Immersion Day

Identity & Access Management

Rev 2015-01

Table of Contents

Overview

This lab will walk you through connecting to the instance and configuring security credentials so that you can interact with the AWS APIs and command line tools. This lab will cover the following topics:

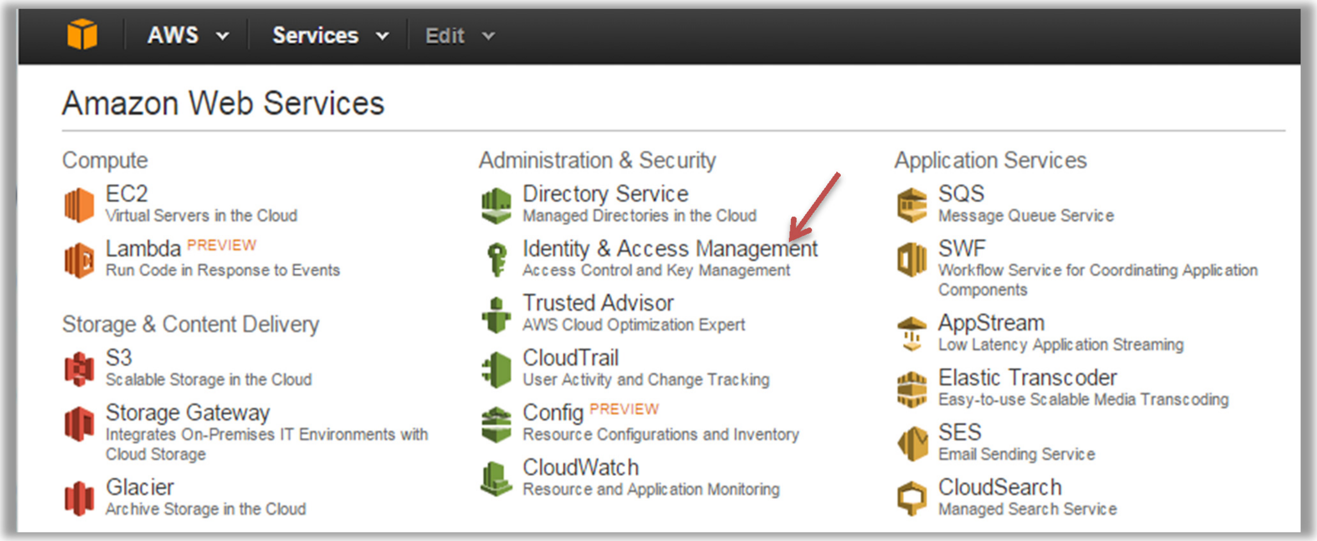
- Creating an IAM Group and adding an IAM user to the Group.
- Exploring the properties of an IAM User.
- Creating an IAM Role for EC2

Copyright 2015, Amazon Web Services, All Rights Reserved

Managing AWS IAM User and Security Credentials

Creating IAM Users and Groups

To generate AWS API credentials, go to IAM dashboard in the AWS console.



Amazon Web Services

Compute

- EC2 Virtual Servers in the Cloud
- Lambda PREVIEW Run Code in Response to Events

Storage & Content Delivery

- S3 Scalable Storage in the Cloud
- Storage Gateway Integrates On-Premises IT Environments with Cloud Storage
- Glacier Archive Storage in the Cloud

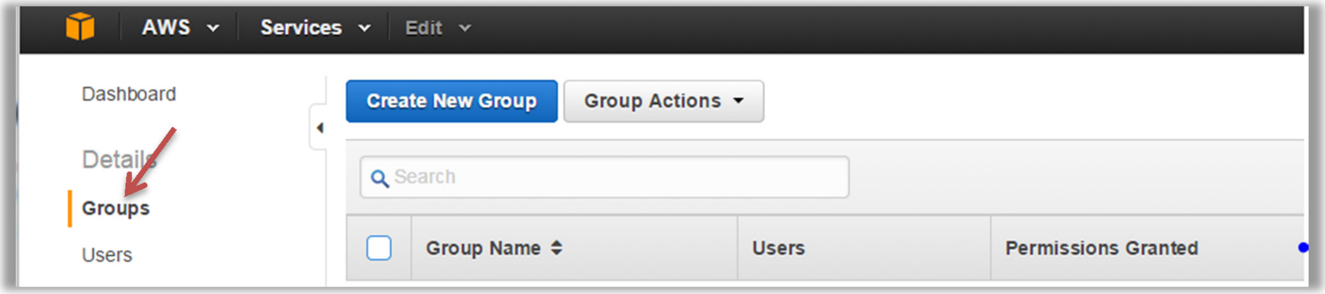
Administration & Security

- Directory Service Managed Directories in the Cloud
- Identity & Access Management Access Control and Key Management
- Trusted Advisor AWS Cloud Optimization Expert
- CloudTrail User Activity and Change Tracking
- Config PREVIEW Resource Configurations and Inventory
- CloudWatch Resource and Application Monitoring

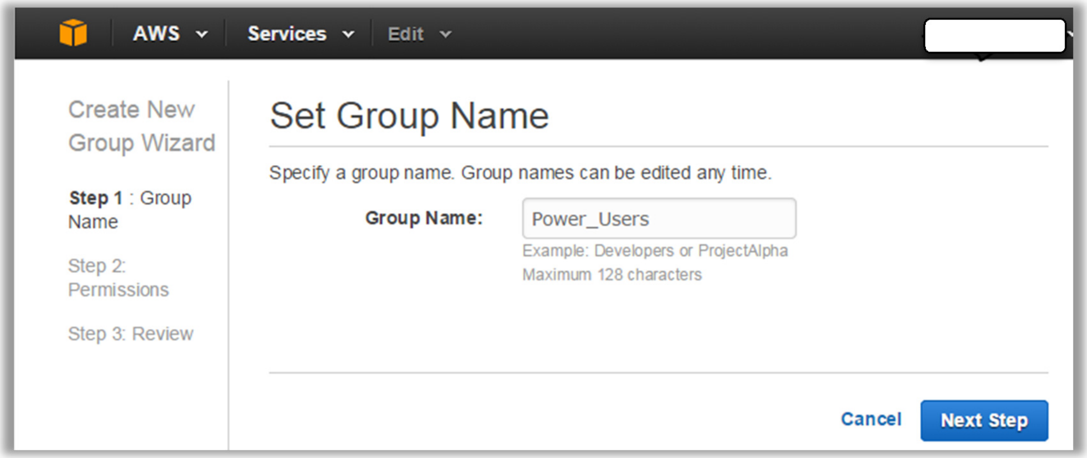
Application Services

- SQS Message Queue Service
- SWF Workflow Service for Coordinating Application Components
- AppStream Low Latency Application Streaming
- Elastic Transcoder Easy-to-use Scalable Media Transcoding
- SES Email Sending Service
- CloudSearch Managed Search Service

Select **"Groups"** then Click the "**Create a New Group**" button.



Type **Power_Users** into the Group Name: text box and click **Next Step**.



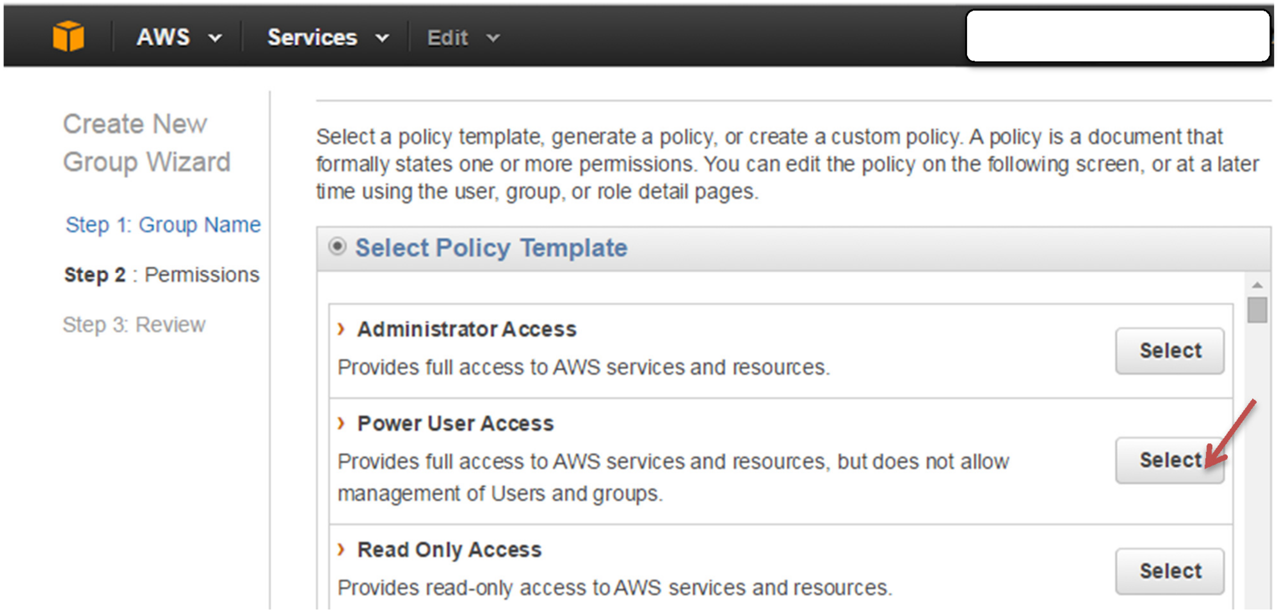
Set Group Name

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

Group Name: Power_Users

Example: Developers or ProjectAlpha
Maximum 128 characters

Click the **Select** button next to **Power User Access.**



Select a policy template, generate a policy, or create a custom policy. A policy is a document that formally states one or more permissions. You can edit the policy on the following screen, or at a later time using the user, group, or role detail pages.

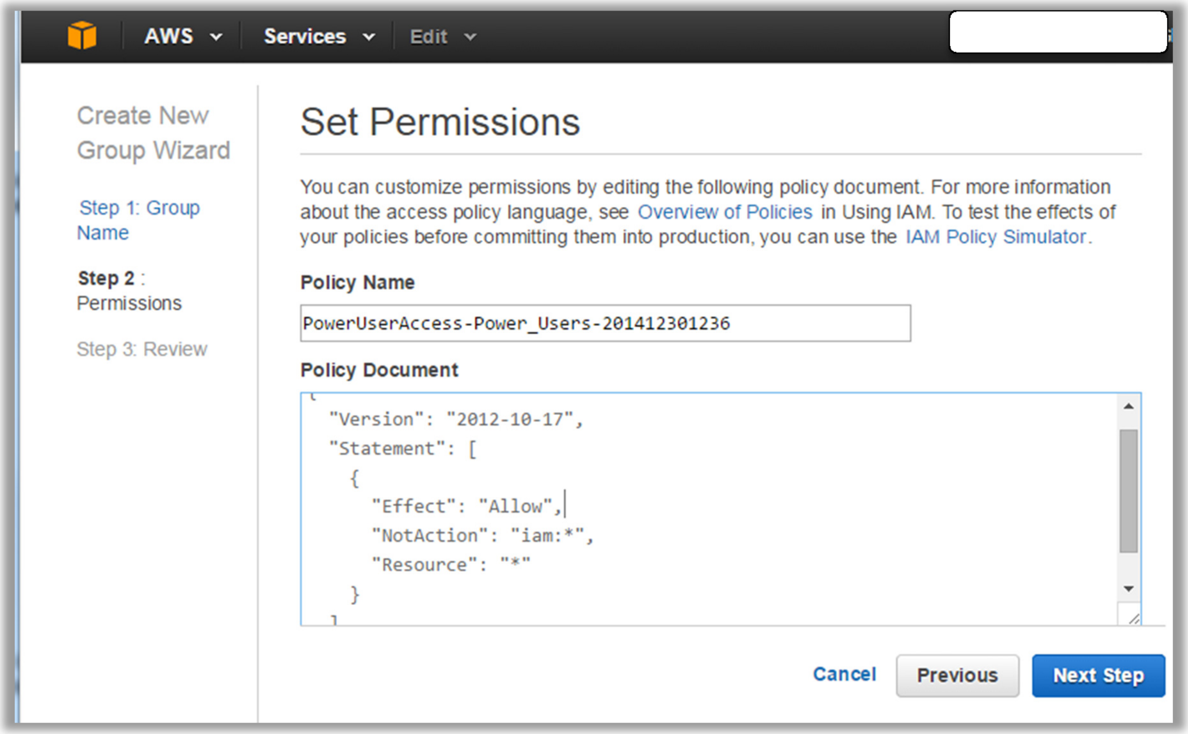
● Select Policy Template

› Administrator Access
Provides full access to AWS services and resources.

› Power User Access
Provides full access to AWS services and resources, but does not allow management of Users and groups.

› Read Only Access
Provides read-only access to AWS services and resources.

This will create a “Power User” IAM policy allowing group members to perform any AWS action except perform IAM management (e.g. notice the “NotAction iam:* line). **Click Next Step**:



Set Permissions

You can customize permissions by editing the following policy document. For more information about the access policy language, see Overview of Policies in Using IAM. To test the effects of your policies before committing them into production, you can use the IAM Policy Simulator.

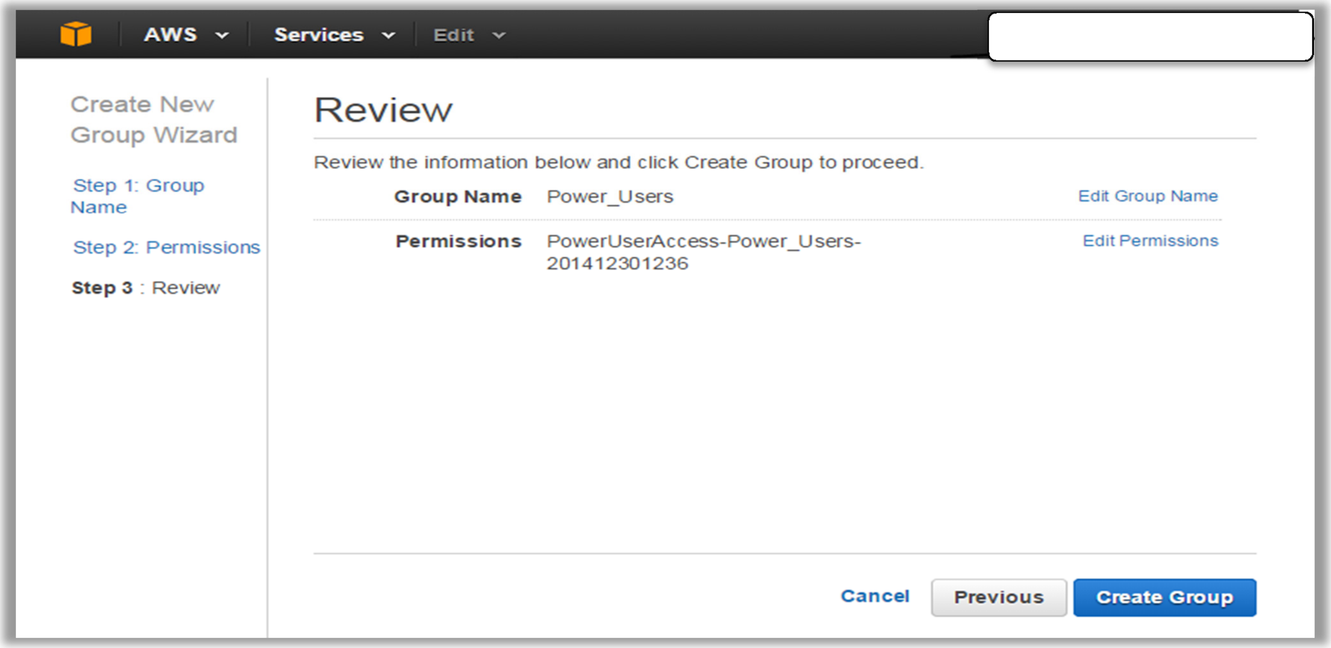
Policy Name

PowerUserAccess-Power_Users-201412301236

Policy Document

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "NotAction": "iam:*",
      "Resource": "*"
    }
  ]
```

Click on “**Create Group**” to create group



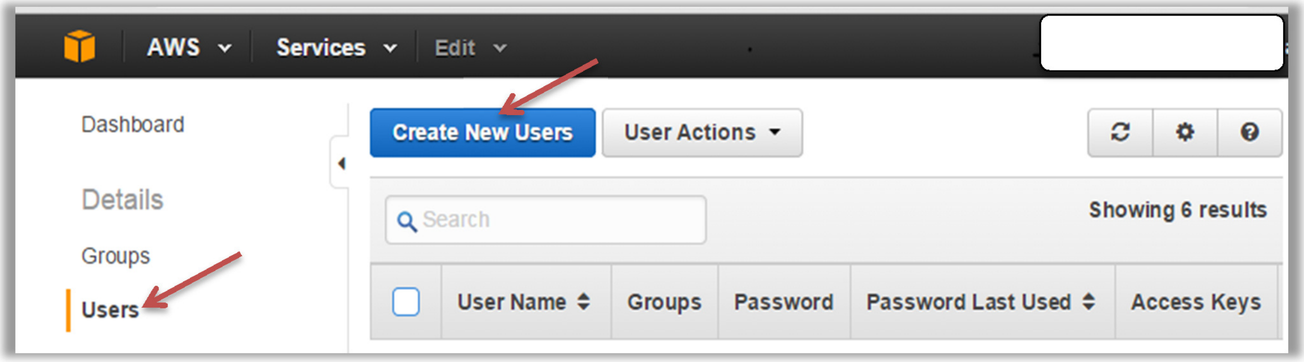
Review

Review the information below and click Create Group to proceed.

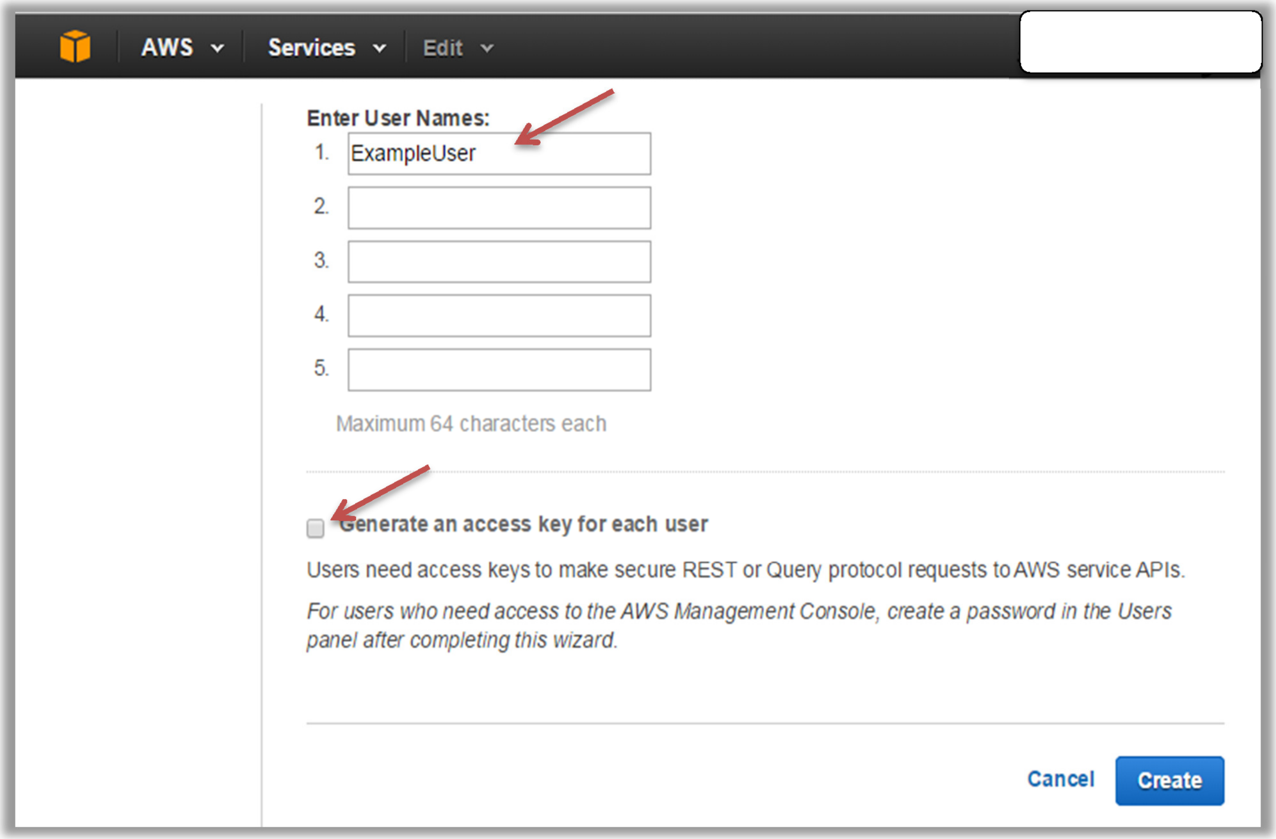
Group Name: Power_Users

Permissions: PowerUserAccess-Power_Users-201412301236

Select “**Users**” then Click the “**Create a New Users**” button.



Enter *ExampleUser* in the first text box under **Enter User Names:** Unselect the check box next to **Generate an access key for each user** and click **Create**.



Enter User Names:

1. ExampleUser
2.
3.
4.
5.

Maximum 64 characters each

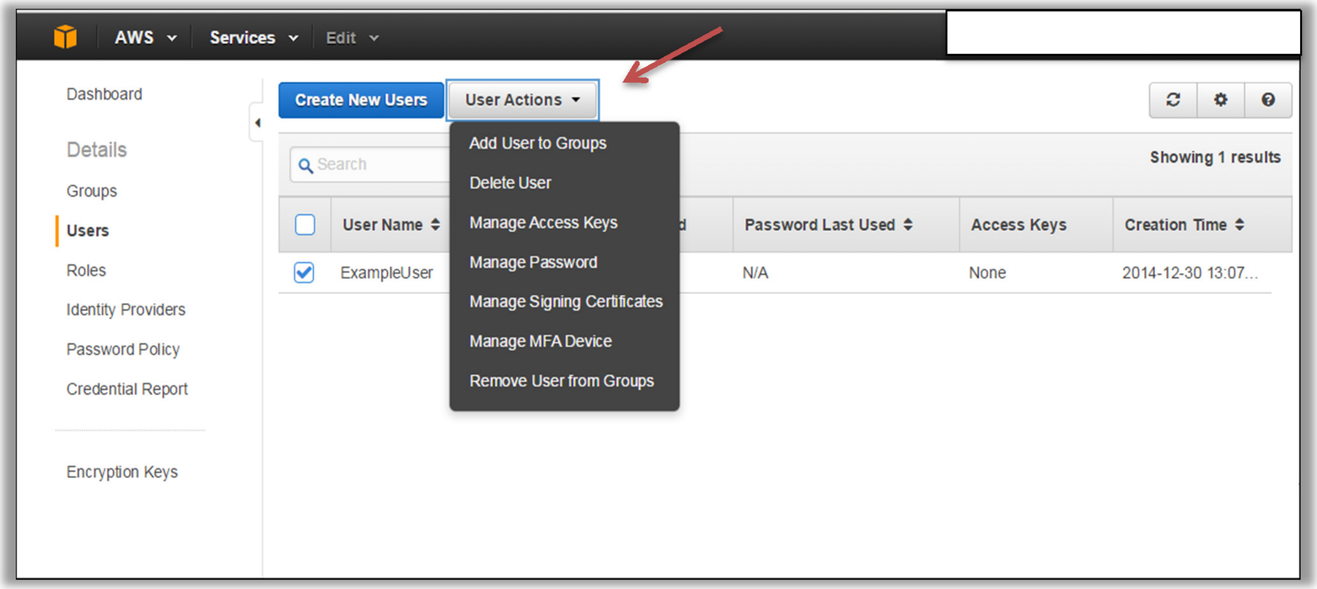
□ Generate an access key for each user

Users need access keys to make secure REST or Query protocol requests to AWS service APIs.

For users who need access to the AWS Management Console, create a password in the Users panel after completing this wizard.

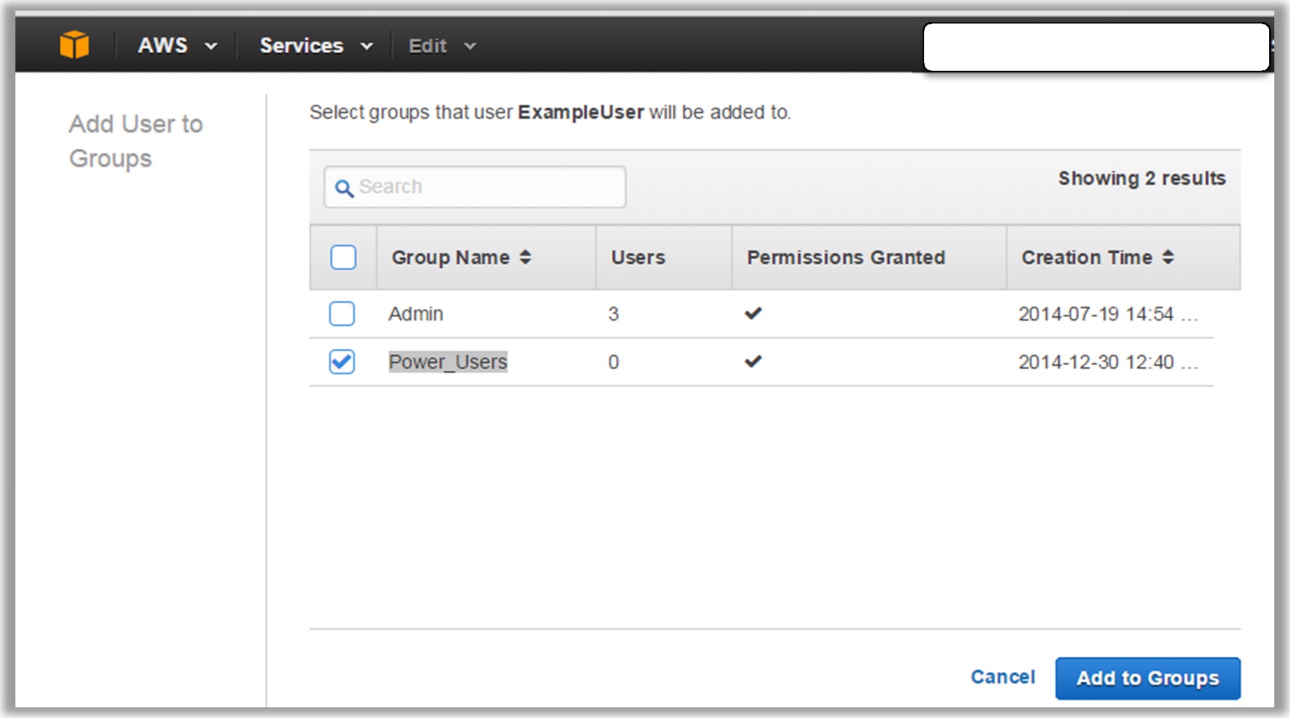
Copyright 2015, Amazon Web Services, All Rights Reserved

Select ExampleUser then Click on **User Actions** and select the **Add User to Groups** menu option.



☐	User Name	d	Password Last Used	Access Keys	Creation Time
☑	ExampleUser		N/A	None	2014-12-30 13:07...

Select the Power_User group the click **Add to Groups**



Select groups that user **ExampleUser** will be added to.

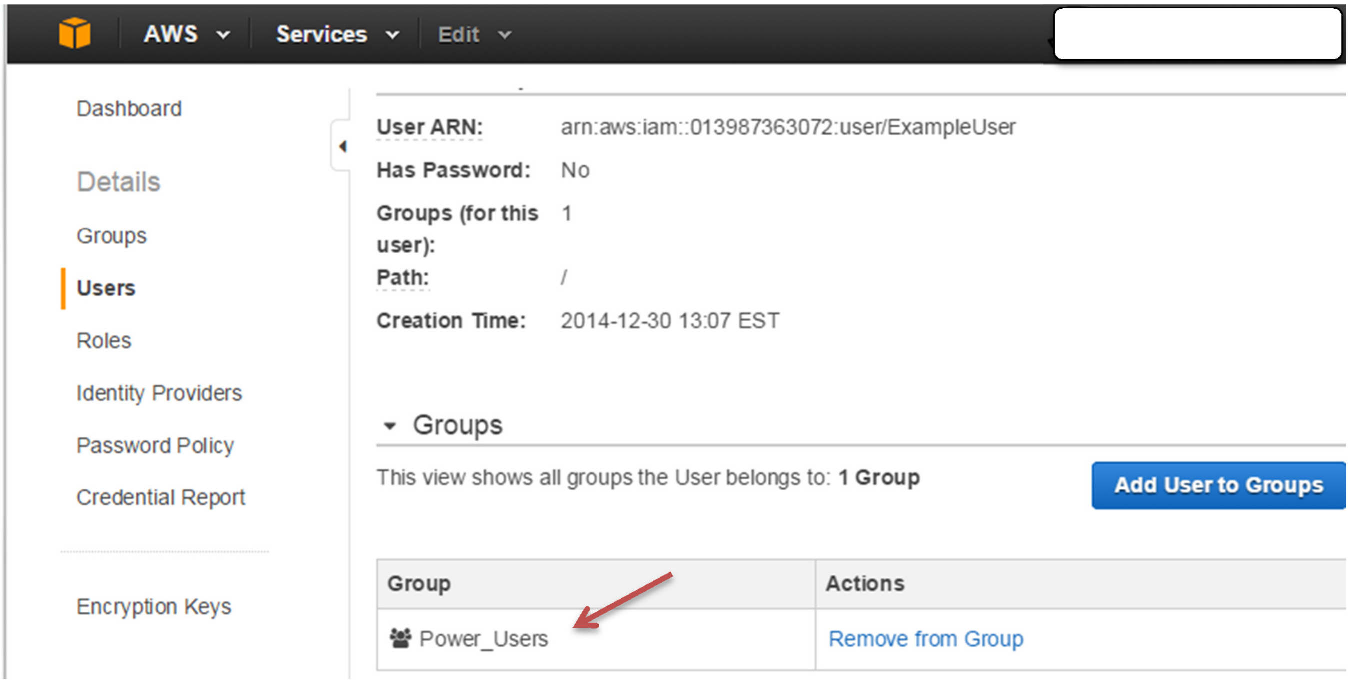
☐	Group Name	Users	Permissions Granted	Creation Time
☐	Admin	3	✓	2014-07-19 14:54 ...
☑	Power_Users	0	✓	2014-12-30 12:40 ...

Your new user and group have now been created.

Managing IAM User Permissions and Credentials

Now that you have created your first IAM user and group, lets take a look at the IAM user properties. Click on the **Users** option in the left-hand menu, then select the *ExampleUser* account that you just created.

Notice the user is a member of the **Power_Users** group that you added them to..



User ARN: arn:aws:iam::013987363072:user/ExampleUser

Has Password: No

Groups (for this user): 1

Path: /

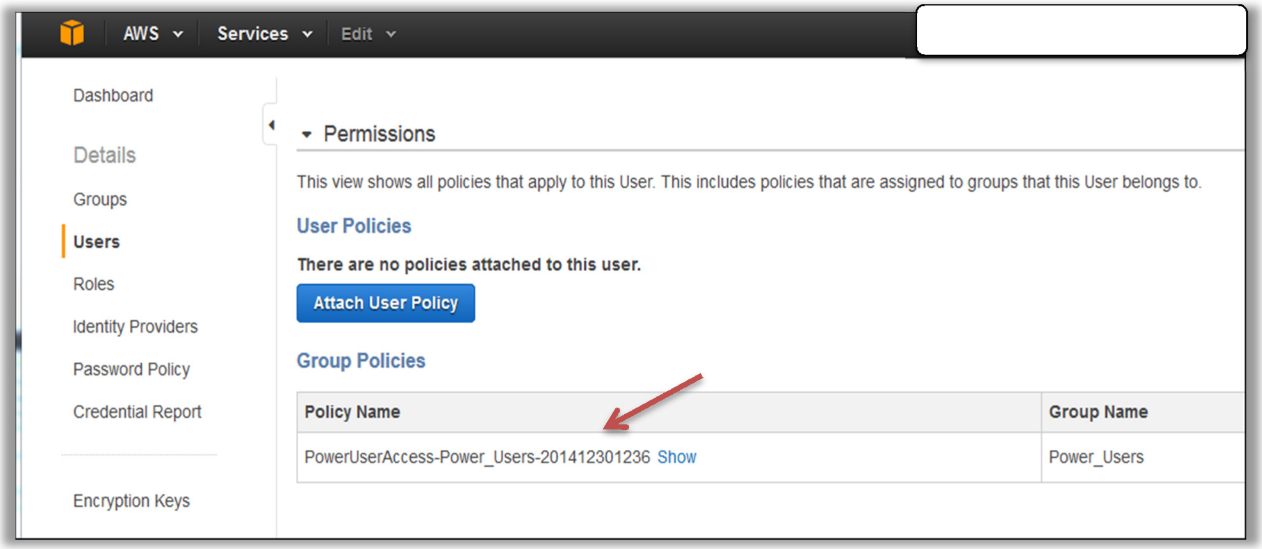
Creation Time: 2014-12-30 13:07 EST

Groups

This view shows all groups the User belongs to: **1 Group**

Group	Actions
Power_Users	Remove from Group

Now scroll down to **Permissions** to see the individual User and Group Policies that will be applied to this account. Note that this user only has the Power_Users group policy (that you just created) applied to the account.



Permissions

This view shows all policies that apply to this User. This includes policies that are assigned to groups that this User belongs to.

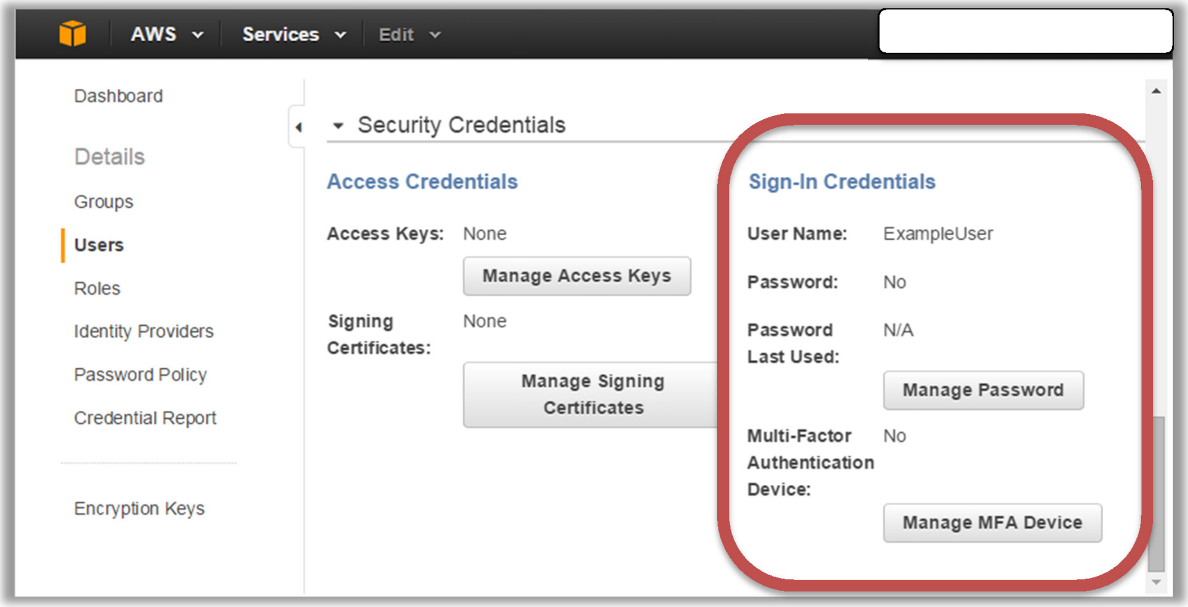
User Policies

There are no policies attached to this user.

Group Policies

Policy Name	Group Name
PowerUserAccess-Power_Users-201412301236 Show	Power_Users

Scroll down to **Security Credentials**. This is where you can assign or change a User's Console Password and Multi-Factor Authentication device.



Security Credentials

Access Credentials

Access Keys: None

Signing Certificates: None

Sign-In Credentials

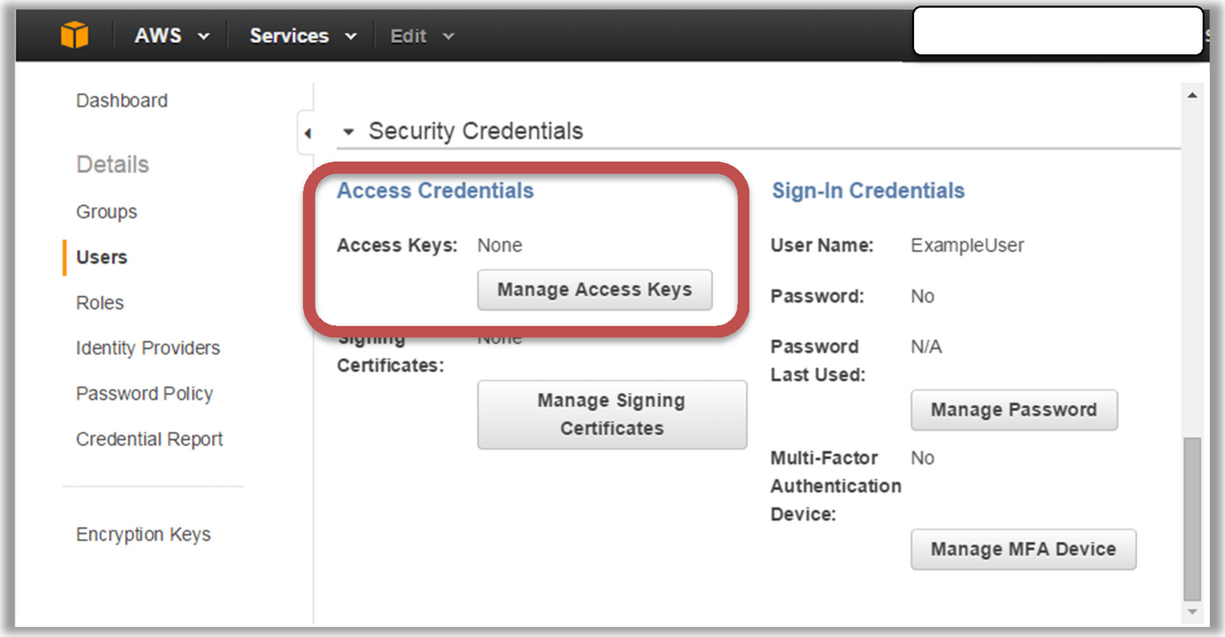
User Name: ExampleUser

Password: No

Password Last Used: N/A

Multi-Factor Authentication Device: No

And also Create, Rotate, or Revoke a user's API Access Keys (for using the AWS Command Line tools or other direct access to the AWS APIs through custom or 3rd party applications).



Access Keys: None

Signing Certificates: None

User Name: ExampleUser

Password: No

Password Last Used: N/A

Multi-Factor Authentication Device: No

IAM Roles for EC2

Applications or Command Line Tools running on Amazon Elastic Compute Cloud (Amazon EC2) instances that make requests to Amazon Web Services (AWS) must sign all AWS API requests with AWS access keys. AWS Identity and Access Management (IAM) Roles for EC2 instances, is a feature that makes it easier for your applications and command line tools to securely access AWS service APIs from EC2 instances. An IAM role with a set of permissions can be created and attached to an EC2 instance on launch. AWS access keys with the specified permissions will then be automatically made available on EC2 instances that have been launched with an IAM role. IAM roles for EC2 instances manages the muck of securely distributing and rotating your AWS access keys out to your EC2 instances so that you don’t have to.

Using IAM roles for instances, you can securely distribute AWS access keys to instances and define permissions that applications on those instances use when accessing other services in AWS. Here are some things you should know about using IAM roles for instances:

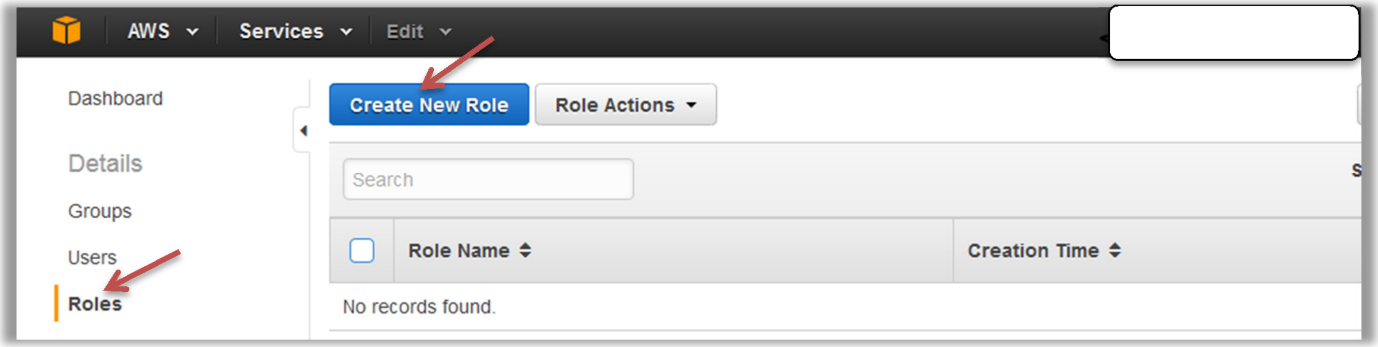
- AWS access keys for signing requests to other services in AWS are automatically made available on running instances.
- AWS access keys on an instance are rotated automatically multiple times a day. New access keys will be made available at least five minutes prior to the expiration of the old access keys.
- You can assign granular service permissions for applications running on an instance that make requests to other services in AWS.
- You can include an IAM role when you launch On-Demand, Spot, or Reserved Instances.
- IAM roles can be used with all Windows and Linux AMIs.

Warning

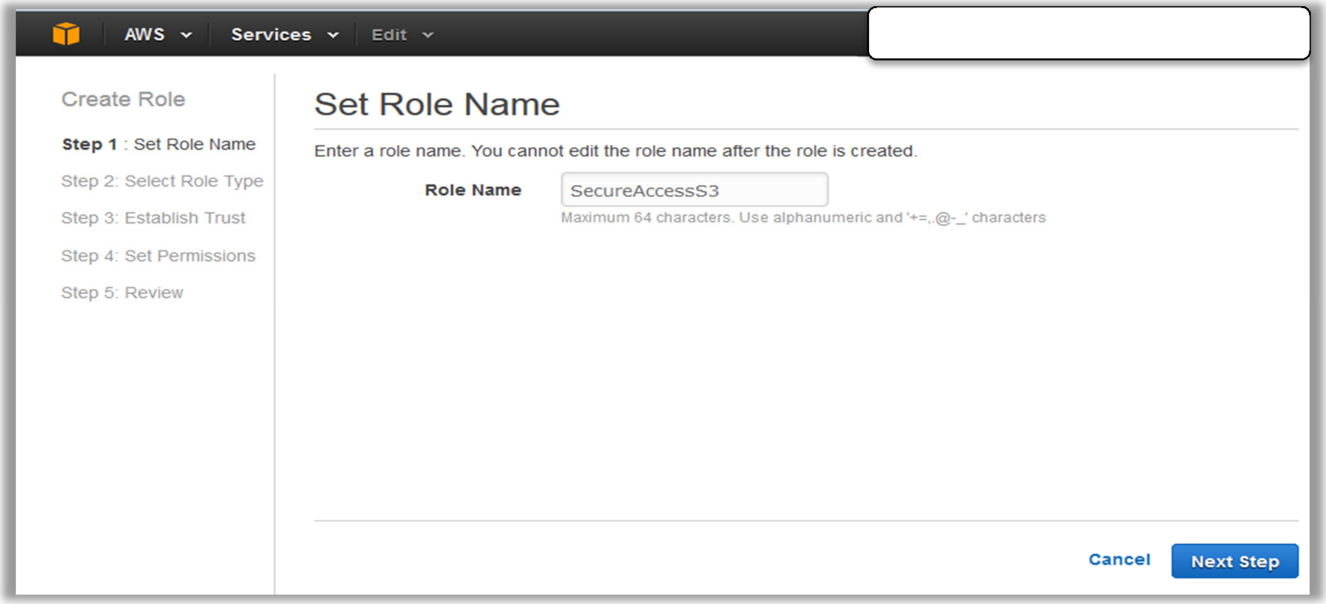
If you are using services that use instance metadata service (IMDS) with IAM roles, you should ensure that you do not expose your credentials when the services make HTTP calls on your behalf. You should either include logic to ensure that these services cannot leak information from IMDS, or you should have the appropriate firewall rules in place so that the services cannot access IMDS. Types of services that could expose your credentials include:

- HTTP proxies
- HTML/CSS validator services
- XML processors that support XML inclusion

To create an IAM Role for EC2, click on the **Roles** link on the left-hand menu and click **Create New Role**:



Create a new role called **SecureAccessS3** and click **Next Step**:



Set Role Name

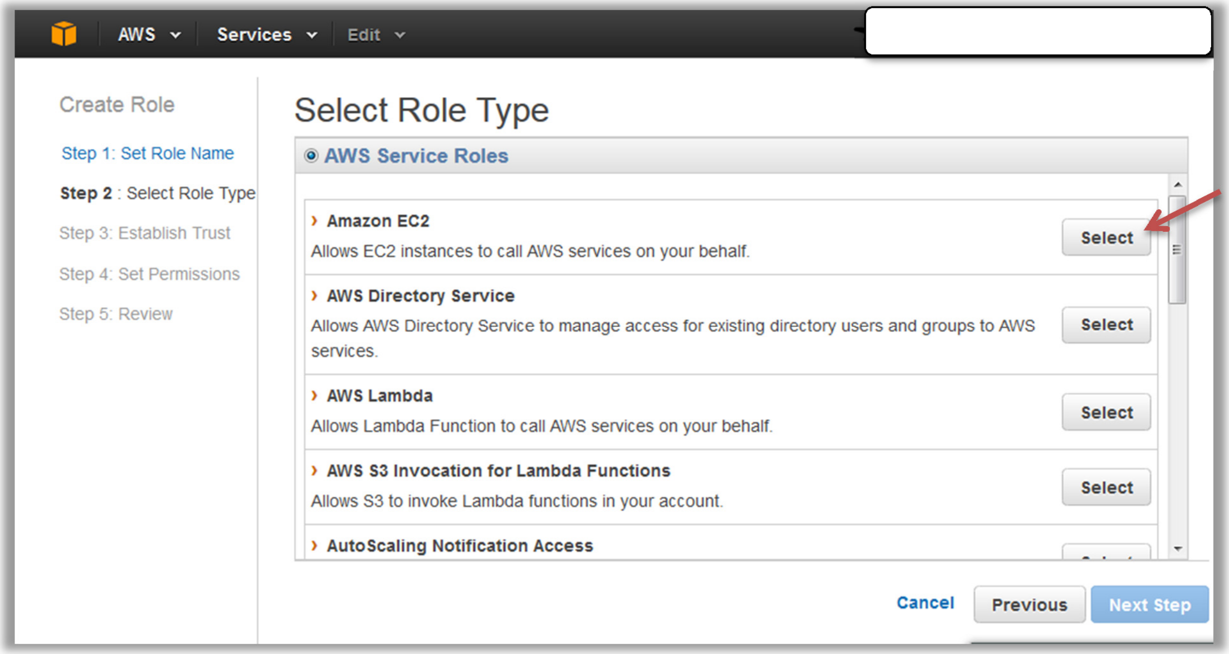
Enter a role name. You cannot edit the role name after the role is created.

Role Name: SecureAccessS3

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

Copyright 2015, Amazon Web Services, All Rights Reserved

IAM supports several different types of Roles – select the **Amazon EC2** Service Role for this example, but IAM roles can be used to grant access to AWS Services, other AWS Accounts, and 3rd Party Identity Providers.



Select Role Type

AWS Service Roles

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

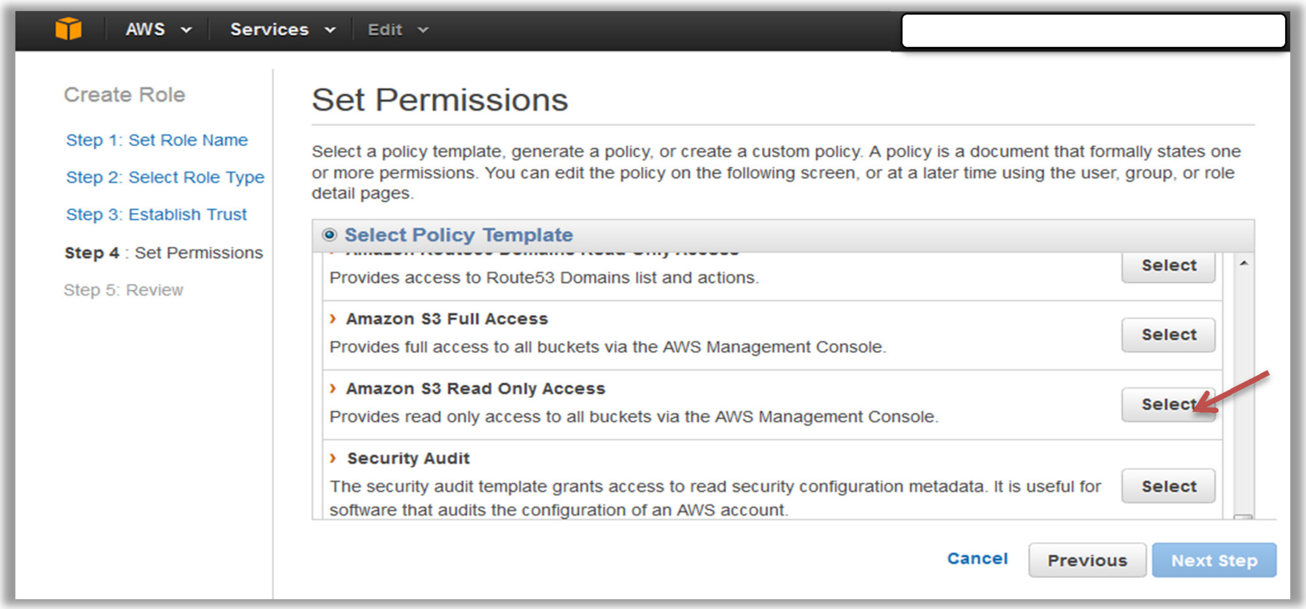
AWS Directory Service
Allows AWS Directory Service to manage access for existing directory users and groups to AWS services.

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

AWS S3 Invocation for Lambda Functions
Allows S3 to invoke Lambda functions in your account.

AutoScaling Notification Access

We now need to set permissions for this new role. Scroll down to find the **Amazon S3 Full Access** template and click **Select.**



Set Permissions

Select a policy template, generate a policy, or create a custom policy. A policy is a document that formally states one or more permissions. You can edit the policy on the following screen, or at a later time using the user, group, or role detail pages.

Select Policy Template

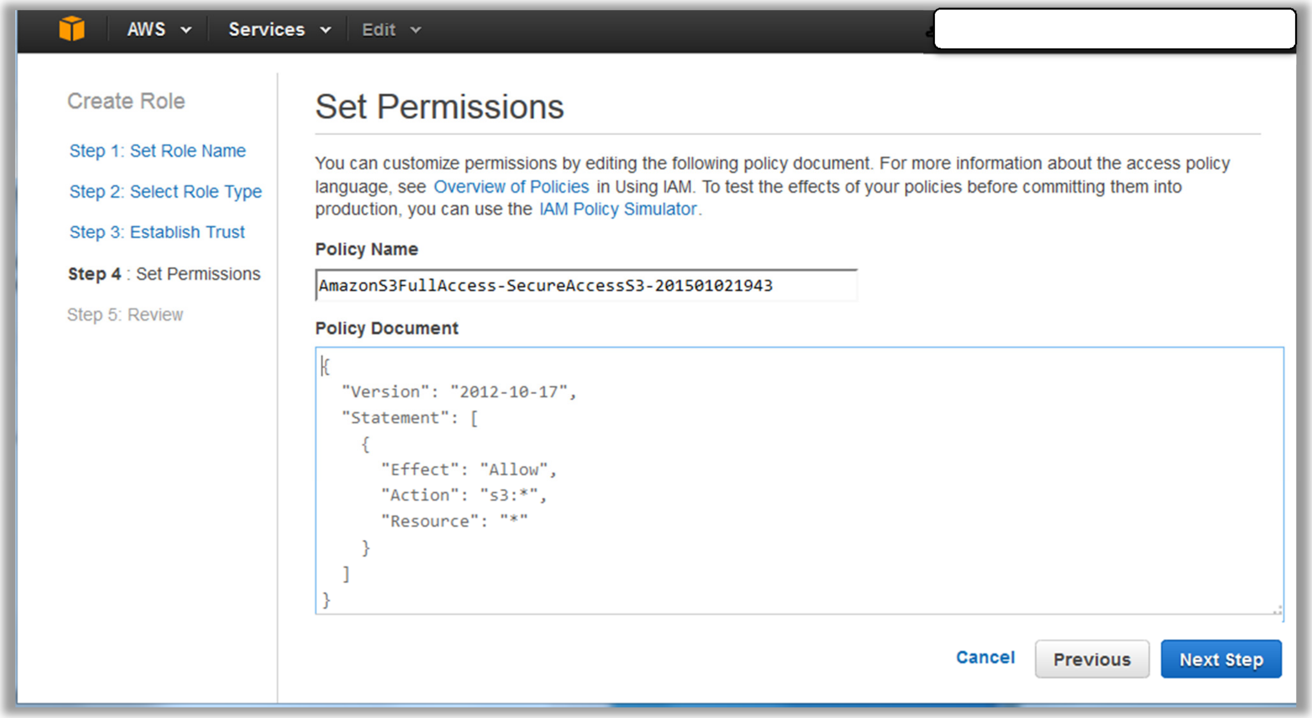
Provides access to Route53 Domains list and actions.

Amazon S3 Full Access
Provides full access to all buckets via the AWS Management Console.

Amazon S3 Read Only Access
Provides read only access to all buckets via the AWS Management Console.

Security Audit
The security audit template grants access to read security configuration metadata. It is useful for software that audits the configuration of an AWS account.

You now have the opportunity to edit the policy template. Following the principle of least priviledge, we would recommend that you further restrict S3 access to specific S3 buckets or resources. However for the purposes of this lab, click **Next Step** to accept the default policy.



Set Permissions

You can customize permissions by editing the following policy document. For more information about the access policy language, see Overview of Policies in Using IAM. To test the effects of your policies before committing them into production, you can use the IAM Policy Simulator.

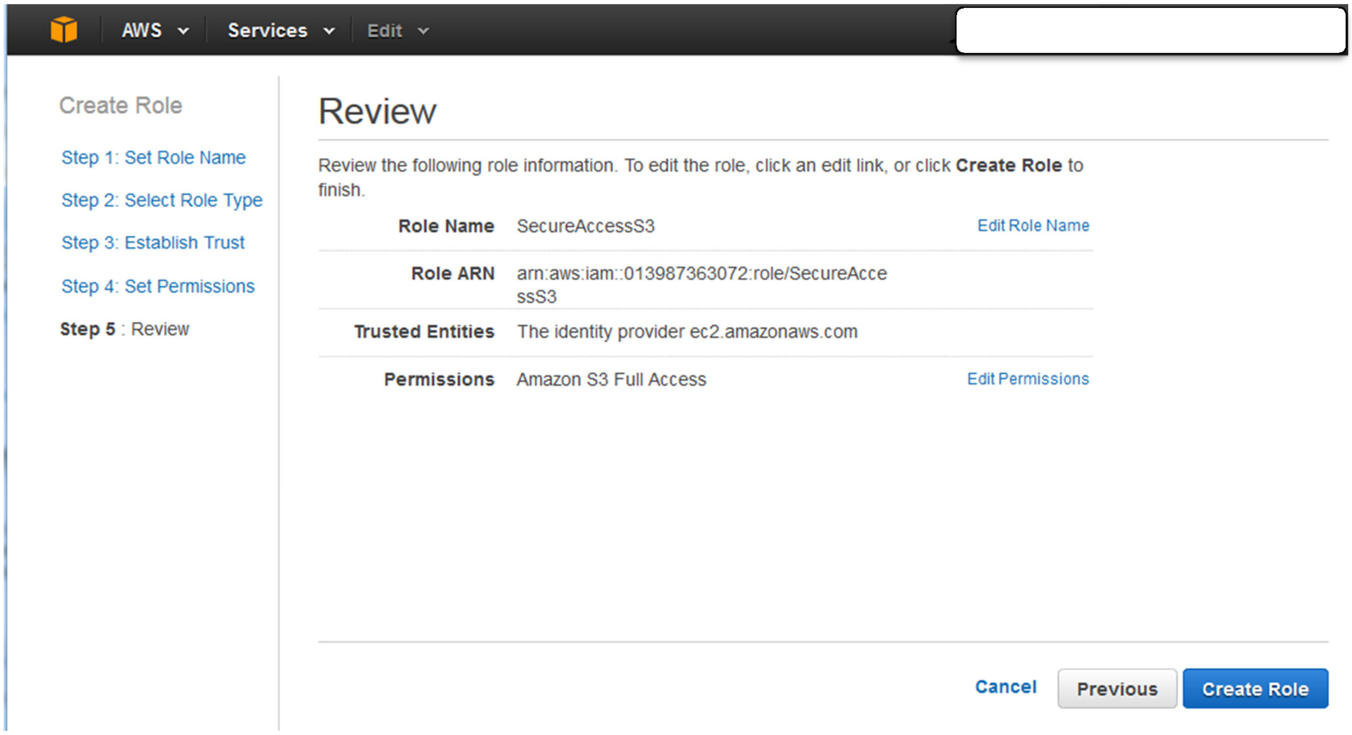
Policy Name

AmazonS3FullAccess-SecureAccessS3-201501021943

Policy Document

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "s3:*",
      "Resource": "*"
    }
  ]
}
```

Finally, click the **Create Role** button to create this role

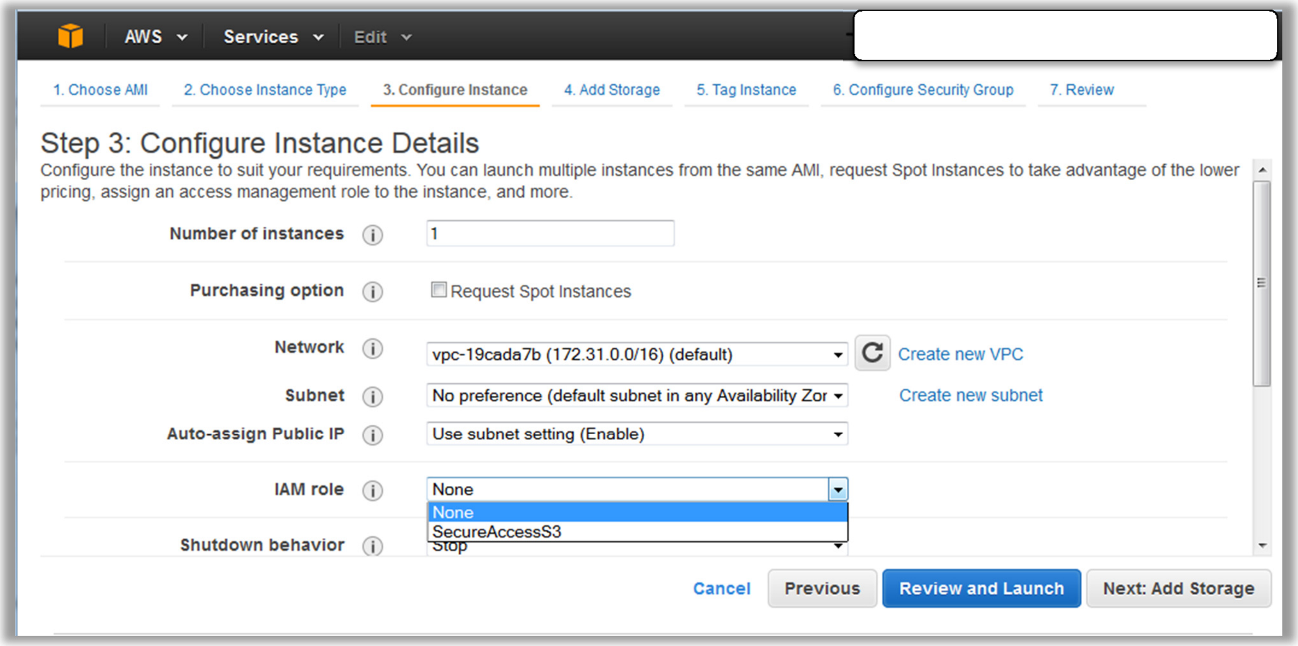


Review

Review the following role information. To edit the role, click an edit link, or click **Create Role** to finish.

Role Name	SecureAccessS3
Role ARN	arn:aws:iam::013987363072:role/SecureAccessS3
Trusted Entities	The identity provider ec2.amazonaws.com
Permissions	Amazon S3 Full Access

You can now use the newly created IAM Role when you launch an EC2 instance. For example, in the EC2 Console, you can select the role as part of launch process. Once the instance is launched, applications and tools that access AWS services will automatically pick up temporary credentials made available to the instance by the infrastructure.



Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot Instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances	1
Purchasing option	☐ Request Spot Instances
Network	vpc-19cada7b (172.31.0.0/16) (default)
Subnet	No preference (default subnet in any Availability Zor
Auto-assign Public IP	Use subnet setting (Enable)
IAM role	None None SecureAccessS3
Shutdown behavior	Stop

Congratulations! You have created your first IAM user, group, and role!

To learn more about IAM Roles for EC2 instances, please visit Working with Roles in the Using IAM guide and Using IAM roles with Amazon EC2 Instances in the Amazon EC2 User Guide.