	2020-02-02 04:33 UTC+0300
☐	LionAdmin	0	✓	2020-02-02 03:30 UTC+0300	1 active	2020-01-29 01:40 UTC+0300
☑	Eric	0	✓	Never	1 active	2020-02-02 04:32 UTC+0300
☑	Mary	0	✓	Never	1 active	2020-02-02 04:32 UTC+0300

Congratulations! Users have been successfully added to the Admins group.



Group ARN: arn:aws:iam::617034070565:group/Admins

Users (in this group): 3

Path: /

Creation Time: 2020-02-02 03:59 UTC+0300

This view shows all users in this group: **3 Users**

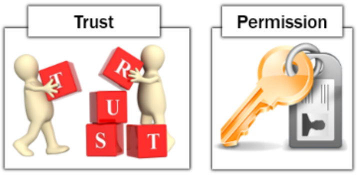
User	Actions
Callahan	Remove User from Group
Mary	Remove User from Group
Eric	Remove User from Group

IAM - Roles

What is an IAM Role?

It is the authorization system that we determine how and with which authorizations an identity can access the AWS resources.

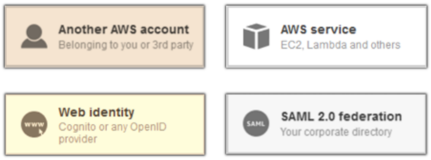
- An IAM role, similar to an IAM user, is an IAM identity that has specific permissions that you can create in your account.
- A role is meant for anyone who needs it to be assumable. An entity is allowed to assume the role and when the role is assumed, the entity gains the permission that that role has.
- By creating a role, in fact, you define entities that can assume this role via policies.



Every role has two policies: A trust policy and a permission policy.

- The trust policy defines who can assume the role. It is a JSON policy document in which you define the principals that you *trust* to assume the role.
- The permission policy is just an IAM policy that gives that role some permissions on some things. It can be thought of as the same as an IAM policy on a group. A permissions document in JSON format in which you define what actions and resources the role can use. The document is written according to the rules of the IAM policy language.

Who can assume an IAM Role?



Another AWS account
Belonging to you or 3rd party

AWS service
EC2, Lambda and others

Web identity
Cognito or any OpenID provider

SAML 2.0 federation
Your corporate directory

IAM roles are a secure way to grant permissions to entities that you trust. Examples of entities include the following:

- IAM user in another account
- Application code running on an EC2 instance that needs to perform actions on AWS resources
- An AWS service that needs to act on resources in your account to provide its features
- Users from a corporate directory who use identity federation with SAML

For example; We can create a role that we can assign to a virtual machine-EC2, so that we can access the S3 service with EC2 instance and read the files there and save files to this service. Or if we want a user in another AWS account to use some resources on our account, we can assign it to the account by assigning a role to do various transactions with our resources.

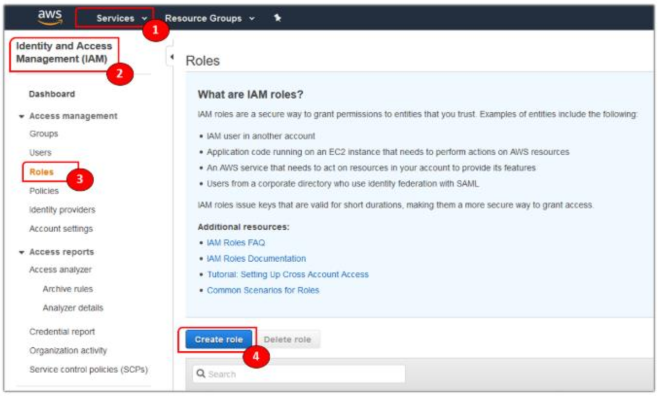
Roles basically contain 2 main components.

- First of all, when creating the role, it is determined who can use this role, that is, where this role can be assigned.
- Then, it is determined what powers the reliable resource to which this role is assigned will have and what it can do. This is done through the IAM policy files we have learned.

Creating IAM Role

If you use the AWS Management Console, a wizard guides you through the steps for creating a role. The wizard has slightly different steps depending on whether you're creating a role for an AWS service, for an AWS account, or for a federated user.

- Sign in to the AWS Management Console.
- Open the **IAM** page using the **Services** tab from the menu bar.
- Click the **Roles** link from the menu on the left.



Roles

What are IAM roles?

IAM roles are a secure way to grant permissions to entities that you trust. Examples of entities include the following:

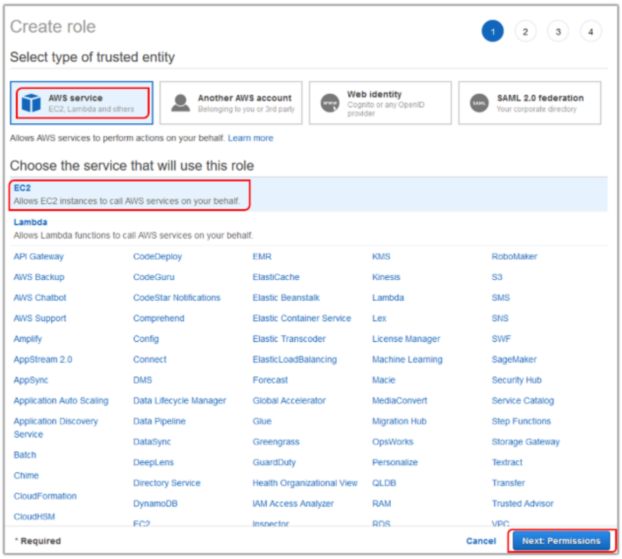
- IAM user in another account
- Application code running on an EC2 instance that needs to perform actions on AWS resources
- An AWS service that needs to act on resources in your account to provide its features
- Users from a corporate directory who use identity federation with SAML

IAM roles issue keys that are valid for short durations, making them a more secure way to grant access.

Additional resources:

- IAM Roles FAQ
- IAM Roles Documentation
- Tutorial: Setting Up Cross Account Access
- Common Scenarios for Roles

- Click **Create Role** tab.



Create role

Select type of trusted entity

AWS service EC2, Lambda and others

Another AWS account Belonging to you or 3rd party

Web identity Cognito or any OpenID provider

SAML 2.0 federation Your corporate directory

Allows AWS services to perform actions on your behalf. Learn more

Choose the service that will use this role

EC2
Allows EC2 instances to call AWS services on your behalf.

Lambda
Allows Lambda functions to call AWS services on your behalf.

API Gateway
AWS Backup
AWS Chatbot
AWS Support
Amplify
AppStream 2.0
AppSync
Application Auto Scaling
Application Discovery Service
Batch
Chime
CloudFormation
CloudHSM

CodeDeploy
CodeGuru
CodeStar Notifications
Comprehend
Config
Connect
DMS
Data Lifecycle Manager
Data Pipeline
DataSync
DeepLens
Directory Service
DynamoDB
EC2

EMR
ElastiCache
Elastic Beanstalk
Elastic Container Service
Elastic Transcoder
ElasticLoadBalancing
Forecast
Global Accelerator
Glue
Greengrass
GuardDuty
Health Organizational View
IAM Access Analyzer
Inspector

KMS
Kinesis
Lambda
Lex
License Manager
Machine Learning
Macie
MediaConvert
Migration Hub
OpsWorks
Personalize
QLDB
RAM
RDS

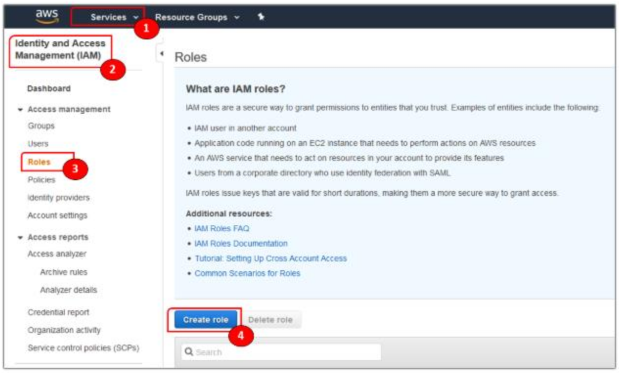
RoboMaker
S3
SMS
SNS
SWF
SageMaker
Security Hub
Service Catalog
Step Functions
Storage Gateway
Textract
Transfer
Trusted Advisor
VPC

* Required

- Here you need to determine the type of entity that can assume the role. Depending on the type of entity you select, the steps of creating the role would differ.

Creating IAM Role - AWS Service

- Sign in to the AWS Management Console.
- Open the **IAM** page using the **Services** tab from the menu bar.
- Click the **Roles** link from the menu on the left.



Roles

What are IAM roles?

IAM roles are a secure way to grant permissions to entities that you trust. Examples of entities include the following:

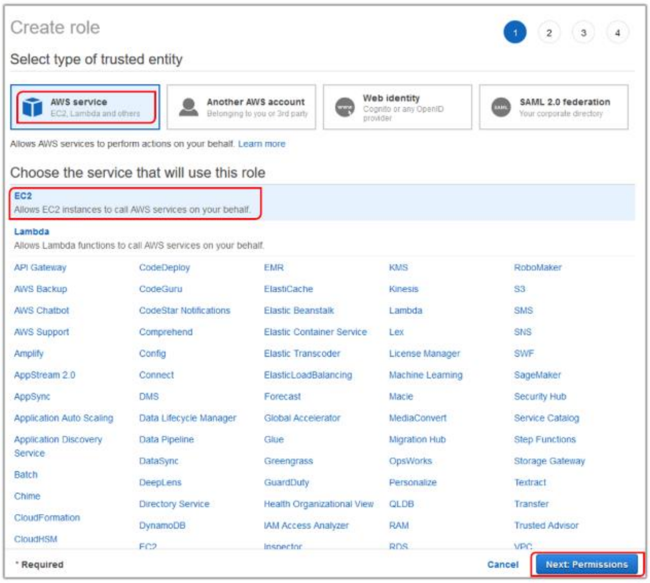
- IAM user in another account
- Application code running on an EC2 instance that needs to perform actions on AWS resources
- An AWS service that needs to act on resources in your account to provide its features
- Users from a corporate directory who use identity federation with SAML

IAM roles issue keys that are valid for short durations, making them a more secure way to grant access.

Additional resources:

- IAM Roles FAQ
- IAM Roles Documentation
- Tutorial: Setting Up Cross Account Access
- Common Scenarios for Roles

- Click the **Create Role** tab.



Create role ① 2 3 4

Select type of trusted entity

- **AWS service** EC2, Lambda and others
- **Another AWS account** Belonging to you or 3rd party
- **Web identity** Cognito or any OpenID provider
- **SAML 2.0 federation** Your corporate directory

Allows AWS services to perform actions on your behalf. Learn more

Choose the service that will use this role

EC2
Allows EC2 instances to call AWS services on your behalf.

Lambda
Allows Lambda functions to call AWS services on your behalf.

API Gateway
AWS Backup
AWS Chatbot
AWS Support
Amplify
AppStream 2.0
AppSync
Application Auto Scaling
Application Discovery Service
Batch
Chime
CloudFormation
CloudHSM

CodeDeploy
CodeGuru
CodeStar Notifications
Comprehend
Config
Connect
DMS
Data Lifecycle Manager
Data Pipeline
DataSync
DeepLens
Directory Service
DynamoDB
EC2

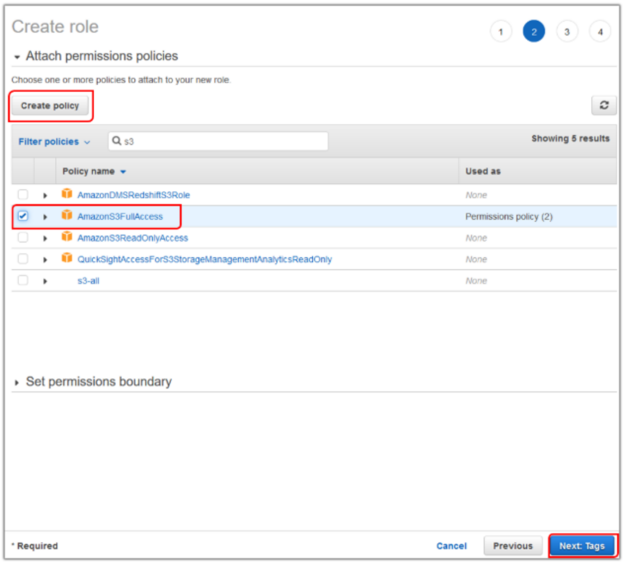
EMR
ElastiCache
Elastic Beanstalk
Elastic Container Service
Elastic Transcoder
ElasticLoadBalancing
Forecast
Global Accelerator
Glue
Greengrass
GuardDuty
Health Organizational View
IAM Access Analyzer
Inspector

KMS
Kinesis
Lambda
Lex
License Manager
Machine Learning
Macie
MediaConvert
Migration Hub
OpsWorks
Personalize
QLDB
RAM
RDS

RoboMaker
S3
SMS
SNS
SWF
SageMaker
Security Hub
Service Catalog
Step Functions
Storage Gateway
Textract
Transfer
Trusted Advisor
VPC

* Required   Cancel   **Next: Permissions**

- Choose **the service** that will use this role.
- Click **Next: Permissions** tab.



Create role 1 ② 3 4

▾ Attach permissions policies

Choose one or more policies to attach to your new role.

Create policy

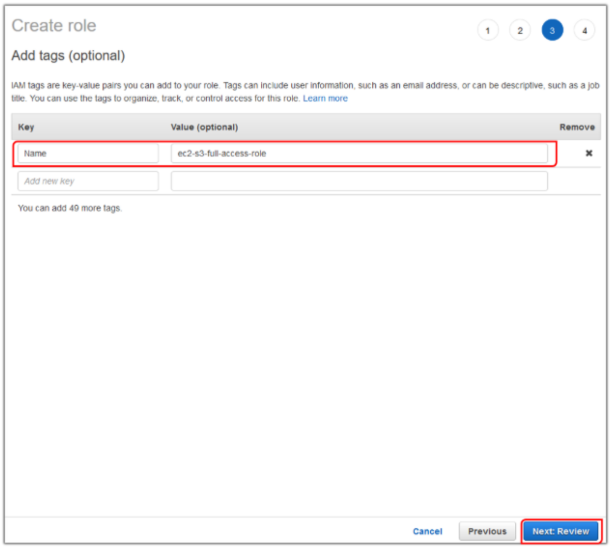
Filter policies ⌄   🔍 s3   Showing 5 results

	Policy name ▾	Used as
□	▸ AmazonDMSRedshiftS3Role	None
☑	▸ AmazonS3FullAccess	Permissions policy (2)
□	▸ AmazonS3ReadOnlyAccess	None
□	▸ QuickSightAccessForS3StorageManagementAnalyticsReadOnly	None
□	▸ s3-all	None

▸ Set permissions boundary

* Required   Cancel   Previous   **Next: Tags**

- If possible, select the policy to use as the permission policy or choose **Create policy** tab to open a new browser tab and create a new policy from scratch.
- Let's select AmazonS3FullAccess policy via the search bar for this example.
- This policy will allow EC2 services to access all S3 resources when EC2 assumes this role.
- Then click **Next: Tags** tab.



Create role 1 2 ③ 4

Add tags (optional)

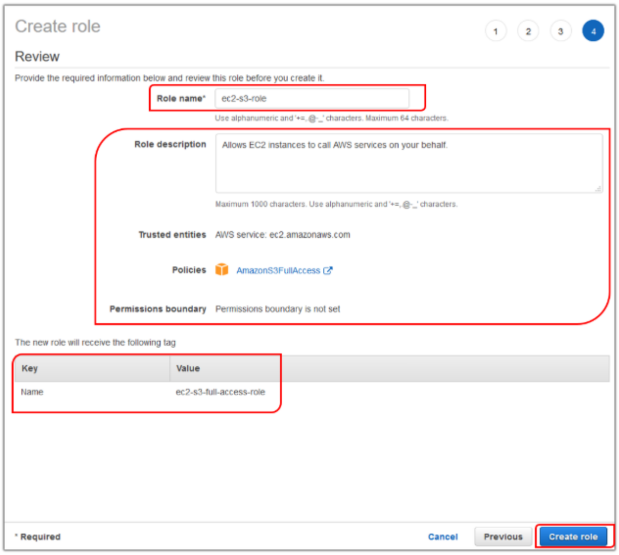
IAM tags are key-value pairs you can add to your role. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this role. Learn more

Key	Value (optional)	Remove
Name	ec2-s3-full-access-role	×
Add new key		

You can add 49 more tags.

Cancel   Previous   **Next: Review**

- Optionally, type a tag or tags as key-value pairs.
- Click **Next: Review** tab.



Create role 1 2 3 ④

Review

Provide the required information below and review this role before you create it.

Role name* ec2-s3-role
Use alphanumeric and '+=,.@-_' characters. Maximum 64 characters.

Role description Allows EC2 instances to call AWS services on your behalf.
Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.

Trusted entities AWS service: ec2.amazonaws.com

Policies AmazonS3FullAccess

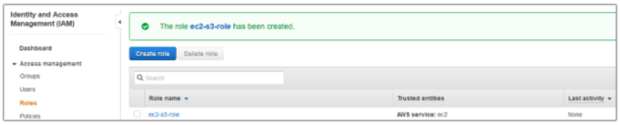
Permissions boundary Permissions boundary is not set

The new role will receive the following tag

Key	Value
Name	ec2-s3-full-access-role

* Required   Cancel   Previous   **Create role**

- Type a **name** for your role. Role names must be **unique** within your AWS account.
- Use a maximum of 64 **alphanumerics** and **'+=,.@-_' characters**.
- Click **Create role** tab.



Identity and Access Management (IAM)

Dashboard
▾ Access management
Groups
Users
Roles
Policies

The role **ec2-s3-role** has been created.

Create role   Delete role

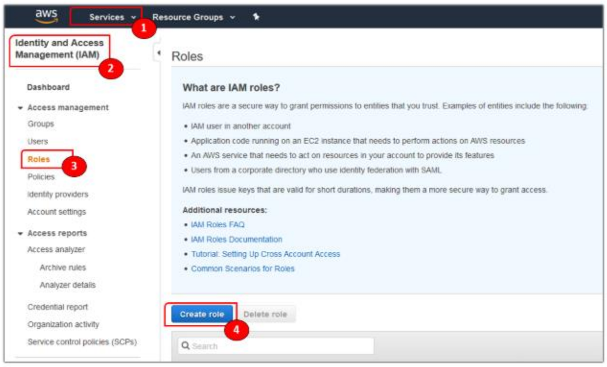
🔍 Search

	Role name ▾	Trusted entities	Last activity ▾
□	ec2-s3-role	AWS service: ec2	None

Congratulations! The role has been created successfully.

Creating IAM Role - Another AWS Account

- Sign in to the AWS Management Console.
- Open the **IAM** page using the **Services** tab from the menu bar.
- Click the **Roles** link from the menu on the left.



Identity and Access Management (IAM)

Dashboard
▾ Access management
Groups
Users
Roles
Policies
Identity providers
Account settings
▾ Access reports
Access analyzer
Archive rules
Analyzer details
Credential report
Organization activity
Service control policies (SCPs)

Roles

What are IAM roles?

IAM roles are a secure way to grant permissions to entities that you trust. Examples of entities include the following:

- IAM user in another account
- Application code running on an EC2 instance that needs to perform actions on AWS resources
- An AWS service that needs to act on resources in your account to provide its features
- Users from a corporate directory who use identity federation with SAML

IAM roles issue keys that are valid for short durations, making them a more secure way to grant access.

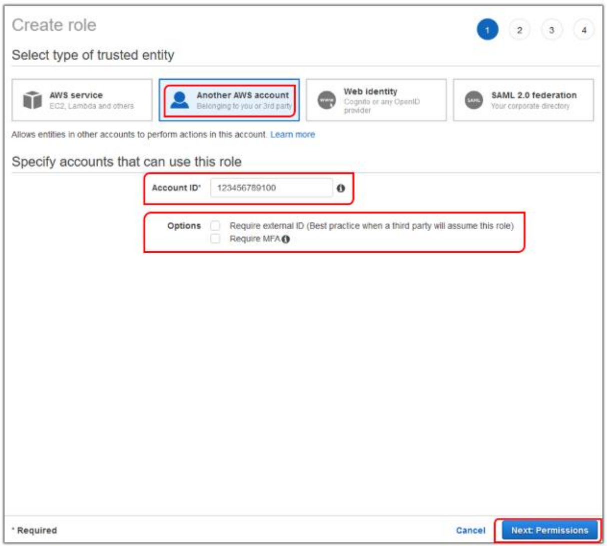
Additional resources:

- IAM Roles FAQ
- IAM Roles Documentation
- Tutorial: Setting Up Cross Account Access
- Common Scenarios for Roles

Create role   Delete role

🔍 Search

- Click the **Create Role** tab.



Create role ① 2 3 4

Select type of trusted entity

- **AWS service** EC2, Lambda and others
- **Another AWS account** Belonging to you or 3rd party
- **Web identity** Cognito or any OpenID provider
- **SAML 2.0 federation** Your corporate directory

Allows entities in other accounts to perform actions in this account. Learn more

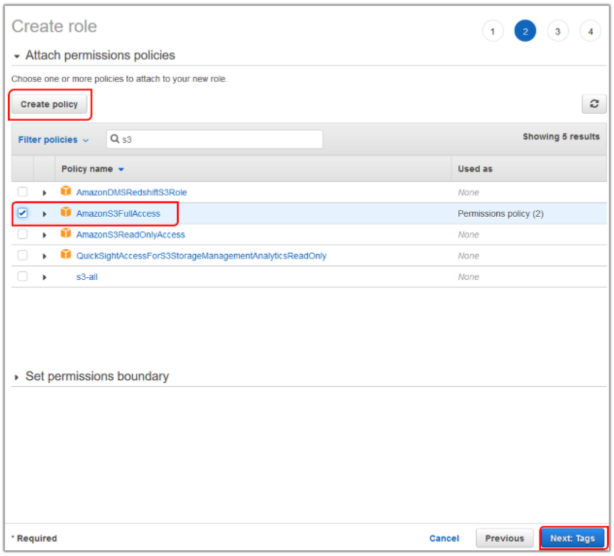
Specify accounts that can use this role

Account ID* 123456789100

Options □ Require external ID (Best practice when a third party will assume this role)
□ Require MFA

* Required   Cancel   **Next: Permissions**

- Choose **Another AWS account** option that will use this role.
- For **Account ID**, type the AWS account ID to which you want to grant access to your resources.
- If you are granting permissions to users from an account that you do not control, and the users will assume this role programmatically, then select **Require external ID**. The external ID can be any word or number that is agreed upon between you and the administrator of the third-party account. This option automatically adds a condition to the trust policy that allows the user to assume the role only if the request includes the correct ID.
- If you want to restrict the role to users who sign in with multi-factor authentication (MFA), select **Require MFA**. This adds a condition to the role's trust policy that checks for an MFA sign-in.
- Click **Next: Permissions** tab.



Create role

Attach permissions policies

Choose one or more policies to attach to your new role.

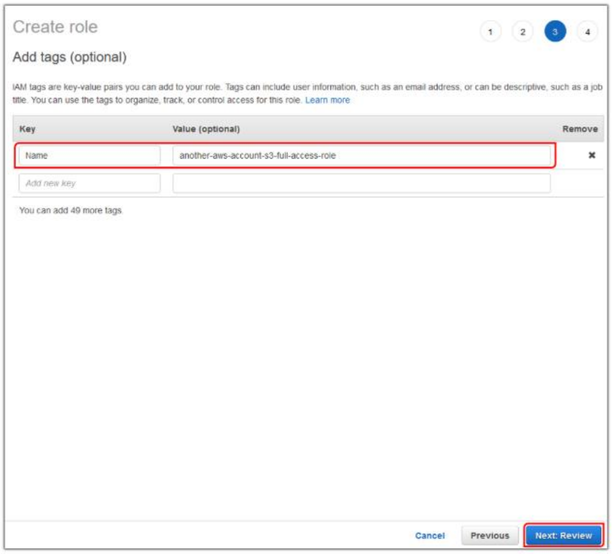
Filter policies: s3 — Showing 5 results

	Policy name	Used as
□	AmazonDMSRedshiftS3Role	None
☑	AmazonS3FullAccess	Permissions policy (2)
□	AmazonS3ReadOnlyAccess	None
□	QuickSightAccessForS3StorageManagementAnalyticsReadOnly	None
□	s3-all	None

Set permissions boundary

* Required

- If possible, select the policy to use as the permission policy or choose **Create policy** to open a new browser tab and create a new policy from scratch.
- Let's select AmazonS3FullAccess policy via the search bar for this example.
- This policy will allow another AWS accounts to access all S3 resources when EC2 assumes this role.
- Then click **Next: Tags** tab.



Create role

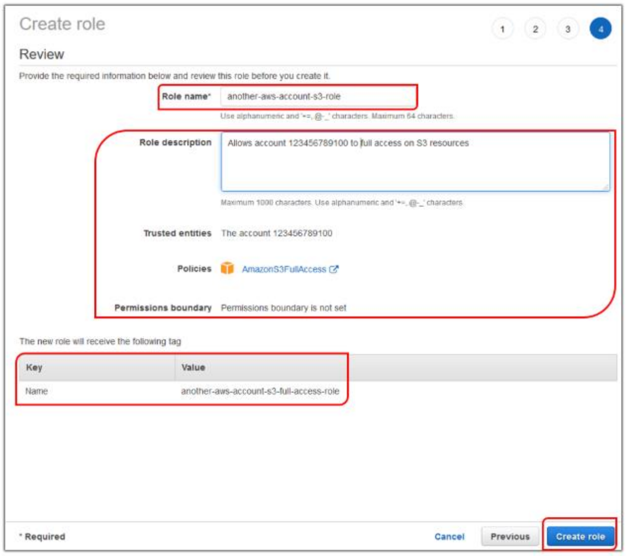
Add tags (optional)

IAM tags are key-value pairs you can add to your role. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this role. Learn more

Key	Value (optional)	Remove
Name	another-aws-account-s3-full-access-role	✖
Add new key		

You can add 49 more tags.

- Optionally, type a tag or tags as key-value pairs.
- Click Next: Review tab.



Create role

Review

Provide the required information below and review this role before you create it.

Role name*: another-aws-account-s3-role

Use alphanumeric and '+=,.@-_' characters. Maximum 64 characters.

Role description: Allows account 123456789100 to full access on S3 resources

Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.

Trusted entities: The account 123456789100

Policies: AmazonS3FullAccess

Permissions boundary: Permissions boundary is not set

The new role will receive the following tag

Key	Value
Name	another-aws-account-s3-full-access-role

* Required

- Type a **name** for your role. Role names must be **unique** within your AWS account.
- Use a maximum of 64 **alphanumerics** and **'+=,.@-_' characters**.
- Click **Create role** tab.
- The role would be created successfully.