

Step 1

Access the AWS console through the following link:

<https://450006219561.signin.aws.amazon.com/console>

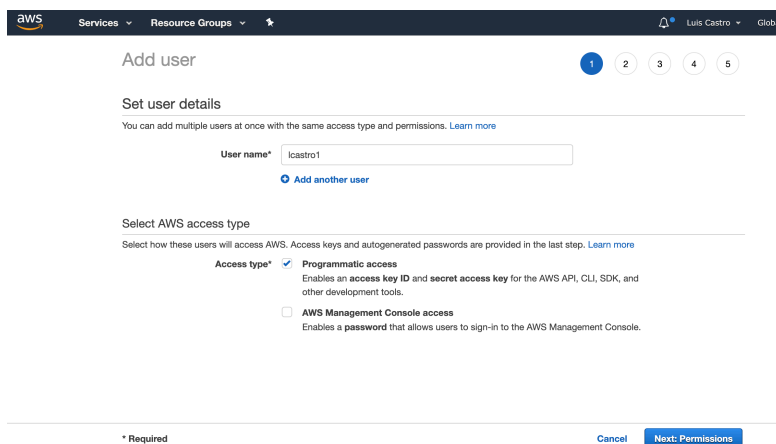
Step 2

Enter the Identity Access Management service and choose Users> Create New Users

Create a user with the same username by adding the number 1

- Ex: lcastro1

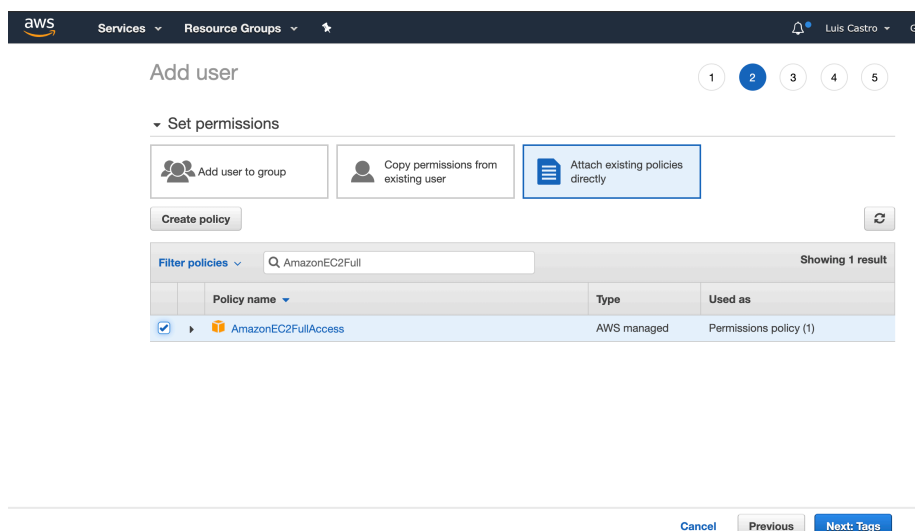
Check Programmatic Access



The screenshot shows the AWS IAM 'Add user' console. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and a user profile for 'Luis Castro'. The main heading is 'Add user' with a progress indicator showing steps 1 through 5. Step 1, 'Set user details', is active. It includes a text input for 'User name*' containing 'lcastro1' and a link to 'Add another user'. Below this is the 'Select AWS access type' section, which has two options: 'Programmatic access' (selected) and 'AWS Management Console access'. The 'Programmatic access' option is described as enabling an access key ID and secret access key for the AWS API, CLI, SDK, and other development tools. At the bottom, there are 'Cancel' and 'Next: Permissions' buttons.

Step 3

Set Permissions and click Attach existing policies directly and search for AmazonEC2FullAccess policy



The screenshot shows the AWS IAM 'Add user' console at Step 2, 'Set permissions'. The top navigation bar is the same as in Step 1. The progress indicator shows steps 1 through 5, with step 2 being the active one. Under the 'Set permissions' heading, there are three buttons: 'Add user to group', 'Copy permissions from existing user', and 'Attach existing policies directly'. The 'Attach existing policies directly' button is highlighted with a blue border. Below these buttons is a 'Create policy' button and a search bar. The search bar contains the text 'AmazonEC2Full'. Below the search bar is a table showing the search results. The table has columns for 'Policy name', 'Type', and 'Used as'. There is one result: 'AmazonEC2FullAccess', which is an 'AWS managed' policy and is used as a 'Permissions policy (1)'. At the bottom, there are 'Cancel', 'Previous', and 'Next: Tags' buttons.

Policy name	Type	Used as
AmazonEC2FullAccess	AWS managed	Permissions policy (1)

Step 4

Create User

Services
Resource Groups
★
Luis Castro

Add user
1 2 3 4 5

Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name	lcastro1
AWS access type	Programmatic access - with an access key
Permissions boundary	Permissions boundary is not set

Permissions summary

The following policies will be attached to the user shown above.

Type	Name
Managed policy	AmazonEC2FullAccess

Tags

No tags were added.

Cancel Previous Create user

Step 5

Download Access Key

Add user
1 2 3 4 5

✓ Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://lcastrose.signin.aws.amazon.com/console>

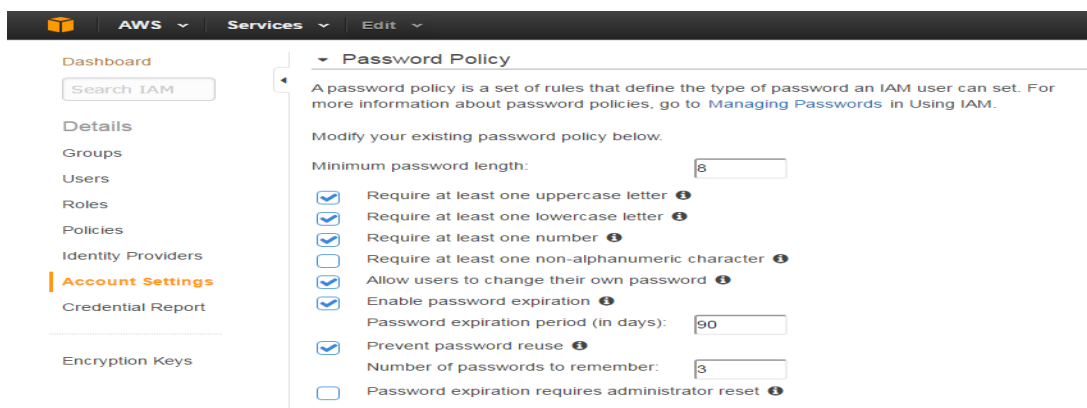
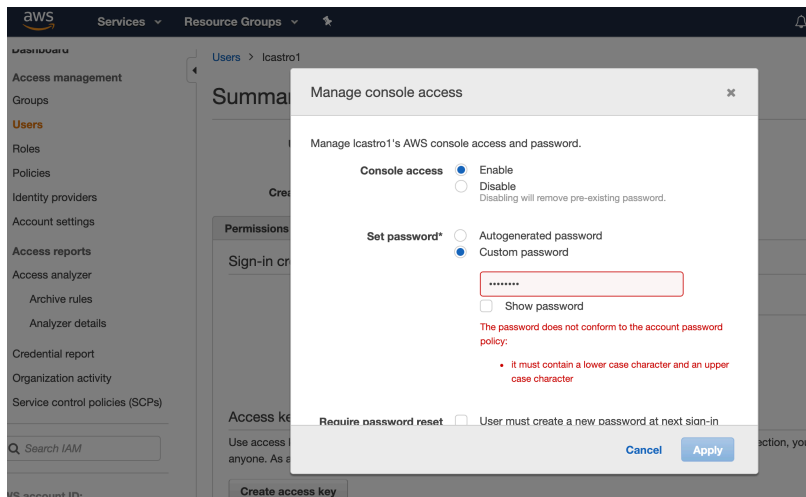
Download .csv

	User	Access key ID	Secret access key
▶	✓ lcastro1	AKIARYHDXUF35JP63XDZ	***** Show

Step 6

Mark the user created in the main panel and click on Users> Security Credentials> Console Password> Manage

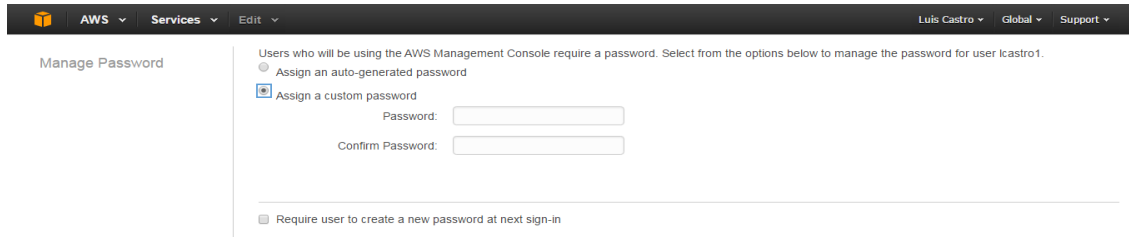
- Enable Access
- Custom password
 - First use the following password
 - 12345678
 - Check the error message
 - Go to the main menu in Account Settings and validate the Password policies



Step 7

Go back to Users> Security Credentials> Console Password> Manage

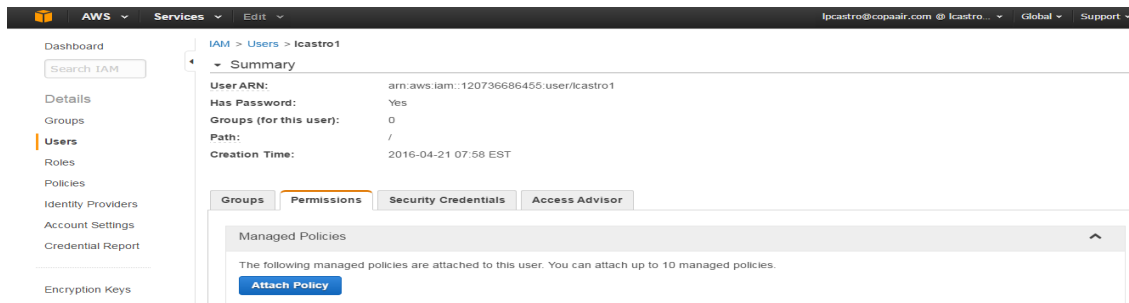
- Create a password of your convenience following the defined password guidelines
- Leave unchecked
- **Require user to create a new password at Next sign-in**



The screenshot shows the 'Manage Password' page in the AWS IAM console. The page title is 'Manage Password'. Below the title, there is a heading 'Users who will be using the AWS Management Console require a password. Select from the options below to manage the password for user lcastro1.' There are two radio button options: 'Assign an auto-generated password' (which is selected) and 'Assign a custom password'. Below the 'Assign a custom password' option, there are two input fields: 'Password:' and 'Confirm Password:'. At the bottom of the page, there is a checkbox labeled 'Require user to create a new password at next sign-in' which is currently unchecked.

Step 8

Mark the user created in the main panel and click on the Permissions tab> Attach Policy



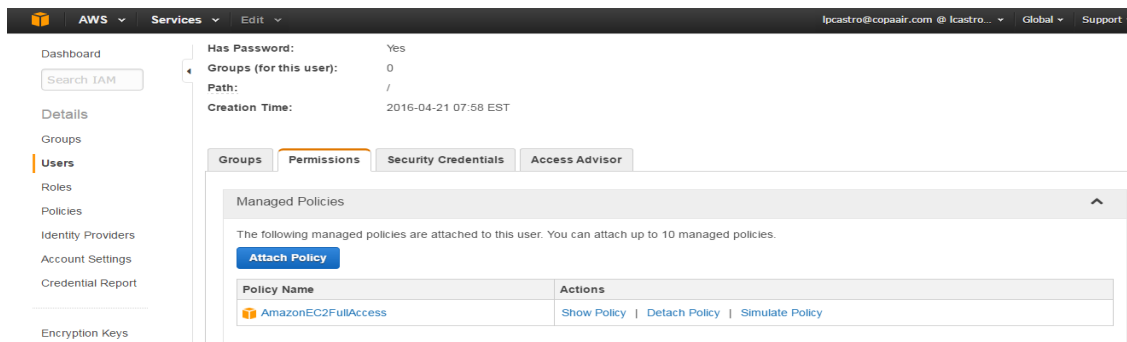
The screenshot shows the 'Permissions' tab in the AWS IAM console for user 'lcastro1'. The page title is 'IAM > Users > lcastro1'. Below the title, there is a 'Summary' section with the following details: 'User ARN: arn:aws:iam::120736686455:user/lcastro1', 'Has Password: Yes', 'Groups (for this user): 0', 'Path: /', and 'Creation Time: 2016-04-21 07:58 EST'. Below the summary, there are four tabs: 'Groups', 'Permissions' (which is selected), 'Security Credentials', and 'Access Advisor'. Under the 'Permissions' tab, there is a section titled 'Managed Policies' with the text 'The following managed policies are attached to this user. You can attach up to 10 managed policies.' and a blue button labeled 'Attach Policy'.

- Look for the policy AmazonEC2FullAccess



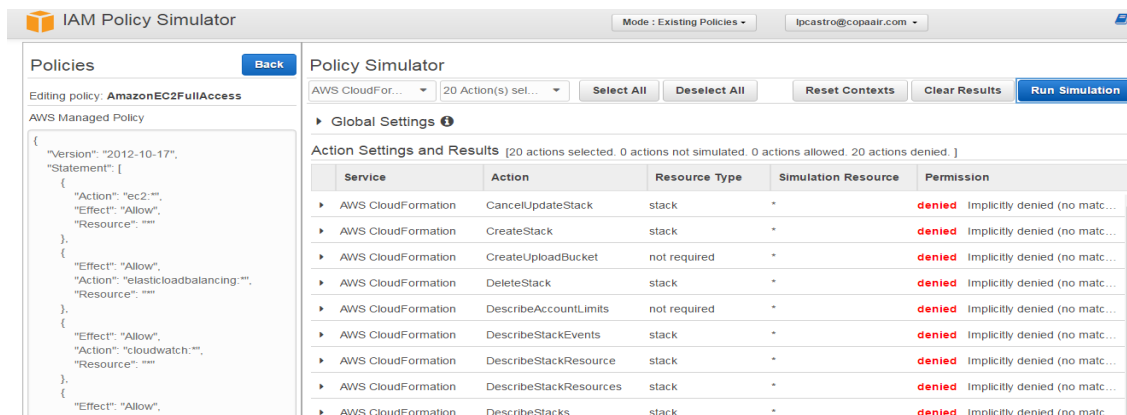
The screenshot shows the 'Attach Policy' page in the AWS IAM console. The page title is 'Attach Policy'. Below the title, there is a heading 'Select one or more policies to attach. Each user can have up to 10 policies attached.' There is a search bar with the text 'Filter: Policy Type' and a 'Filter' button. Below the search bar, there is a table with the following columns: 'Policy Name', 'Attached Entities', 'Creation Time', and 'Edited Time'. The table shows one policy: 'AmazonEC2FullAccess' with 4 attached entities, created on 2015-02-06 13:40 EST, and edited on 2015-02-06 13:40 EST. The table also indicates 'Showing 197 results'.

- Next click **Simulate Policy**
 - Check your username and the policy created (In case you don't take it automatically)



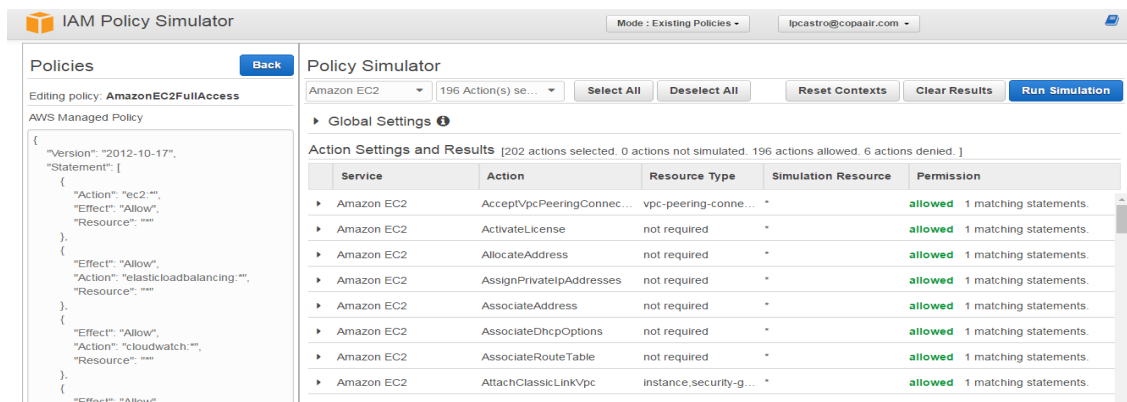
The screenshot shows the AWS IAM console interface. The left sidebar contains navigation links: Dashboard, Search IAM, Details, Groups, Users (selected), Roles, Policies, Identity Providers, Account Settings, Credential Report, and Encryption Keys. The main content area shows user details for 'lpcastro@copair.com', including 'Has Password: Yes', 'Groups (for this user): 0', 'Path: /', and 'Creation Time: 2016-04-21 07:58 EST'. Below this, there are tabs for Groups, Permissions (selected), Security Credentials, and Access Advisor. The 'Permissions' tab shows 'Managed Policies' with a table listing attached policies. One policy, 'AmazonEC2FullAccess', is listed with links to 'Show Policy', 'Detach Policy', and 'Simulate Policy'.

- A new browser will open, select the following services one by one, Select All and click Run Simulation:
 - o S3, Route53 y Cloudfront
 - o Validate actions are denied



The screenshot shows the IAM Policy Simulator interface. The left sidebar shows 'Policies' with a 'Back' button and 'Editing policy: AmazonEC2FullAccess'. The main content area shows 'Policy Simulator' with a dropdown for 'AWS CloudFor...' and '20 Action(s) selected'. There are buttons for 'Select All', 'Deselect All', 'Reset Contexts', 'Clear Results', and 'Run Simulation'. Below these, there's a 'Global Settings' section and an 'Action Settings and Results' table. The table has columns: Service, Action, Resource Type, Simulation Resource, and Permission. The results show that all 20 actions are denied, with the permission being 'Implicitly denied (no match...)'. The denied actions include CancelUpdateStack, CreateStack, CreateUploadBucket, DeleteStack, DescribeAccountLimits, DescribeStackEvents, DescribeStackResource, DescribeStackResources, and DescribeStacks.

- Click Deselect All and choose EC2 service and Select All, Run Simulator
- Validate that actions are allowed

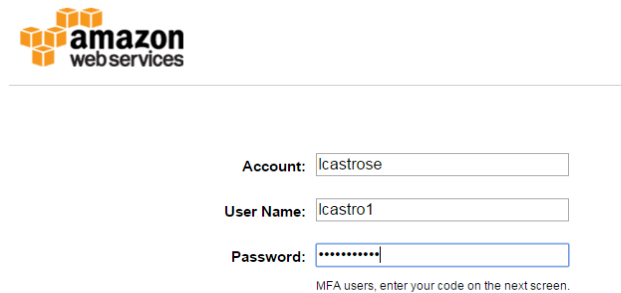


The screenshot shows the IAM Policy Simulator interface. The left sidebar shows 'Policies' with a 'Back' button and 'Editing policy: AmazonEC2FullAccess'. The main content area shows 'Policy Simulator' with a dropdown for 'Amazon EC2' and '196 Action(s) selected'. There are buttons for 'Select All', 'Deselect All', 'Reset Contexts', 'Clear Results', and 'Run Simulation'. Below these, there's a 'Global Settings' section and an 'Action Settings and Results' table. The table has columns: Service, Action, Resource Type, Simulation Resource, and Permission. The results show that all 196 actions are allowed, with the permission being '1 matching statements'. The allowed actions include AcceptVpcPeeringConnections, ActivateLicense, AllocateAddress, AssignPrivateIpAddresses, AssociateAddress, AssociateDhcpOptions, AssociateRouteTable, AttachClassicLinkVpc, AttachInternetGateway, and AttachNetworkInterface.

Step 9

Click on Dashboard in the AIM main menu and use the IAM users sign-in link to test the access of the new user:

<https://450006219561.signin.aws.amazon.com/console>



The image shows the Amazon Web Services (AWS) login page. At the top is the AWS logo. Below it, there are three input fields: "Account:" with the value "lcastrose", "User Name:" with the value "lcastro1", and "Password:" with a masked password "*****". Below the password field, there is a small text note: "MFA users, enter your code on the next screen."

- Click on the AWS RDS service and validate if you have permissions to access the settings



- Then go to VPC, CloudFormation and Cloudfront services and you should behave the same way
- Enter the EC2 service and create a new instance t2.micro with default values