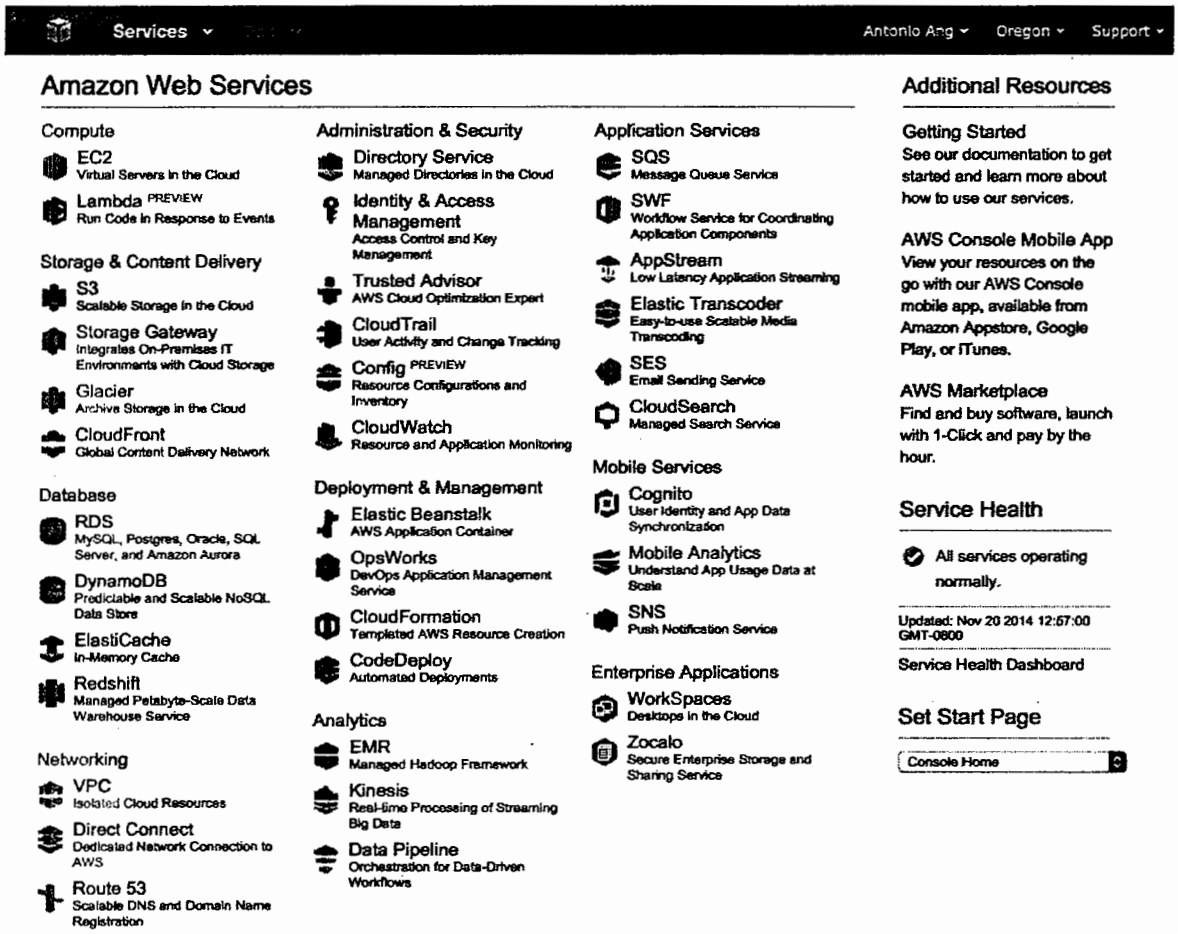
Lab 11

ADVANCED ROLES AND GROUPS MANAGEMENT USING IAM

STEP 1: Log In to the Amazon Web Service Console

This laboratory experience is about Amazon Web Services and you will use the AWS Management Console in order to complete all the lab steps.



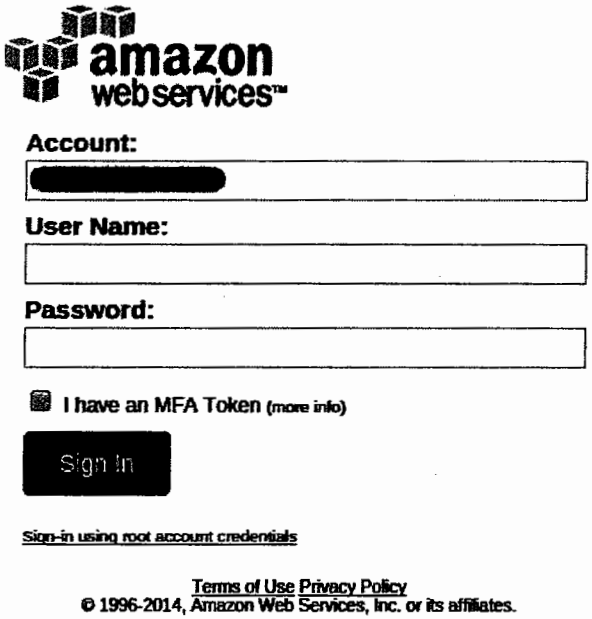
The AWS Management Console is a web control panel for managing all your AWS resources, from EC2 instances to SNS topics. The console enables cloud management for all aspects of the AWS account, including managing security credentials, or even setting up new IAM Users.

Log in to the AWS Management Console

In order to start the laboratory experience, open the Amazon Console by clicking this button:

Open AWS Console

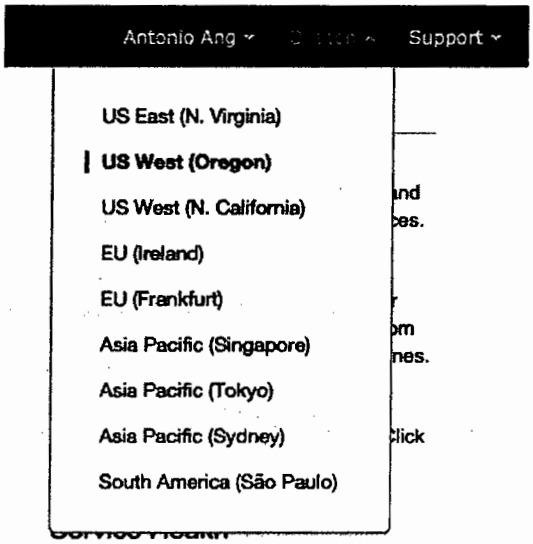
Log in with the username **xxxx** and the password **xxxx**



Select the right AWS Region

Amazon Web Services is available in different regions all over the world, and the console lets you provision resources across multiple regions. You usually choose a region that best suits your business needs to optimize your customer's experience, but you must use the region **US West (Oregon)** for this laboratory.

You can select the **US West (Oregon)** region using the upper right dropdown menu on the AWS Console page.

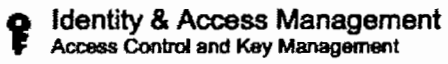


STEP 2: Create IAM User

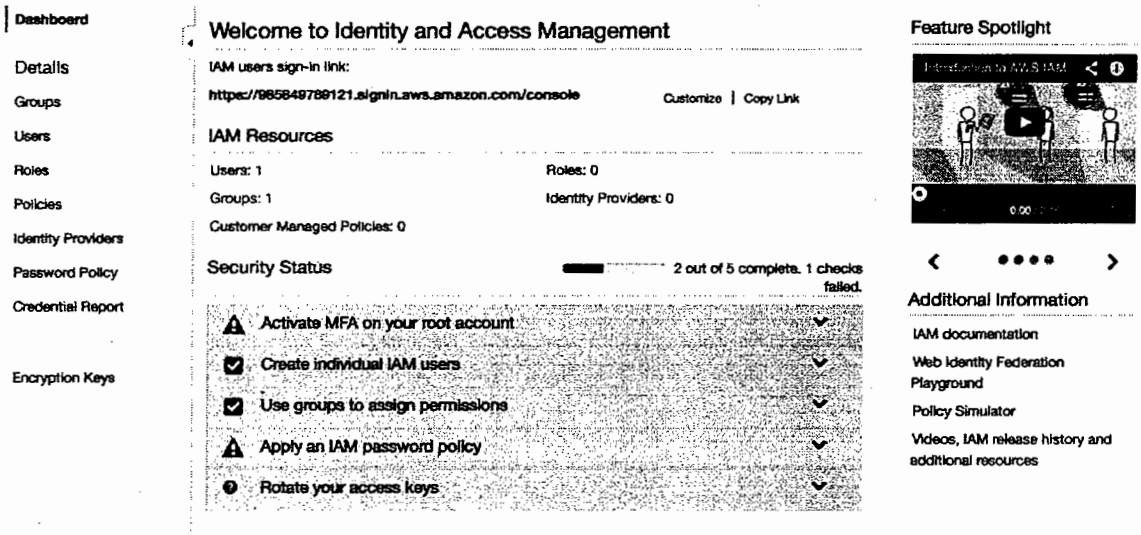
AWS Identity and Access Management (IAM) enables you to securely control access to AWS services and resources for your users. Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources.

You can create a new **IAM User** using the AWS Management Console.

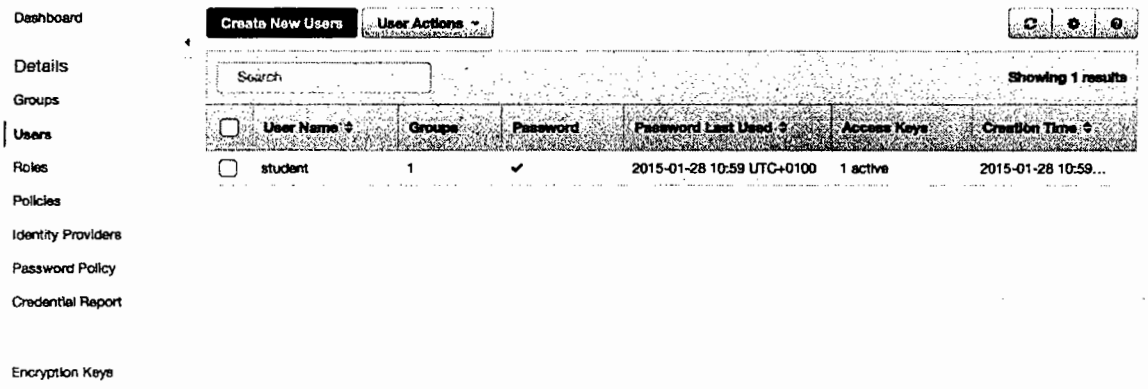
Select the **Identity & Access Management** (IAM) service from the Management Console dashboard:



From the IAM dashboard, click on **Users** link in the sidebar menu.

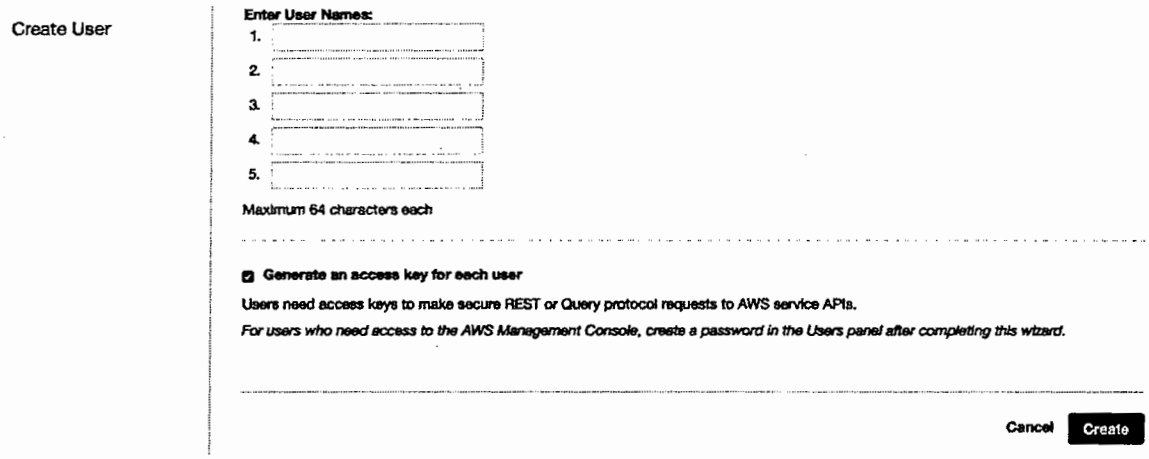


The **Users** page lists all available IAM Users, click on the **Create New Users** blue button for creating a new user.

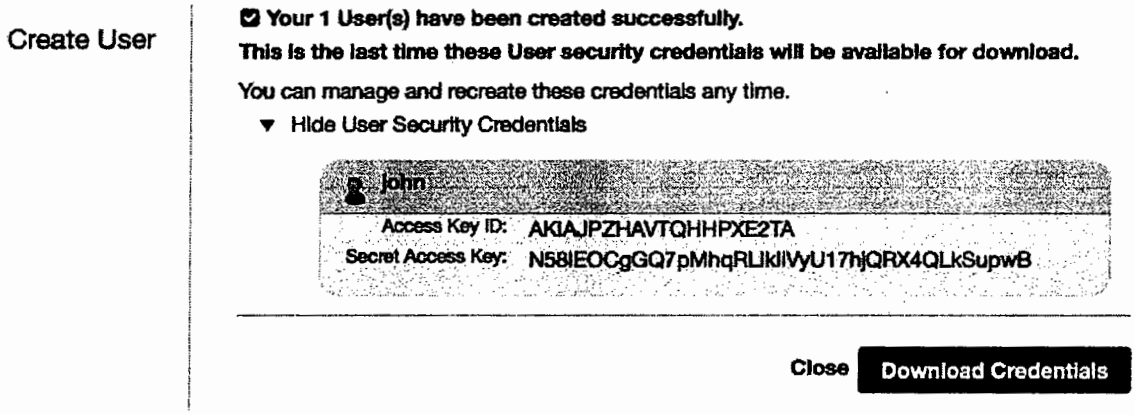


You can create up to 5 users at a time with usernames that don't exceed 64 characters.

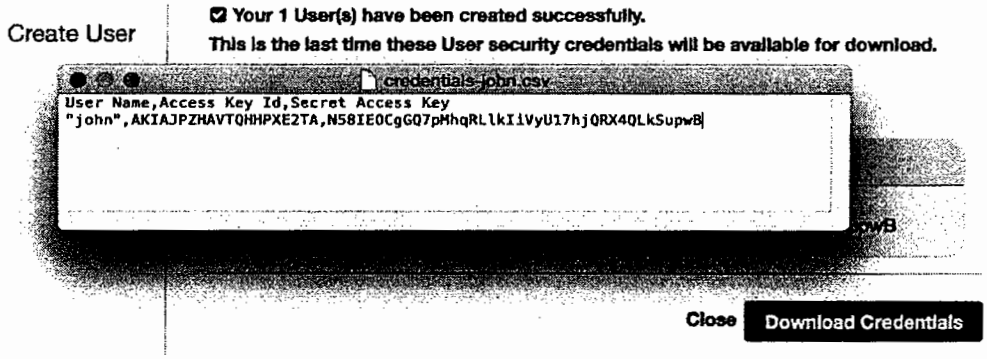
You need to enter the following username(s): **lab-user** and then click **Create**.



The AWS Management Console displays the list of all Access Key IDs and Secret Access Keys created for each user.

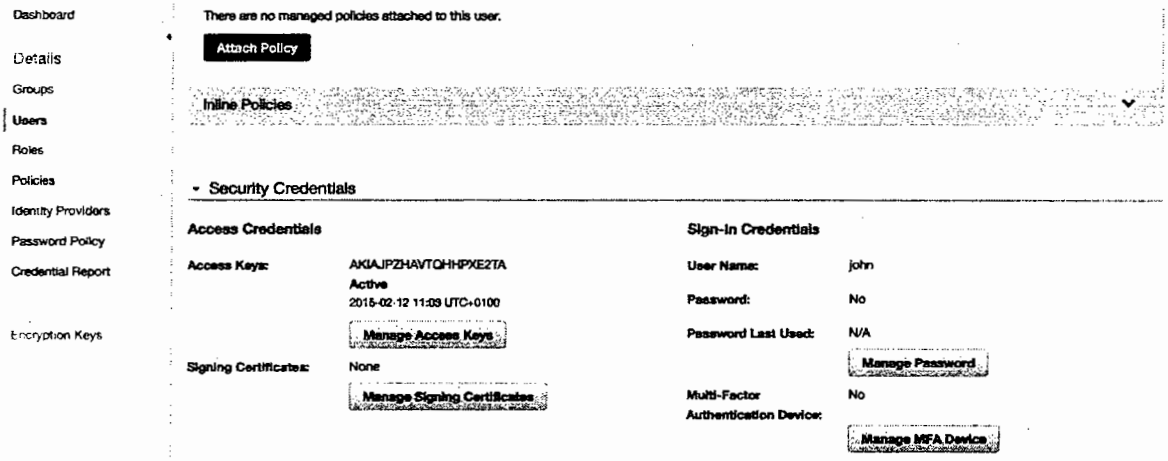


Click **Download Credentials** for downloading a csv file containing the security credentials (you won't be able to show them again if you click close).



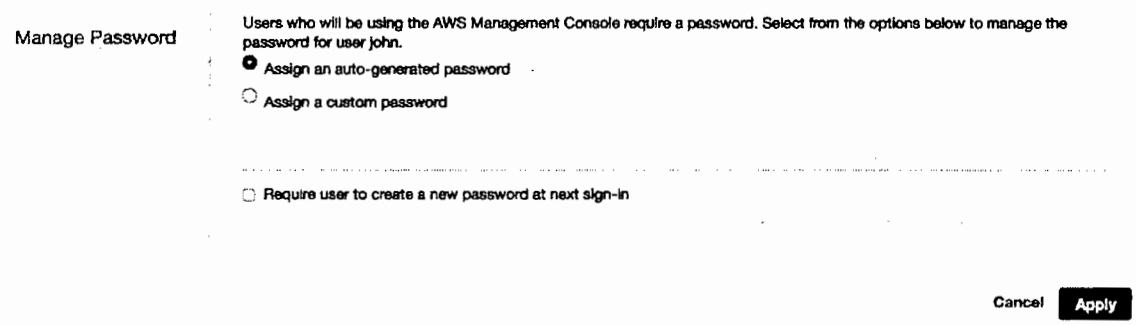
Generate a user password

Each created user comes without a password, so you cannot use it for logging into the AWS Management Console. You can generate a password for a specific user by opening the user details page and clicking on the **Manage Password** grey button.

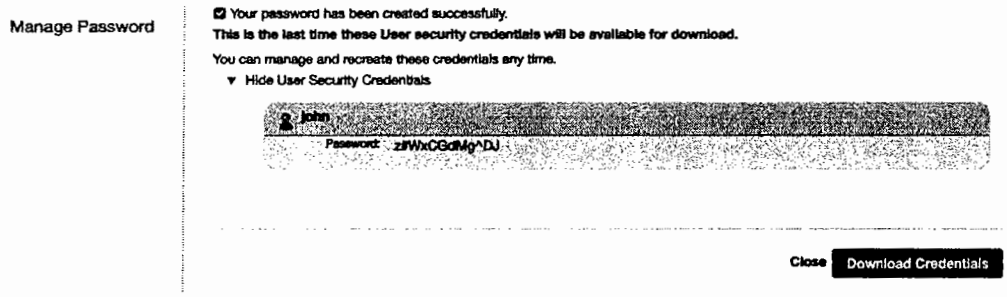


You can assign a specific password or let the system generate it for you. Forcing the user to create a new password at next sign-in usually is a good idea if you want to keep your user password secret.

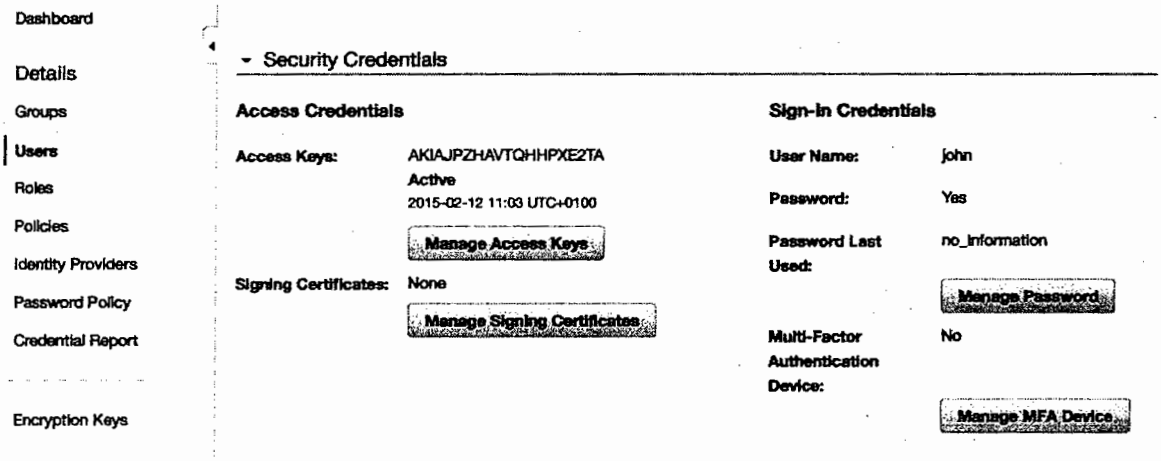
Click **Apply** for generating it.



The management console will show you the password once, save it in a safe place before closing the tab.



You user account is set up and you can use it for accessing the AWS Management Console.



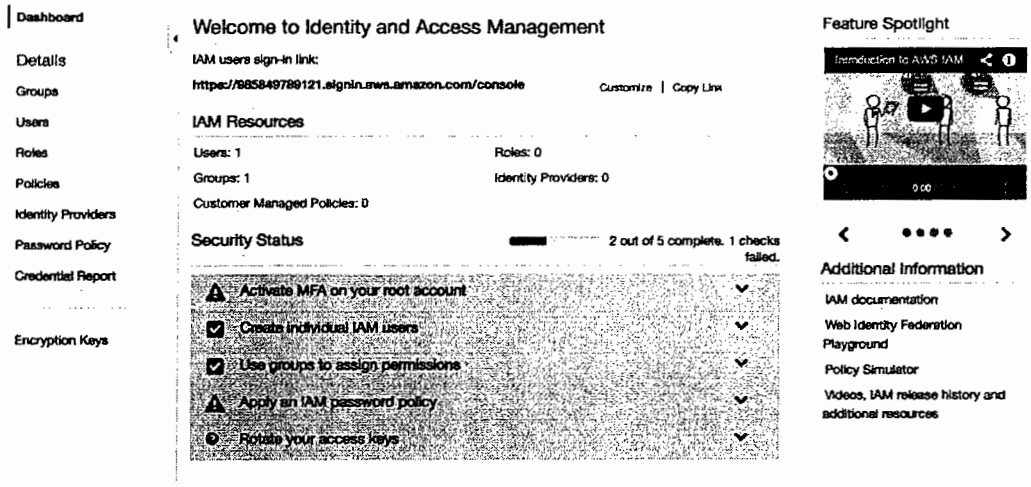
STEP 3: Create IAM Group

AWS Identity and Access Management (IAM) enables you to securely control access to AWS services and resources for your users. Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources.

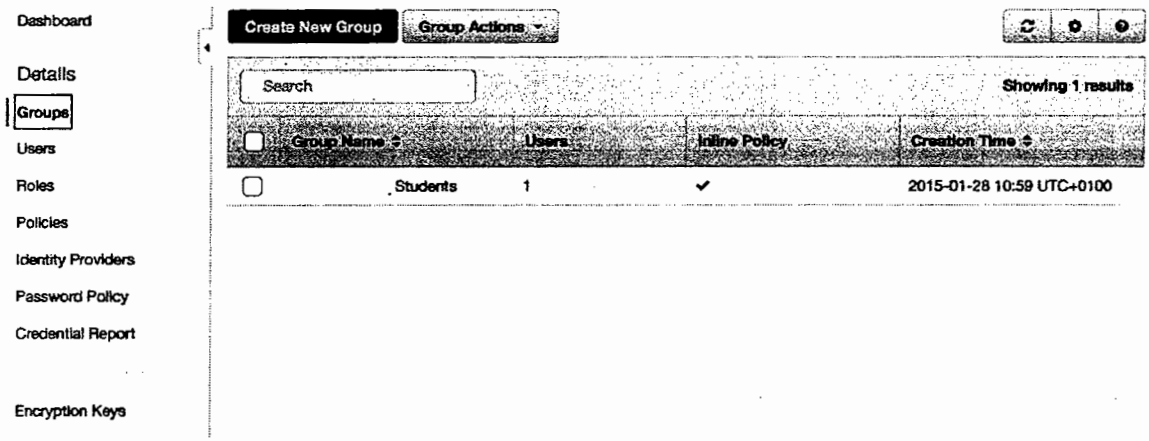
You can create a new **IAM Group** using the AWS Management Console.

Select the **Identity & Access Management** (IAM) service from the Management Console dashboard:

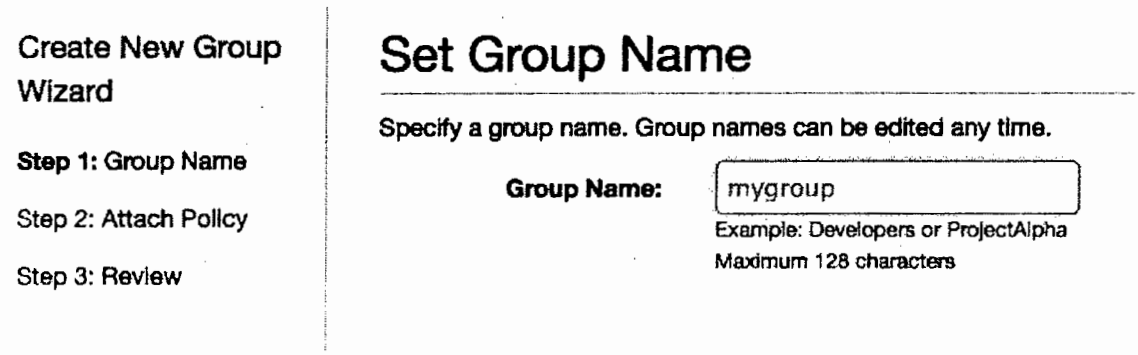
From the IAM dashboard, click on **Groups** link in the sidebar menu.



The **Groups** page lists all available IAM Groups, click on the **Create New Group** blue button for creating a new IAM group.



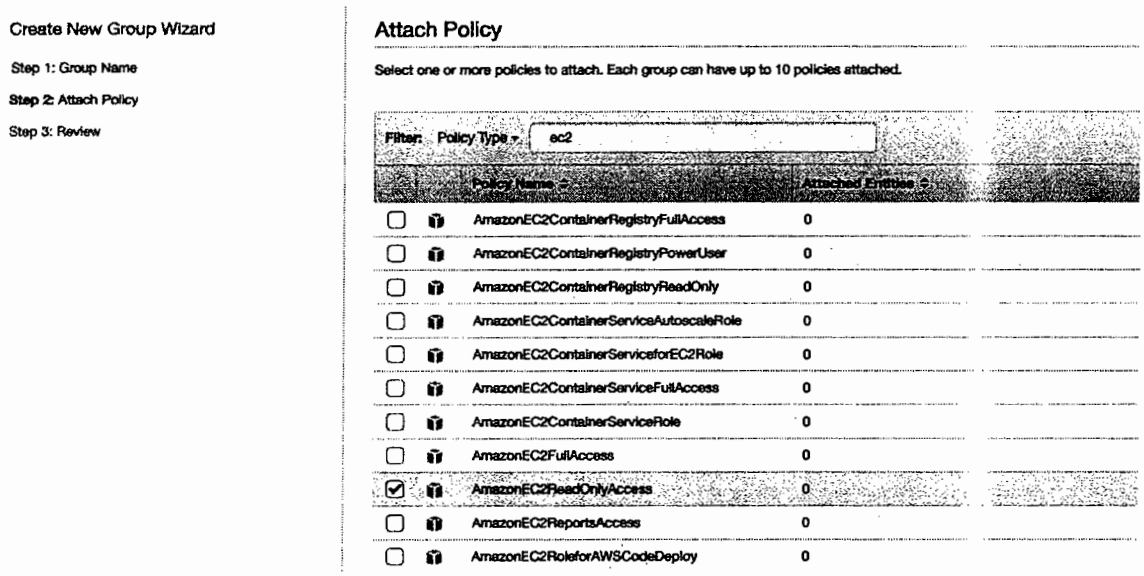
The **Create New Group** wizard is composed by 3 simple steps. You need to insert the name of the IAM group during the first step. Use **lab-ec2-viewers** as **Group Name** and then click the **Next Step** button.



You need to select one or more policies to attach to the group. They will be inherited by any user of the group.

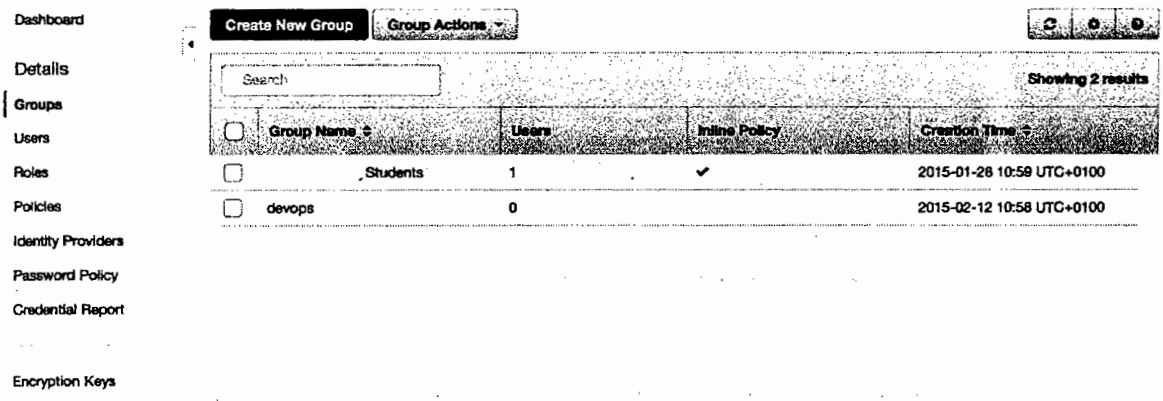
Please select the following policy(ies): **AmazonEC2ReadOnlyAccess**

Click **Next Step** to review your choices.



You are almost done, check all inserted data and the click **Create Group.**

The **Groups** page now lists the new group and you are able to assign the **lab-ec2-viewers** group to any available user.

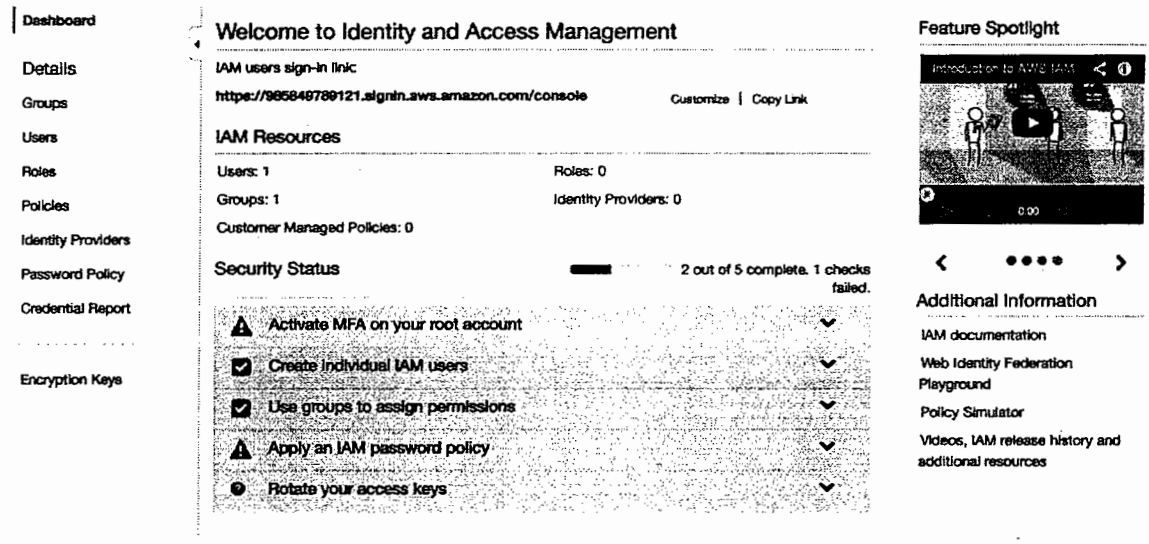


STEP 4: Add IAM User to Group

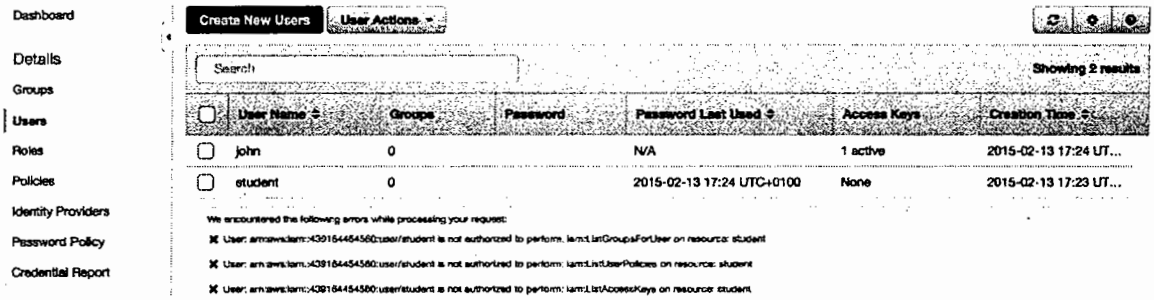
You can attach an **IAM User** to one or more IAM groups using the AWS Management Console.

Select the **Identity & Access Management** (IAM) service from the Management Console dashboard:

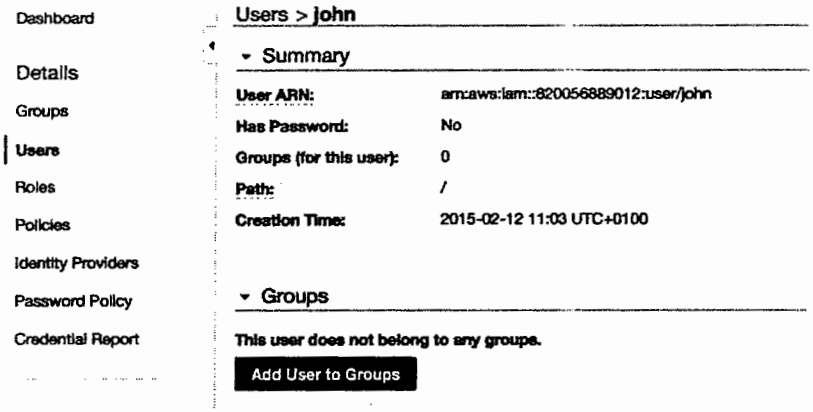
From the IAM dashboard, click on **Users** link in the sidebar menu.



The **Users** page lists all available IAM Users, click on the **lab-user** user for opening the details page.

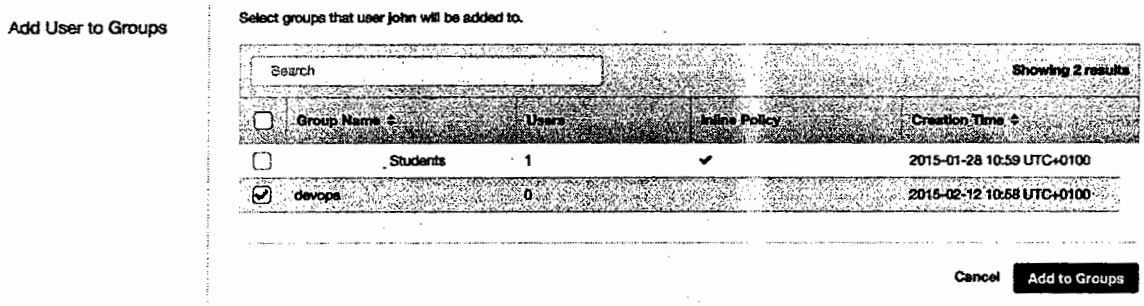


Click **Add User to Groups.**



You can add a single user to more than one group by selecting them one by one.

Select the following groups for completing this step: **lab-ec2-viewers**



Click **Add to Groups** and the user will be assigned to the selected groups.

STEP 5: Create customer managed policy with policy generator

You can create customer managed policies to define sets of permissions to attach to principal entities (users, groups, and roles) in your AWS account.

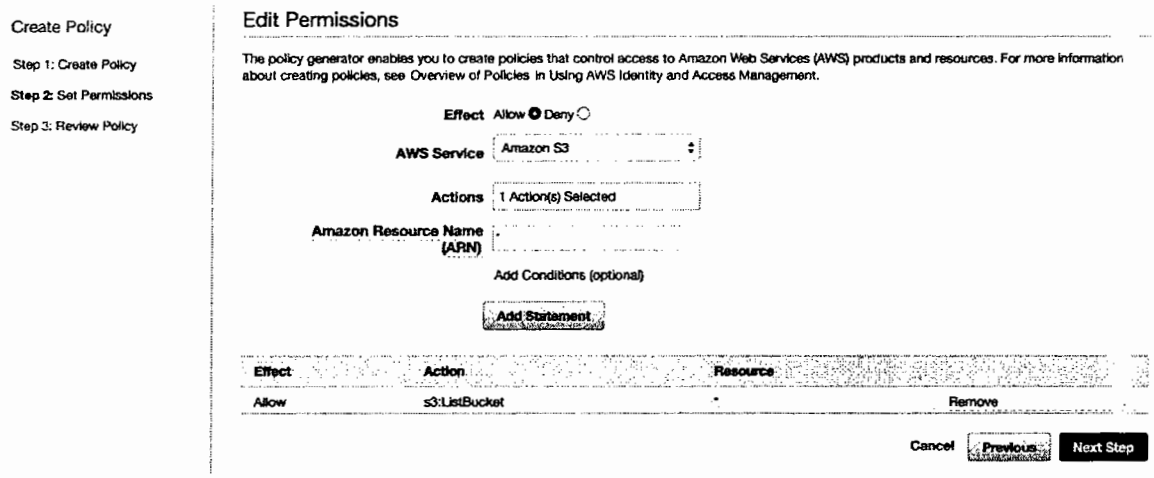
Select the **Identity & Access Management** (IAM) service from the Management Console dashboard:

In the navigation pane, choose Policies, and then choose **Create Policy**.

1. Choose the Select button that corresponds to the Policy Generator for build the policy using Policy Generator tool

   - ✓ On **Effect** Select Allow
   - ✓ On of **AWS Service** select **Amazon S3**

- ✓ On **Actions** select **ListBucket** item
- ✓ On **Amazon Resource Name ARN** specify ***** Then choose **Add Statement**



2. Select **Next Steps** and in Policy Name filed insert **lab-s3-policy**
3. After you complete your changes, choose Validate Policy and ensure that no errors display in a red box at the top of the screen. Correct any errors that are reported.
4. Choose Create Policy to save your new policy.

STEP 6: Attach policy to Users

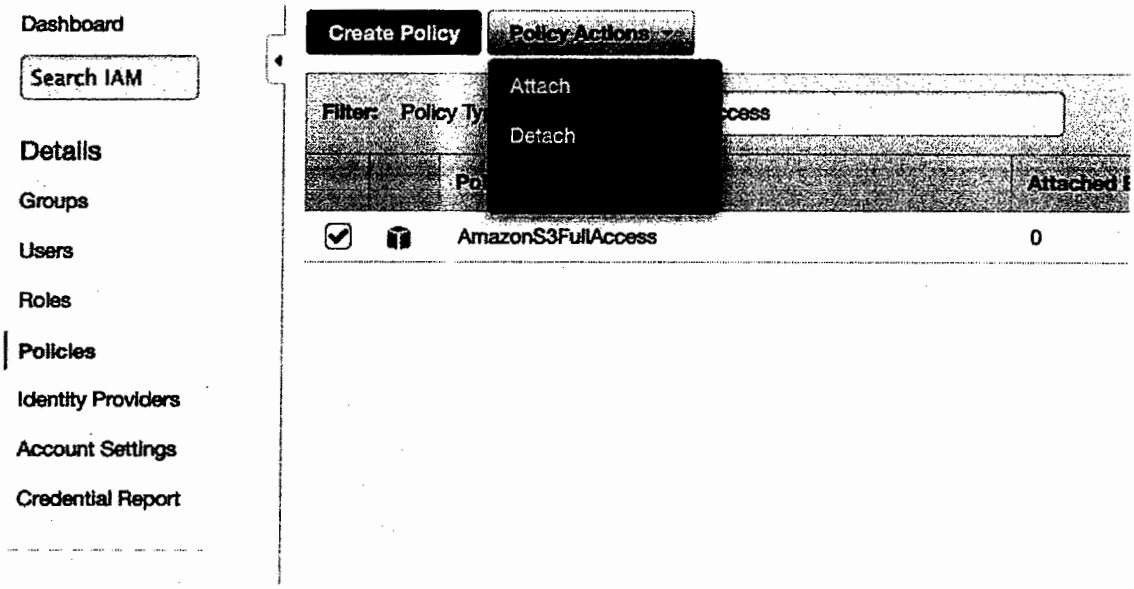
AWS Identity and Access Management (IAM) enables you to securely control access to AWS services and resources for your users. Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources.

You can create customer managed policies to define sets of permissions to attach to principal entities (users, groups, and roles) in your AWS account.

Select the **Identity & Access Management** (IAM) service from the Management Console dashboard:

In the navigation pane, choose Policies.

1. In the list of policies, select the check box next to the name of the policy to attach. Use the Filter menu and the Search box to filter the list of policies with name **AmazonS3ReadOnlyAccess**.
2. Choose Policy Actions, and then choose Attach.
3. Select the **lab-user** User to attach the policy.
4. After selecting the User, choose **Attach Policy**.



Now the User **lab-user** has the attached policy and relative privileges

STEP 7: Create IAM Role

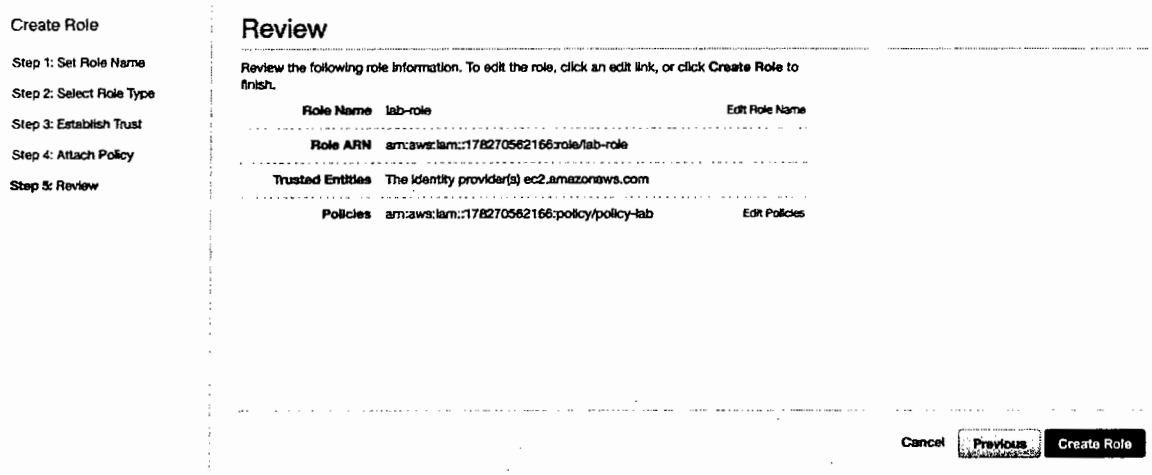
You can create a new **IAM Role** using the AWS Management Console.

Select the **Identity & Access Management** (IAM) service from the Management Console dashboard:

From the IAM dashboard, click on the **Roles** link in the sidebar menu then:

1. Click Create **New Role**
2. For Role name type the role name **lab-role**

3. On the Select Role Type page, select AWS Services Roles, then search for **Amazon EC2**, and click on **Select**
4. On the Attach Policy page, filter for the name **S3Full** and select the policy **AmazonS3FullAccess**
5. Click Next Step to review the role and copy your **Role ARN**
6. Then click **Create Role**



STEP 8: Launch EC2 Instances with IAM Profile

You can launch an EC2 instance using the EC2 launch wizard.

Select the EC2 service from the Management Console dashboard:

From the dashboard, click **Launch Instance**.

EC2 Dashboard
Events
Tags
Reports
Limits
INSTANCES
Instances
Spot Requests
Reserved Instances
IMAGES
AMIs
Bundle Tasks

Resources

You are using the following Amazon EC2 resources in the US West (Oregon) region:

0 Running Instances	1 Elastic IPs
0 Volumes	0 Snapshots
0 Key Pairs	0 Load Balancers
0 Placement Groups	2 Security Groups

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Launch Instance

Note: Your instances will launch in the US West (Oregon) region

The **Select an Amazon Machine Image (AMI)** page displays a list of basic configurations called **Amazon Machine Images (AMIs)** that serve as templates for your instance. Select the 64-bit **Amazon Linux AMI**.

1. Choose AMI | 2. Choose Instance Type | 3. Configure Instance | 4. Add Storage | 5. Tag Instance | 6. Configure Security Group | 7. Review

Step 1: Choose an Amazon Machine Image (AMI)

Cancel and Exit

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Quick Start | 1 to 22 of 22 AMIs

My AMIs
AWS Marketplace
Community AMIs
Free tier only

Amazon Linux AMI 2014.09.1 (HVM) - ami-b5a7ea85
Amazon Linux — Free tier eligible
The Amazon Linux AMI is an EBS backed image. It includes the 3.14 kernel, Ruby 2.1, PHP 5.5, PostgreSQL 9.3, Docker 1.2, the AWS command line tools, and repository access to many other packages.
Root device type: ebs Virtualization type: hvm
Select — 64-bit

Red Hat Enterprise Linux 7.0 (HVM), SSD Volume Type - ami-99bef1a9
Red Hat — Free tier eligible
Red Hat Enterprise Linux version 7.0 (HVM), EBS General Purpose (SSD) Volume Type
Root device type: ebs Virtualization type: hvm
Select — 64-bit

On the **Select an Instance Type** page, do not change any option and click on **Next, Configure Instance Details.**

1. Choose AMI | 2. Choose Instance Type | 3. Configure Instance | 4. Add Storage | 5. Tag Instance | 6. Configure Security Group | 7. Review

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

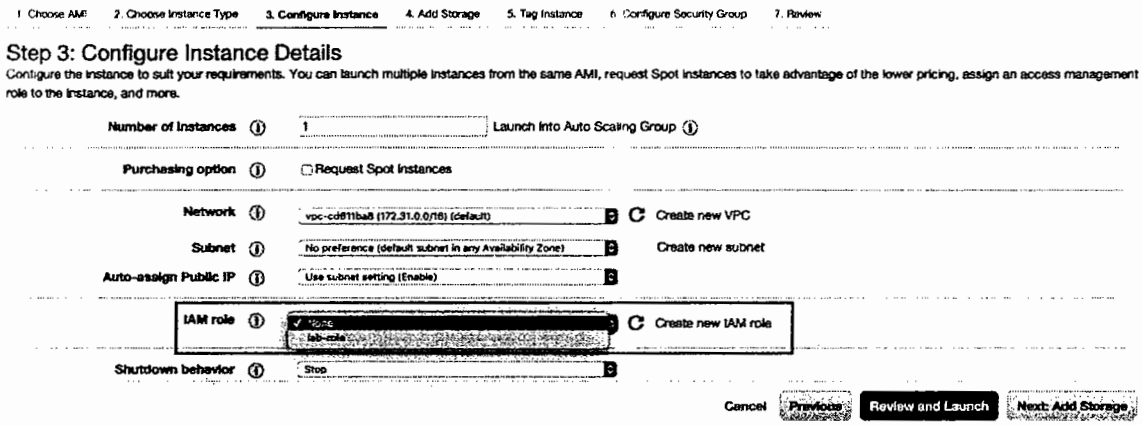
Filter by: All instance types | Current generation | Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
■	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate
	General purpose	t2.small	1	2	EBS only	-	Low to Moderate

Cancel | Previous | Review and Launch | Next: Configure Instance Details

On the **3. Configure Instance** tab,
select **Network** **172.31.0.0/16** and **Subnet** **172.31.16.0/24** make sure to select **IAM Role** **lab-role** and then click **Next, Add Storage**.



On the **4. Add Storage** tab, do not change any option and click "**Review and Launch**" button.

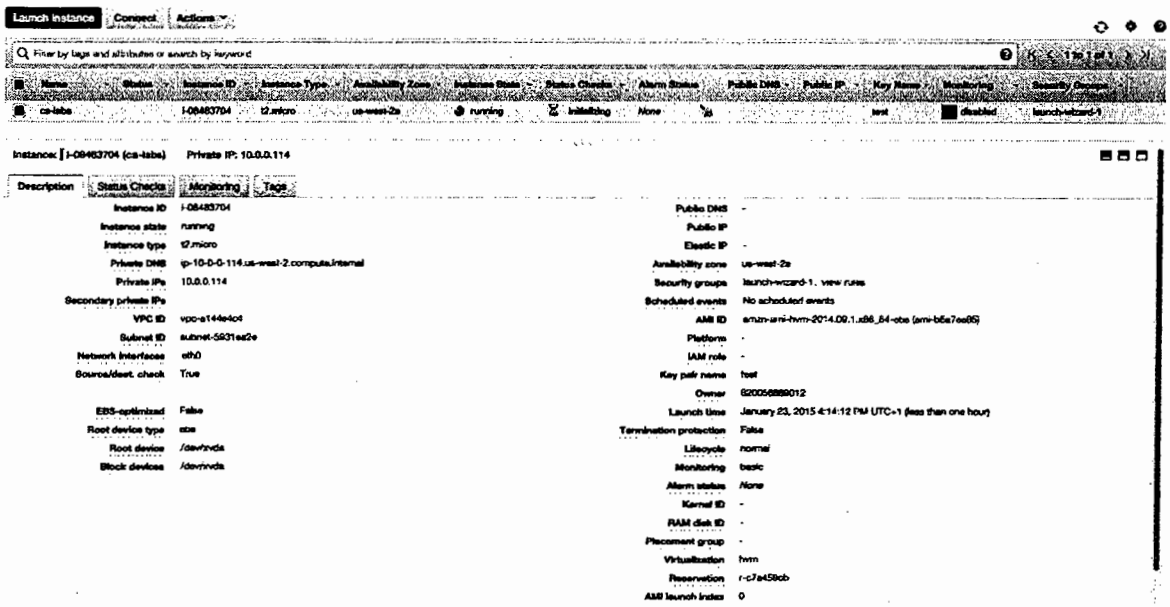
On the Review Instance Launch page, click **Launch**.

In the **Select an existing key pair or create a new key pair** dialog box, select **Create a new key pair**, then choose a KeyPair name and download it.

Select the acknowledgment check box, and then click **Launch Instances**.

A confirmation page will let you know that your instance is launching. Click **View Instances** to close the confirmation page and return to the console.

On the Instances screen, you can view the status of your instance. It will take a short time for your instance to be launched. When you launch an instance, its initial state is **pending**. After the instance starts, its state changes to **running**, and it receives a public DNS name.



STEP 9: Connect to a remote shell using an SSH connection

In order to manage a remote Linux server, you must employ an **SSH Client**. Secure Shell (SSH) is a cryptographic network protocol for securing data communication. It establishes a secure channel over an insecure network. Common applications include remote command-line login and remote command execution.

Connect using Linux / Mac OS

Linux distributions and Mac OS are shipped with a fully working SSH client that accepts standard PEM Keys.

Starting a remote SSH session is easy:

- ✓ Open your **Terminal** application
- ✓ Write and run the following command: **ssh -i /path/to/your/keypair.pem user@server-ip** .

server-ip is the Public IP of your server, you can find it in the EC2 instance details

user is the remote system user that will be used for the remote authentication

Amazon Linux AMIs typically use **ec2-user** as username.

Ubuntu AMIs login user is **ubuntu** , Debian AMIs use **admin** instead.

Assuming that you selected the Amazon Linux AMI, your assigned public IP is 123.123.123.123, and your keypair (named "keypair.pem") is stored in /home/youruser/keypair.pem, the right command to run is: **ssh -i /home/youruser/keypair.pem ec2-user@123.123.123.123**

Note: your SSH Client may refuse to start the connection, warning that the key file is unprotected. You should deny the file access to any other system user by changing its permissions. Issue the following command and then try again:

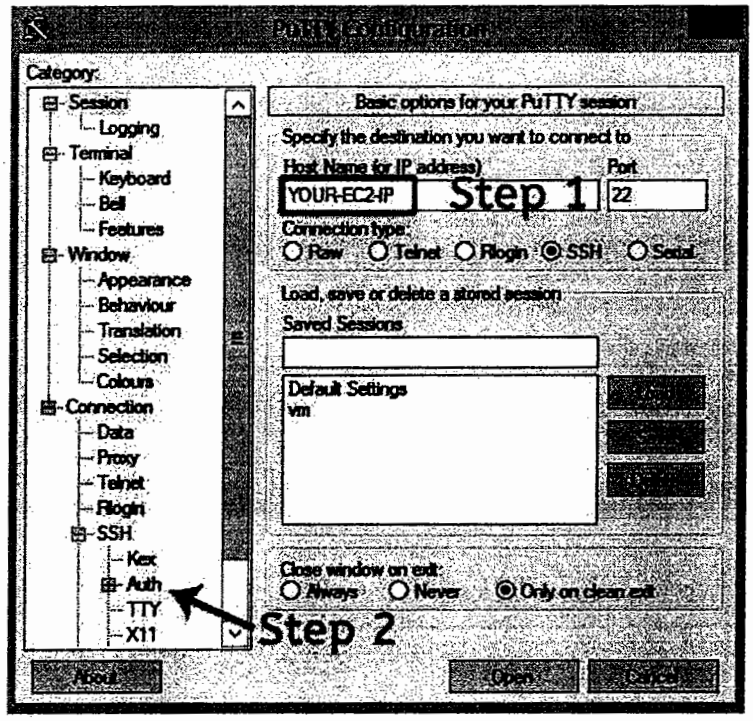
chmod 600 /home/youruser/keypair.pem

Connect using Windows

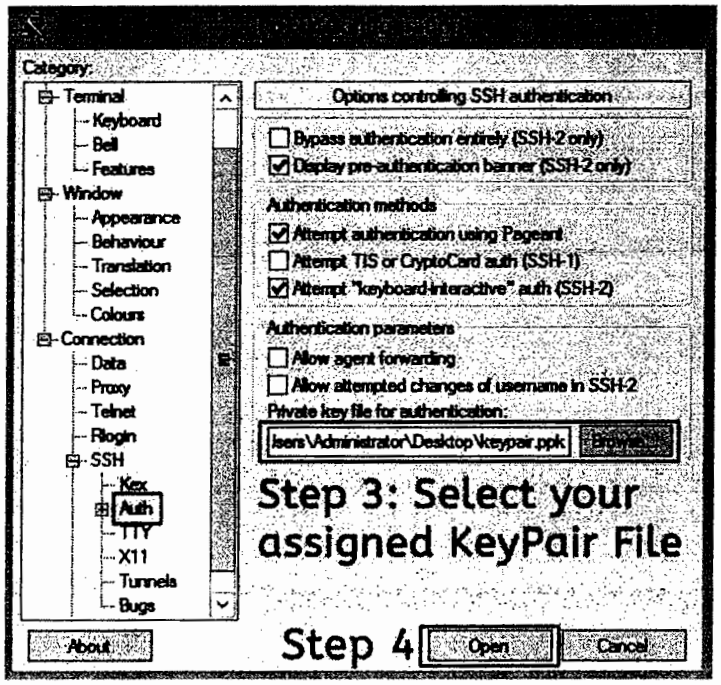
Windows has no SSH client, so you must use PuTTY and convert the PEM key to PPK using PuTTYgen.

Starting a remote SSH session using PuTTY is easy:

- ✓ Open PuTTY and insert the EC2 instance IP Address in the Host Name field.



✓ Select **Connection** > **SSH** > **Auth** section and then select the downloaded Keypair that you previously converted to PPK format.



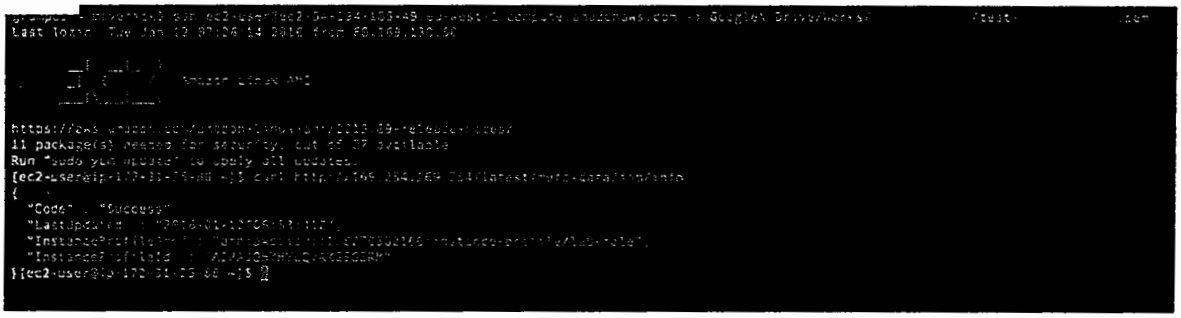
✓ After some seconds, you will see the authentication form. **Login as** **ec2-user** and you will see the EC2 server welcome banner.

STEP 10: Test IAM Profile from EC2 Linux instance

Instead of creating and distributing your AWS credentials, you can delegate permission to make API requests using IAM roles and specifing the role when you launch your instances

When are you logged on EC2 Instances you can use this command for retrieve the current IAM Profile

curl http://169.254.169.254/latest/meta-data/iam/info

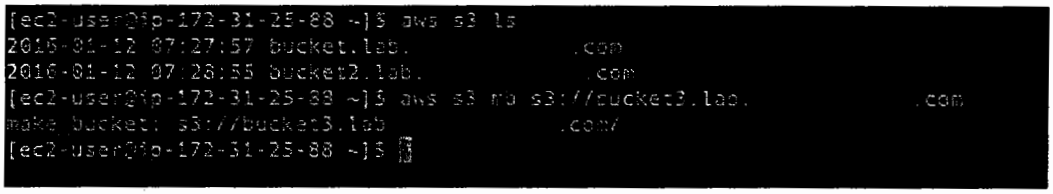


The key **InstanceProfileArn** identifie the ARN of role associated to EC2 instance

Now you can test the IAM Profile using AWS Cli for list bucket and create bucket without specify AK and SK. Rememeber that role **lab-role** has policy **AmazonS3FullAccess**

aws s3 ls for list bucket

aws s3 mb s3://bucket-name for make a bucket



Now you can call the AWS API specifying the AK and SK on command line. This overwrite the IAM Profile and you will be able to interact with EC2 service in read only mode because the policy associated to Group **lab-ec2-viewers** is the policy **AmazonEC2ReadOnlyAccess** and able to interact with S3 service in read only mode because the policy attached to user is **AmazonS3ReadOnlyAccess**

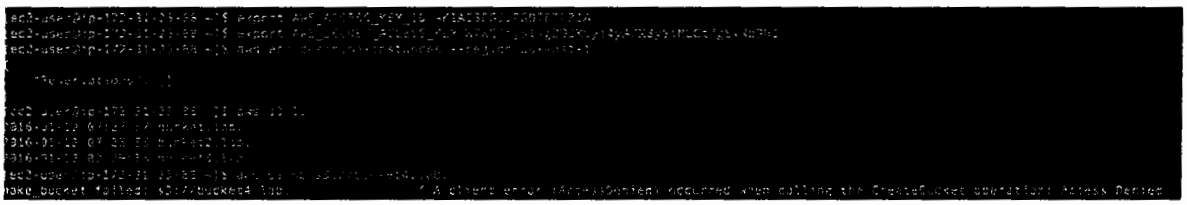
export AWS_ACCESS_KEY_ID=access-key-of-user-created-before

export AWS_SECRET_ACCESS_KEY=secret-key-of-user-created-before

now you can use cli

aws ec2 describe-instances --region us-east-2 for list regions

aws s3 ls for list bucket



How you can see you have the privileges for describe instances and list bucket, but you not have the privileges for make bucket.