

Task 1

Create 2 IAM Users with different Permission

1-create first user with autogenerated password and has CLI and GUI access

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Access type* **Programmatic access**

Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

AWS Management Console access

Enables a **password** that allows users to sign-in to the AWS Management Console.

Console password* Autogenerated password

Custom password

Require password reset User must create a new password at next sign-in

Users automatically get the [IAMUserChangePassword](#) policy to allow them to change their own password.

2-Create group has s3 full access privileges

Create group

Create a group and select the policies to be attached to the group. Using groups is a best-practice way to manage users' permissions by job functions, AWS service access, or your custom permissions. [Learn more](#)

Group name

Filter policies

Showing 8 results

	Policy name	Type	Used as	Description
<input type="checkbox"/>	AmazonDMSRedshiftS3...	AWS managed	None	Provides access to manage S3 settings for Redshift endpoint...
<input checked="" type="checkbox"/>	AmazonS3FullAccess	AWS managed	None	Provides full access to all buckets via the AWS Management ...
<input type="checkbox"/>	AmazonS3ReadOnlyAcc...	AWS managed	None	Provides read only access to all buckets via the AWS Manag...
<input type="checkbox"/>	AWSDataSyncS3Bucket...	Customer managed	None	

3-Add user to group

▼ Set permissions

 Add user to group  Copy permissions from existing user  Attach existing policies directly

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

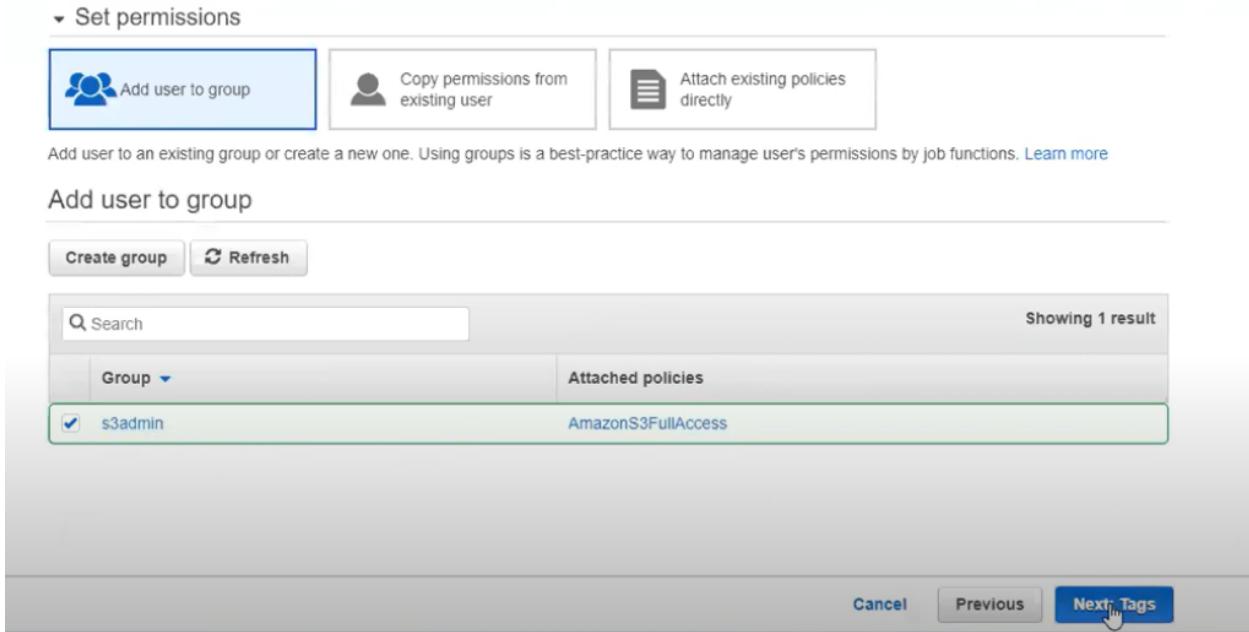
Add user to group

[Create group](#) [Refresh](#)

Group	Attached policies
<input checked="" type="checkbox"/> s3admin	AmazonS3FullAccess

Showing 1 result

[Cancel](#) [Previous](#) [Next !\[\]\(331831374f10e8c7fe483c7fa2c6e388_img.jpg\)](#) [Tags](#)



▼ Set permissions

 Add user to group  Copy permissions from existing user  Attach existing policies directly

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

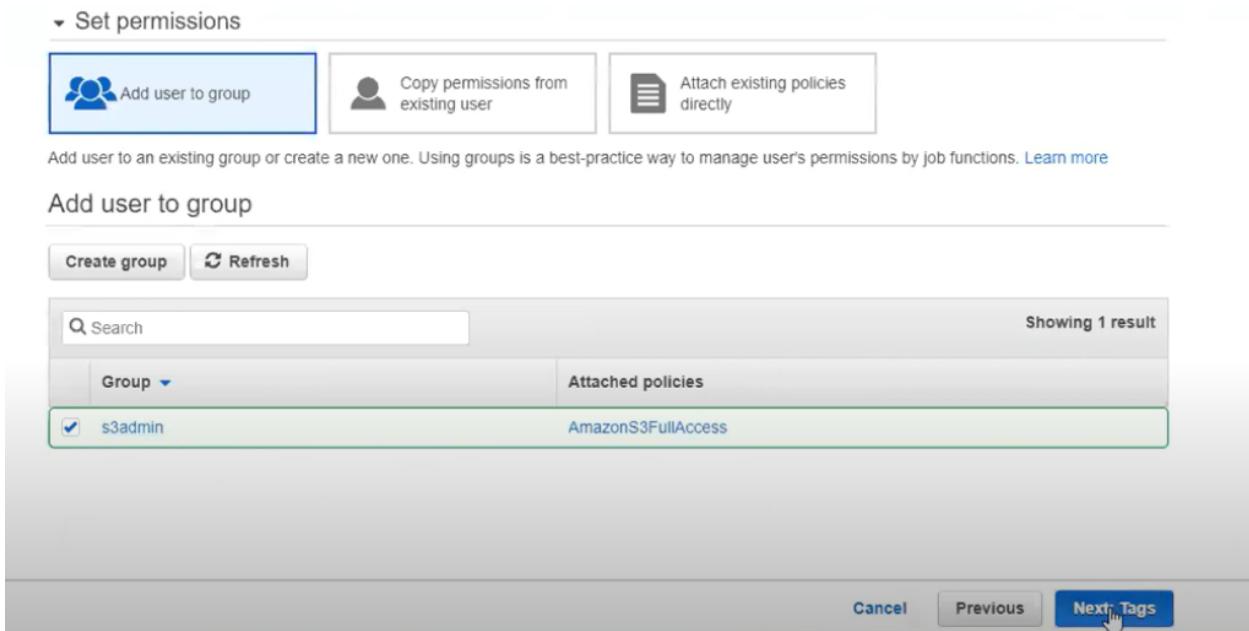
Add user to group

[Create group](#) [Refresh](#)

Group	Attached policies
<input checked="" type="checkbox"/> s3admin	AmazonS3FullAccess

Showing 1 result

[Cancel](#) [Previous](#) [Next !\[\]\(cc550a6475ae0a4b1b30a2c3ecc0f2f6_img.jpg\)](#) [Tags](#)



3-Again in second user ,same process to The first User but we add him to the new created Group (admin: full access)

	Policy Name	Attached Entities
<input type="checkbox"/>	AmazonS3FullAccess	1
<input type="checkbox"/>	IAMUserChangePassword	1
<input checked="" type="checkbox"/>	AdministratorAccess	0

Search		
	Group Name	Users
<input type="checkbox"/>	Admin	0
<input type="checkbox"/>	s3admin	1

Install Apache Server on EC2 instance

1-Create Security Group

The screenshot shows two tables for managing security group rules. The top table, titled 'Inbound rules (3)', lists three rules: one for ICMP (All ICMP - IPv4), one for HTTP (TCP port 80), and one for SSH (TCP port 22). The bottom table, titled 'Outbound rules (1)', lists one rule allowing all traffic (IPv4) from the security group.

Inbound rules (3)					
	Name	Security group rule...	IP version	Type	Protocol
<input type="checkbox"/>	-	sgr-0d36c7facfb14de19	IPv4	All ICMP - IPv4	ICMP
<input type="checkbox"/>	-	sgr-0a1c0a510b795f4e7	IPv4	HTTP	TCP
<input type="checkbox"/>	-	sgr-0353764da22bc54...	IPv4	SSH	TCP

Outbound rules (1)					
	Name	Security group rule...	IP version	Type	
<input type="checkbox"/>	-	sgr-0e027533e65f636b6	IPv4	All traffic	<input type="checkbox"/>

2>Create and Launch Instance

-Create instance by default settings then select already created security group then install instance pair keys to access server

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group
 Select an existing security group

Security Group ID	Name	Description	Actions
<input type="checkbox"/> sg-c7adad8d	default	default VPC security group	Copy to new
<input type="checkbox"/> sg-073fdfc11268ccac	launch-wizard-1	launch-wizard-1 created 2021-09-09T14:20:02.201+02:00	Copy to new
<input checked="" type="checkbox"/> sg-0c5359cf7c97c1d27	test-webserver	lgggg	Copy to new

⚠ Warning

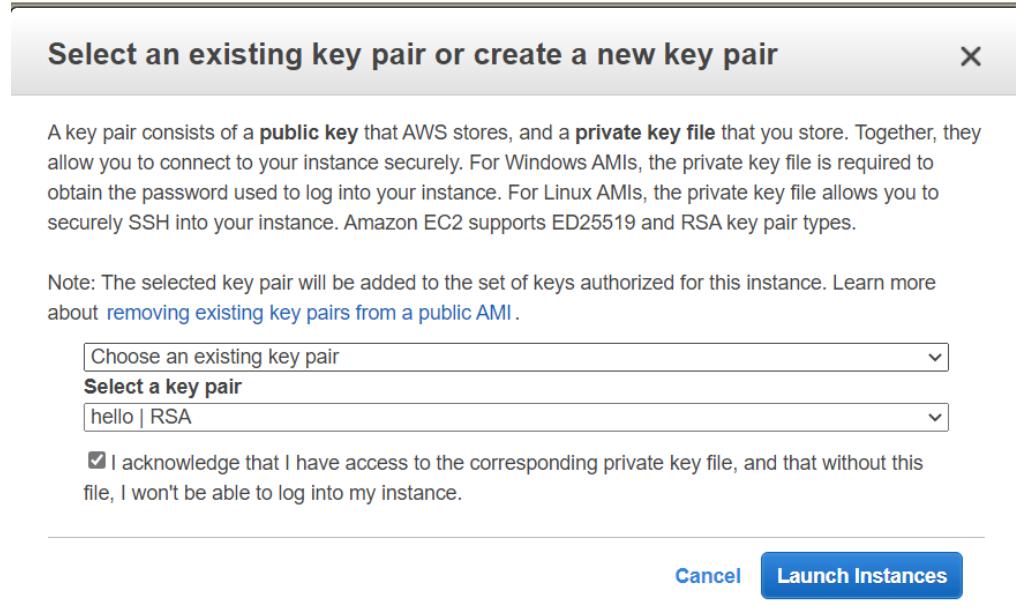
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Inbound rules for sg-0c5359cf7c97c1d27 (Selected security groups: sg-0c5359cf7c97c1d27)

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	
SSH	TCP	22	0.0.0.0/0	
All ICMP - IPv4	All	N/A	0.0.0.0/0	

[Cancel](#) [Previous](#) [Review and Launch](#)

-I already created key pairs so I will use them in this instance



3-Acess Instance and install and Launch Apache Server on it

-Access Instance through pair key using ssh command

```
C:\Programming\Embedded course>ssh -i hello.pem ec2-user@3.21.166.126
The authenticity of host '3.21.166.126 (3.21.166.126)' can't be established.
ECDSA key fingerprint is SHA256:vhSFAG+/aGUWdh8B1UJ/BQnhv5eh/E10o8NQvefB+lc.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.21.166.126' (ECDSA) to the list of known hosts.

      _|_ _|_
      _| (   /   Amazon Linux 2 AMI
      _| \_ |_|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-38-36 ~]$ sudo su
```

-then give root access then by applying the following commands:

```
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "<h1>Hello World from $(hostname -f)</h1>" > /var/www/html/index.html
```

The Result



Task 2

Create and Configure VPC

1-Create VPC

VPC settings

Name tag - *optional*
Creates a tag with a key of 'Name' and a value that you specify.

IPv4 CIDR block [Info](#)

IPv6 CIDR block [Info](#)
 No IPv6 CIDR block
 Amazon-provided IPv6 CIDR block
 IPv6 CIDR owned by me

Tenancy [Info](#)

2-Create Public Subnet

VPC ID
Create subnets in this VPC.

vpc-0ff8989215b1f2145 (Demo-VPC) ▾

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Demo-Public-SubnetA

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (Ohio) / us-east-2a ▾

IPv4 CIDR block [Info](#)
Q 10.0.0.0/24 X

▼ Tags - optional

Key	Value - optional
-----	------------------

3-Create Private Subnet

VPC ID
Create subnets in this VPC.

vpc-0ff8989215b1f2145 (Demo-VPC) ▾

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Demo-Private-SubnetB

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (Ohio) / us-east-2b ▾

IPv4 CIDR block [Info](#)
Q 10.0.1.0/24 X
10.0.1.0/24

Key Value - optional

4 - Create NAT

NAT gateway settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.

Demo-Nat-Gateway-PublicA

The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.

subnet-0e7a9b472718f4d0d (Demo-Public-SubnetA) ▾

Connectivity type
Select a connectivity type for the NAT gateway.

Public
 Private

Elastic IP allocation ID [Info](#)
Assign an Elastic IP address to the NAT gateway.

Select an Elastic IP ▾ [Allocate Elastic IP](#)

5-Create Internet Gateway

Internet gateway settings

Name tag
Creates a tag with a key of 'Name' and a value that you specify.

Demo-ig

6-Create Router

Route table settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.

Demo-Router

VPC
The VPC to use for this route table.

vpc-0ff8989215b1f2145 (Demo-VPC) ▾

Edit routes		
Destination	Target	Status
10.0.0.0/16	<input type="text" value="local"/> <input type="button" value="X"/>	<input checked="" type="checkbox"/> Active
Propagated		
No		
Edit routes		
Destination	Target	Status
<input type="text" value="0.0.0.0/0"/> <input type="button" value="X"/>	<input type="text" value="nat-0a4c5954745d1cce0"/> <input type="button" value="X"/>	-
Propagated		
No		
<input type="button" value="Remove"/>		
<input type="button" value="Add route"/>		

Cancel

6-Create EC2 on Public Subnet

Network (i)	<input type="text" value="vpc-0ff8989215b1f2145 Demo-VPC"/> <input type="button" value="▼"/>	<input type="button" value="C"/> Create new VPC
Subnet (i)	<input type="text" value="subnet-0e7a9b472718f4d0d Demo-Public-SubnetA"/> <input type="button" value="▼"/>	<input type="button" value="Create new subnet"/>
Assign Public IP (i)	<input type="text" value="Use subnet setting (Enable)"/> <input type="button" value="▼"/>	

7-Create EC2 on Private Subnet

Network (i)	<input type="text" value="vpc-0ff8989215b1f2145 Demo-VPC"/> <input type="button" value="▼"/>	<input type="button" value="C"/> Create new VPC
Subnet (i)	<input type="text" value="subnet-0776a0c7171686d70 Demo-Private-Subnet1"/> <input type="button" value="▼"/>	<input type="button" value="Create new subnet"/>
Auto-assign Public IP (i)	<input type="text" value="Use subnet setting (Enable)"/> <input type="button" value="▼"/>	