

Karthick VM

Batch – CIS 1.3

Mock Assessment 2 – AWS

1. Deploy a Node.js static web app on an EC2 instance with Apache and access it publicly via a browser.

- Launch a Ec2 Instance

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name: [Add additional tags](#)

Application and OS Images (Amazon Machine Image) [Info](#)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

Recents [Quick Start](#)

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux Debian

[Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

Summary

Number of instances [Info](#): 1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.9.2...[read more](#)
ami-04c08f8baa14af291

Virtual server type (instance type)
t3.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#) [Preview code](#)

Key pair name - required
 [Create new key pair](#)

Network settings [Info](#) [Edit](#)

Network [Info](#)
vpc-0c282e294992db20a

Subnet [Info](#)
No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)
Enable

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☐ Create security group ☒ Select existing security group

Common security groups [Info](#)
Select security groups: [Compare security group rules](#)

Security groups that you add or remove here will be added to or removed from all your network interfaces.

Summary

Number of instances [Info](#): 1

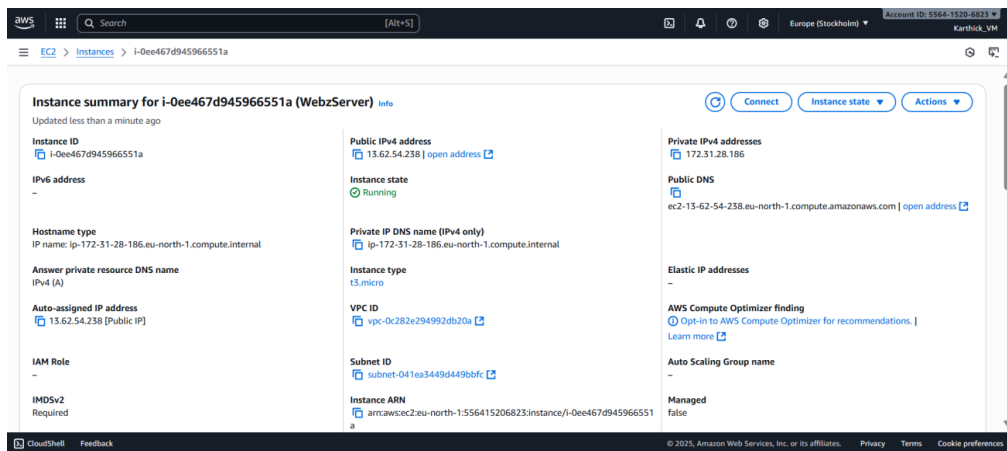
Software Image (AMI)
Amazon Linux 2023 AMI 2023.9.2...[read more](#)
ami-04c08f8baa14af291

Virtual server type (instance type)
t3.micro

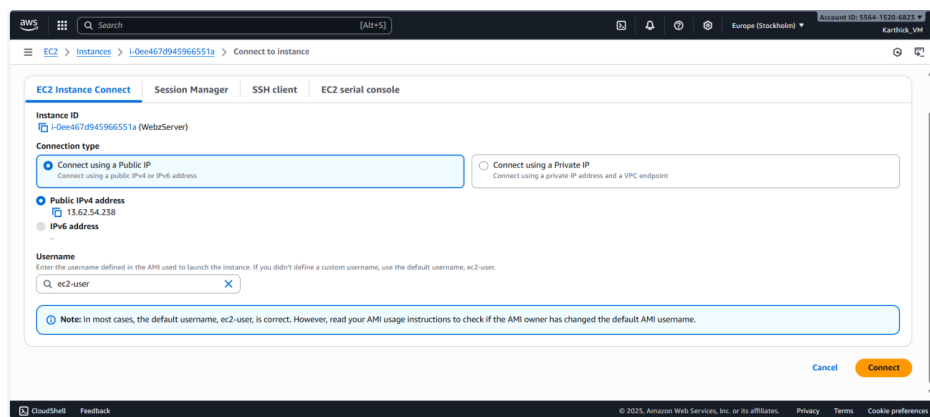
Firewall (security group)
default

Storage (volumes)
1 volume(s) - 8 GiB

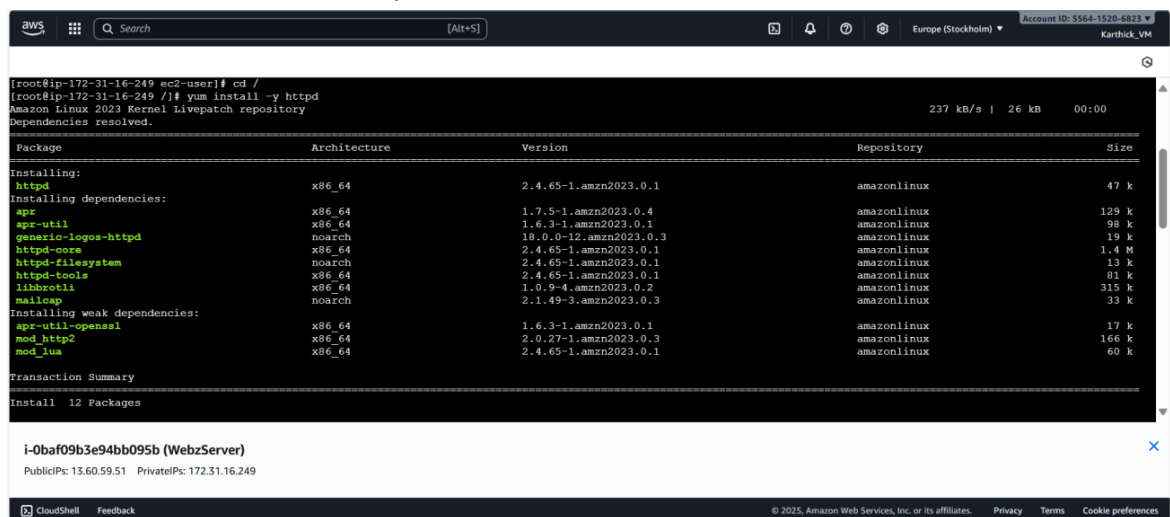
[Cancel](#) [Launch instance](#) [Preview code](#)



- Connect to the Ec2 Instance



- Install and enable httpd server in the server machine



```
Installed:
apr-1.7.5-1.amzn2023.0.4.x86_64          apr-util-1.6.3-1.amzn2023.0.1.x86_64          apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch  httpd-2.4.65-1.amzn2023.0.1.x86_64          httpd-core-2.4.65-1.amzn2023.0.1.x86_64
httpd-filesystem-2.4.65-1.amzn2023.0.1.noarch      httpd-tools-2.4.65-1.amzn2023.0.1.x86_64      libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch              mod_http2-2.0.27-1.amzn2023.0.3.x86_64      mod_lua-2.4.65-1.amzn2023.0.1.x86_64

Complete!
[root@ip-172-31-16-249 /]# cd /var/www/html
[root@ip-172-31-16-249 html]# ls
[root@ip-172-31-16-249 html]# systemctl enable httpd
Unknown command verb enable.
[root@ip-172-31-16-249 html]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service -> /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-16-249 html]# systemctl start httpd
[root@ip-172-31-16-249 html]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Tue 2025-10-07 10:57:31 UTC; 7s ago
     Docs: man:httpd.service(8)
   Main PID: 25845 (httpd)
   Status: "Started, listening on: port 80"
    Tasks: 177 (limit: 1053)
   Memory: 13.3M
      CPU: 70ms
   CGroup: /system.slice/httpd.service
           └─25845 /usr/sbin/httpd -DFOREGROUND
           └─25846 /usr/sbin/httpd -DFOREGROUND
```

- Install npm and nodejs in the server machine

```
[root@ip-172-31-16-249 html]# dnf install npm
Last metadata expiration check: 0:07:29 ago on Tue Oct 7 10:53:07 2025.
Dependencies resolved.

=====
Package                                Architecture      Version                                Repository          Size
=====
Installing:
nodejs-npm                             x86_64            1:10.8.2-1.18.20.8.1.amzn2023.0.2    amazonlinux          1.9 M
Installing dependencies:
nodejs                                 x86_64            1:18.20.8-1.amzn2023.0.2             amazonlinux          13 M
nodejs-libs                            x86_64            1:18.20.8-1.amzn2023.0.2             amazonlinux          14 M
Installing weak dependencies:
nodejs-docs                            noarch            1:18.20.8-1.amzn2023.0.2             amazonlinux          7.8 M
nodejs-full-i18n                       x86_64            1:18.20.8-1.amzn2023.0.2             amazonlinux          8.4 M
=====

Transaction Summary
-----
Install 5 Packages

Total download size: 45 M
Installed size: 223 M
Is this ok [y/N]: y
Downloading Packages:
(1/5): nodejs-docs-18.20.8-1.amzn2023.0.2.noarch.rpm                43 MB/s | 7.8 MB  00:00
(2/5): nodejs-18.20.8-1.amzn2023.0.2.x86_64.rpm                   52 MB/s | 13 MB   00:00
(3/5): nodejs-full-i18n-18.20.8-1.amzn2023.0.2.x86_64.rpm          28 MB/s | 8.4 MB  00:00
(4/5): nodejs-libs-18.20.8-1.amzn2023.0.2.x86_64.rpm              60 MB/s | 14 MB   00:00

i-0baf09b3e94bb095b (WebzServer)
```

```
Complete!
[root@ip-172-31-16-249 html]# dnf install nodejs
Last metadata expiration check: 0:09:08 ago on Tue Oct 7 10:53:07 2025.
Package nodejs-1:18.20.8-1.amzn2023.0.2.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-16-249 html]#
```

i-0baf09b3e94bb095b (WebzServer)
PublicIPs: 13.60.59.51 PrivateIPs: 172.31.16.249

- Create a node app and create script.js and index.html files

```
root@ip-172-31-16-249 node-app]# npm init -y
Wrote to /var/www/html/node-app/package.json:

{
  "name": "app",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": ""
}

root@ip-172-31-16-249 node-app]#
```

i-0baf09b3e94bb095b (WebzServer)

PublicIPs: 13.60.59.51 PrivateIPs: 172.31.16.249

```
aws
Search [Alt+S]
Europe (Stockholm) Account ID: 5564-1520-6523 Karthick_VM

html>
head>
  <title>Node.js Static Web App</title>
</head>
body>
  <h1>Helloo Static Node JS Web Page !</h1>

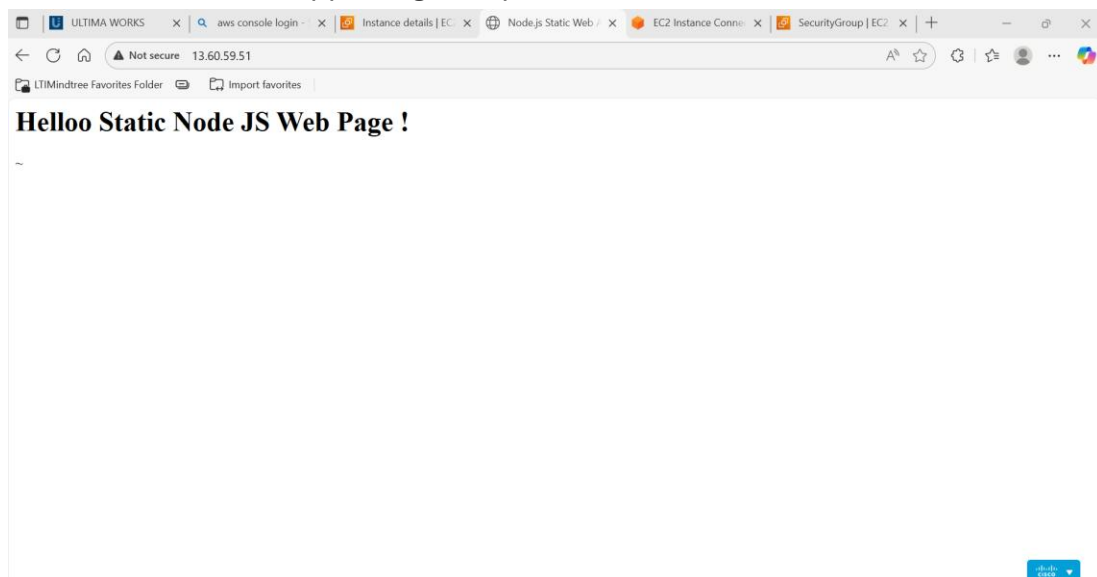
  <script src="js/script.js"></script>
</body>
/html>

-- INSERT -- 11,0 All
```

i-0baf09b3e94bb095b (WebzServer)

PublicIPs: 13.60.59.51 PrivateIPs: 172.31.16.249

- Run the node app using the apache server.



2. Create an S3 bucket, enable versioning, upload multiple file versions, and restore a previous version.

- Create a s3 bucket

The screenshot shows the 'Create bucket' page in the AWS Management Console. The page is for the 'Europe (Stockholm)' region. Under 'General configuration', the 'AWS Region' is 'Europe (Stockholm) eu-north-1'. The 'Bucket type' is set to 'General purpose'. The 'Bucket name' is 'Bucketzz710'. There is a section for 'Copy settings from existing bucket - optional' with a 'Choose bucket' button. At the bottom, there is an 'Object Ownership' section.

Create bucket [info](#)

Buckets are containers for data stored in S3.

General configuration

AWS Region
Europe (Stockholm) eu-north-1

Bucket type [Info](#)

☒ **General purpose**
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ **Directory**
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name [Info](#)

Bucketzz710

Bucket names must be 3 to 63 characters and unique within the global namespace. Bucket names must also begin and end with a letter or number. Valid characters are a-z, 0-9, periods (.), and hyphens (-). [Learn More](#)

Copy settings from existing bucket - optional
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Format: s3://bucket/prefix

Object Ownership [Info](#)

- Enable versioning and upload a file

This section shows the 'Bucket Versioning' configuration for the bucket. It includes a warning about public access and a checkbox to acknowledge that turning off block all public access might result in the bucket and its objects becoming public. The 'Bucket Versioning' is set to 'Enable'.

S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Turning off block all public access might result in this bucket and the objects within becoming public
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☒ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning

☐ Disable

☒ **Enable**

Tags - optional (0)
You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

The screenshot shows the 'Buckets' page in the AWS Management Console. A green banner at the top indicates that the bucket 'bucketzz710' was successfully created. Below this, there is a table of 'General purpose buckets' with columns for Name, AWS Region, and Creation date. The table lists three buckets: 'bucketzz710', 'cf-templates--3otz1ek1bqy1-us-east-1', and 'webzzz'. On the right side, there are sections for 'Account snapshot' and 'External access summary'.

Successfully created bucket "bucketzz710"
To upload files and folders, or to configure additional bucket settings, choose [View details](#).

General purpose buckets [All AWS Regions](#) **Directory buckets**

General purpose buckets (3) [info](#)

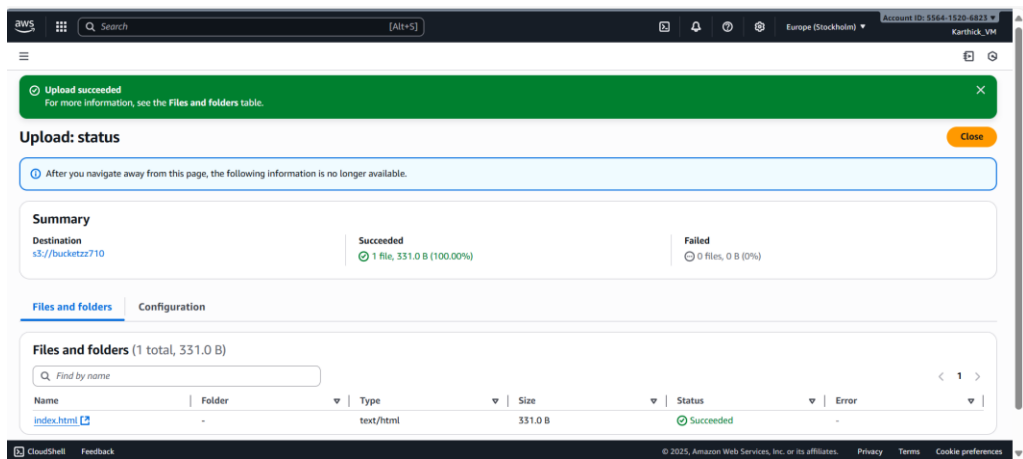
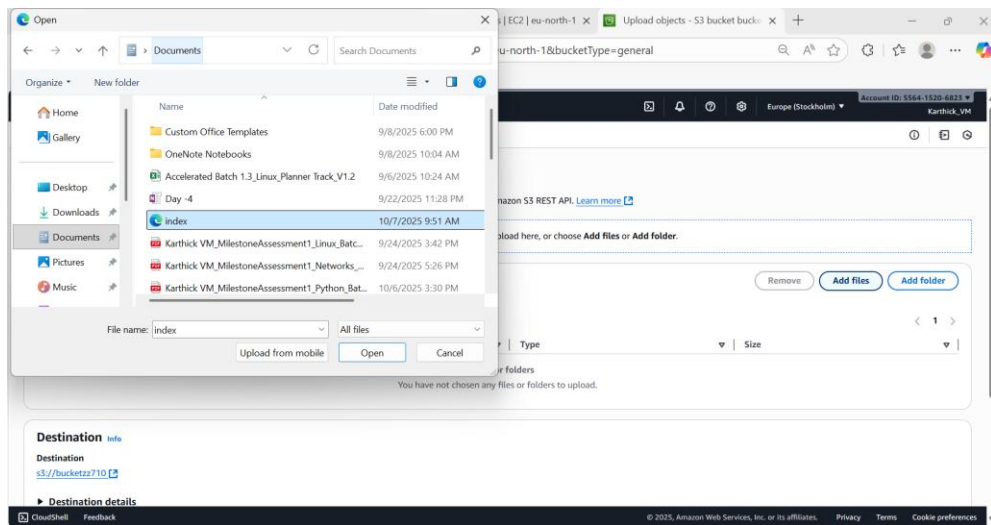
Buckets are containers for data stored in S3.

[Find buckets by name](#)

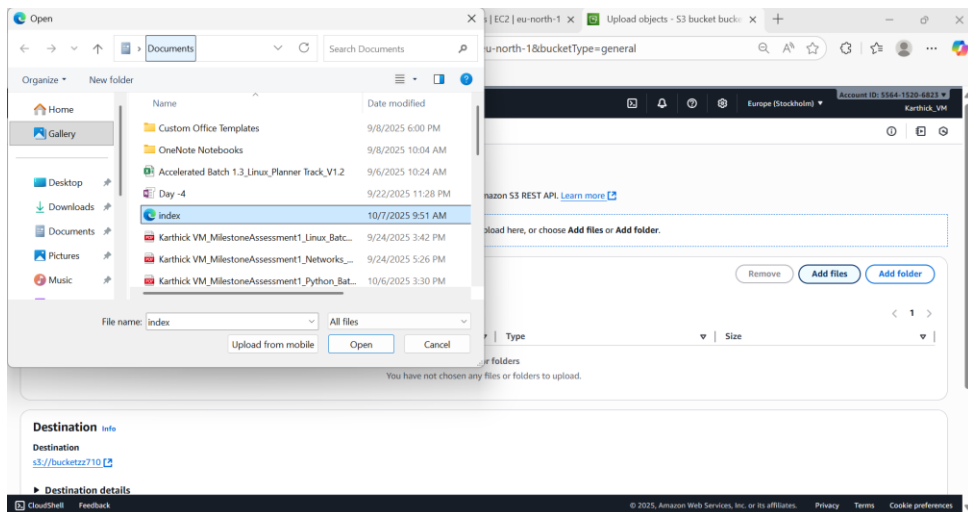
Name	AWS Region	Creation date
<input type="radio"/> bucketzz710	Europe (Stockholm) eu-north-1	October 7, 2025, 16:45:43 (UTC+05:30)
<input type="radio"/> cf-templates--3otz1ek1bqy1-us-east-1	US East (N. Virginia) us-east-1	October 6, 2025, 17:03:18 (UTC+05:30)
<input type="radio"/> webzzz	Europe (Stockholm) eu-north-1	October 7, 2025, 09:52:12 (UTC+05:30)

Account snapshot [info](#) [View dashboard](#)
Updated daily
Storage Lens provides visibility into storage usage and activity trends.

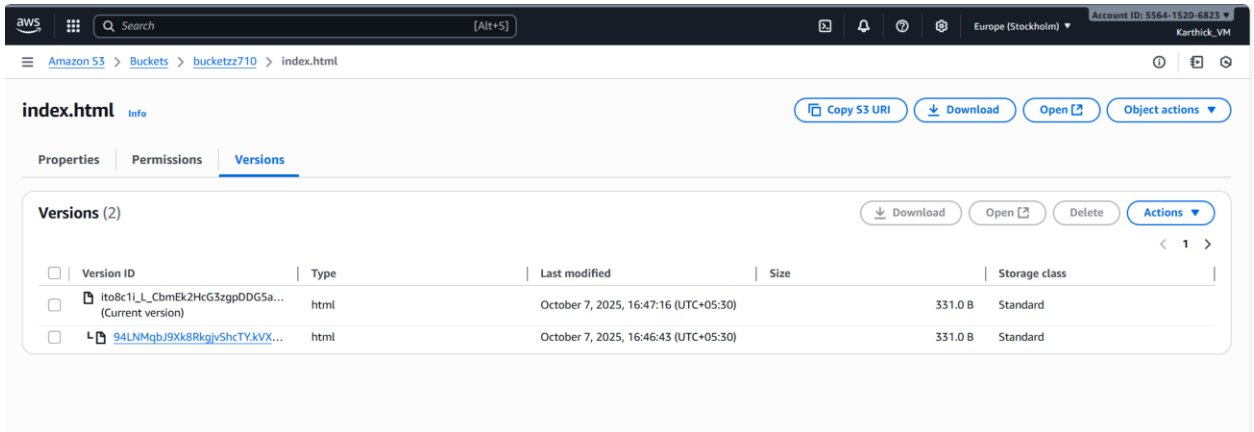
External access summary - new [info](#)
Updated daily
External access findings help you identify bucket permissions that allow public access or access from other AWS accounts.



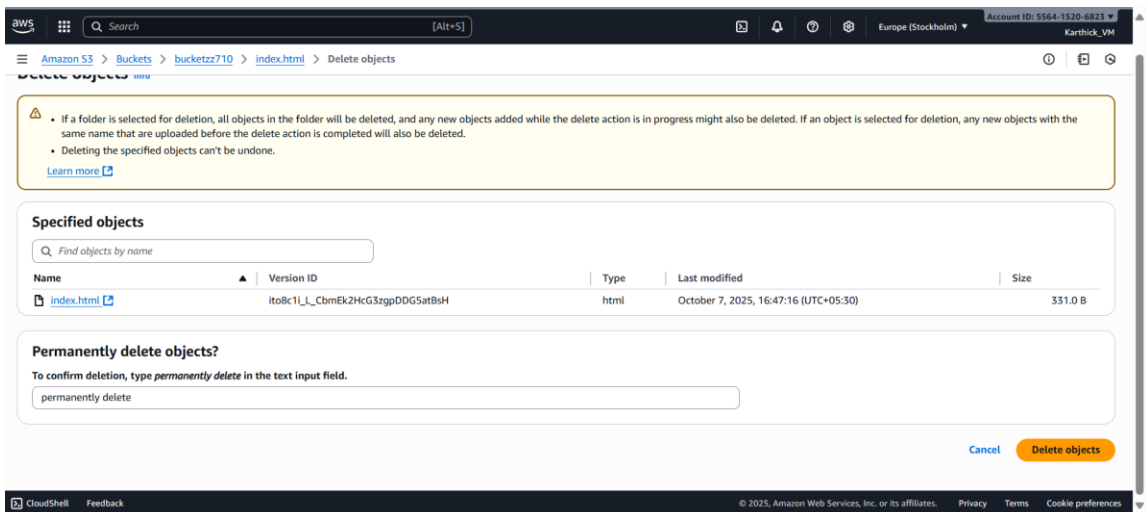
- Upload the same file again



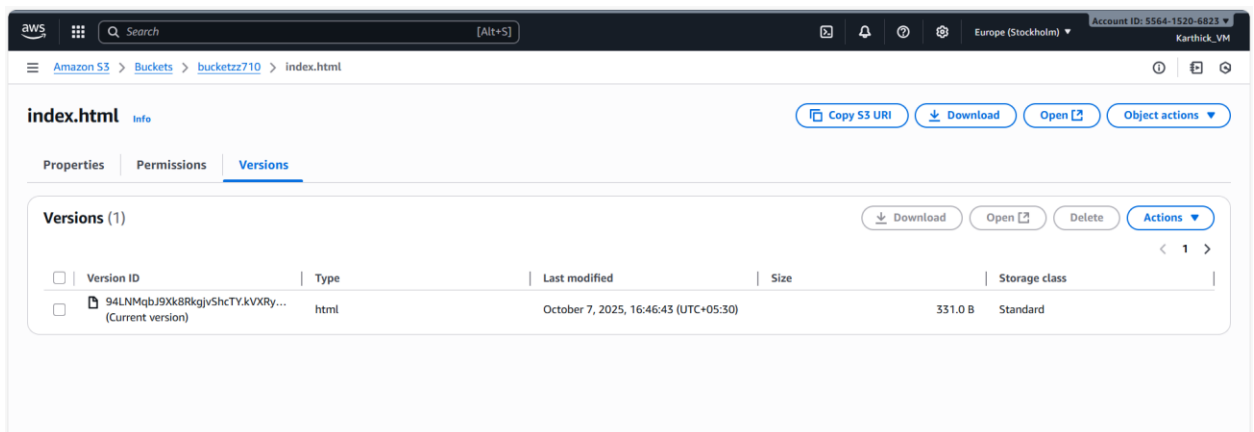
- Go to the file and go to the versions page , there you can see the pervious versions of the file.



- Now delete the current version of the file



- You can see after we delete the current version of the file the previous version is restored.



3. Launch two EC2 instances in different subnets and verify connectivity using securitygroups.

- EC2 Created using default vpc and subnet using default security group --- permissions inclusive of tcp,http,https.

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name
WebServer [Add additional tags](#)

Application and OS Images (Amazon Machine Image) [Info](#)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

Recents **Quick Start**

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux Debian

[Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.9.2...[read more](#)
ami-04c08f8baa14af291

Virtual server type (instance type)
t3.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#) [Preview code](#)

Key pair name - required
Karthick_710 [Create new key pair](#)

Network settings [Info](#) [Edit](#)

Network [Info](#)
vpc-0c282e294992db20a

Subnet [Info](#)
No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)
Enable

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☐ Create security group ☒ Select existing security group

Common security groups [Info](#)
Select security groups
default sg-0e6f4c315e65e0c57 [X](#)
VPC: vpc-0c282e294992db20a [Compare security group rules](#)

Security groups that you add or remove here will be added to or removed from all your network interfaces.

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.9.2...[read more](#)
ami-04c08f8baa14af291

Virtual server type (instance type)
t3.micro

Firewall (security group)
default

Storage (volumes)
1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#) [Preview code](#)

aws

Search

[Alt+S]

Europe (Stockholm)

Account ID: 5564-1520-6823

Karthick_VM

EC2 > Instances > i-0ee467d945966551a

Instance summary for i-0ee467d945966551a (WebzServer) info

Updated less than a minute ago

Instance ID

i-0ee467d945966551a

IPv6 address

-

Hostname type

IP name: ip-172-31-28-186.eu-north-1.compute.internal

Answer private resource DNS name

IPv4 (A)

Auto-assigned IP address

13.62.54.238 [Public IP]

IAM Role

-

IMDSv2

Required

Public IPv4 address

13.62.54.238 | open address

Instance state

Running

Private IP DNS name (IPv4 only)

ip-172-31-28-186.eu-north-1.compute.internal

Instance type

t3.micro

VPC ID

vpc-0c282e294992db20a

Subnet ID

subnet-041ea3449d449bbfc

Instance ARN

arn:aws:ec2:eu-north-1:556415206823:instance/i-0ee467d945966551a

Private IPv4 addresses

172.31.28.186

Public DNS

ec2-13-62-54-238.eu-north-1.compute.amazonaws.com | open address

Elastic IP addresses

-

AWS Compute Optimizer finding

Opt-in to AWS Compute Optimizer for recommendations. | Learn more

Auto Scaling Group name

-

Managed

false

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Search

[Alt+S]

Europe (Stockholm)

Account ID: 5564-1520-6823

Karthick_VM

EC2 > Instances > i-0ee467d945966551a > Connect to instance

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID

i-0ee467d945966551a (WebzServer)

Connection type

Connect using a Public IP

Connect using a public IPv4 or IPv6 address

Public IPv4 address

13.62.54.238

IPv6 address

-

Username

Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ec2-user.

ec2-user

Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel Connect

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS CloudShell terminal session showing the installation of httpd on an Amazon Linux 2023 instance.

```
[root@ip-172-31-16-249 ec2-user]# cd /
[root@ip-172-31-16-249 /]# yum install -y httpd
Amazon Linux 2023 Kernel Livepatch repository
Dependencies resolved.
```

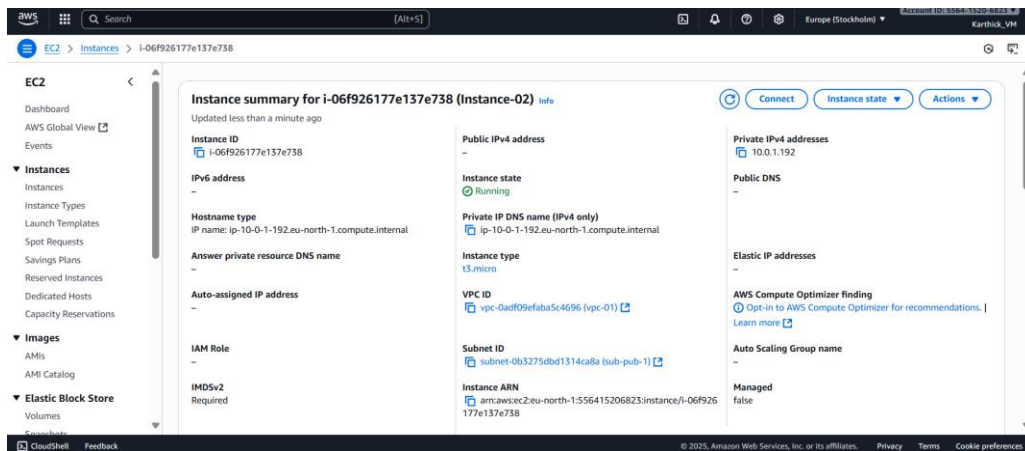
Package	Architecture	Version	Repository	Size
Installing:				
httpd	x86_64	2.4.65-1.amzn2023.0.1	amazonlinux	47 k
Installing dependencies:				
apr	x86_64	1.7.5-1.amzn2023.0.4	amazonlinux	129 k
apr-util	x86_64	1.6.3-1.amzn2023.0.1	amazonlinux	98 k
generic-logos-httpd	noarch	10.0.0-12.amzn2023.0.3	amazonlinux	19 k
httpd-core	x86_64	2.4.65-1.amzn2023.0.1	amazonlinux	1.4 M
httpd-filesystem	noarch	2.4.65-1.amzn2023.0.1	amazonlinux	13 k
httpd-tools	x86_64	2.4.65-1.amzn2023.0.1	amazonlinux	81 k
libbrotli	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	315 k
mailcap	noarch	2.1.49-3.amzn2023.0.3	amazonlinux	33 k
Installing weak dependencies:				
apr-util-openssl	x86_64	1.6.3-1.amzn2023.0.1	amazonlinux	17 k
mod_http2	x86_64	2.0.27-1.amzn2023.0.3	amazonlinux	166 k
mod_lua	x86_64	2.4.65-1.amzn2023.0.1	amazonlinux	60 k
Transaction Summary				
Install 12 Packages				

i-0baf09b3e94bb095b (WebzServer)
PublicIPs: 13.60.59.51 PrivateIPs: 172.31.16.249

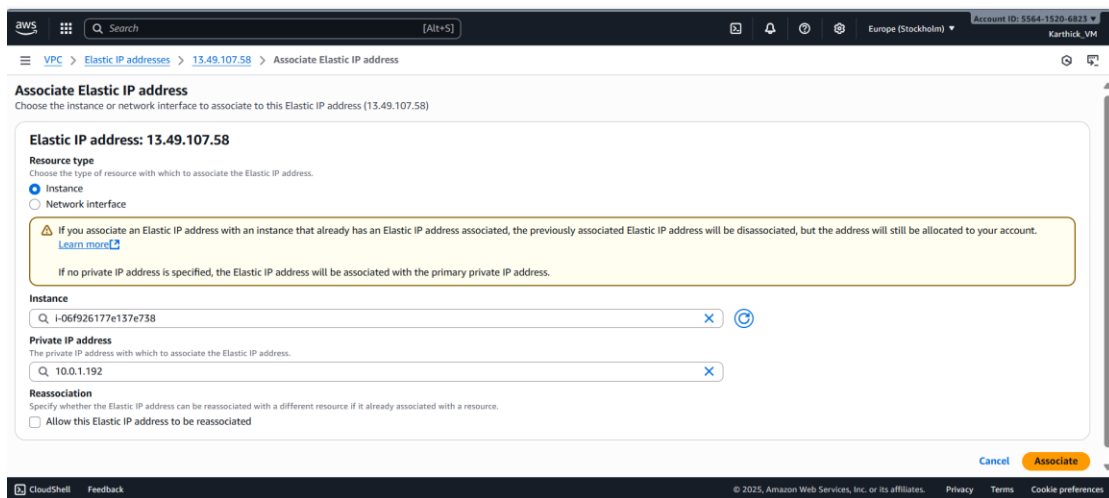
- Create another instance using another vpc i.e vpc-01

AWS Management Console "Launch an instance" page. The "Name and tags" section shows the instance name "Instance-02". The "Application and OS Images (Amazon Machine Image)" section shows the selected AMI "Amazon Linux 2023 AMI 2023.9.2...". The "Summary" section shows the instance configuration: 1 instance, Software Image (AMI) Amazon Linux 2023 AMI 2023.9.2..., Virtual server type (instance type) t3.micro, Firewall (security group) New security group, and Storage (volumes) 1 volume(s) - 8 GiB. The "Launch instance" button is visible.

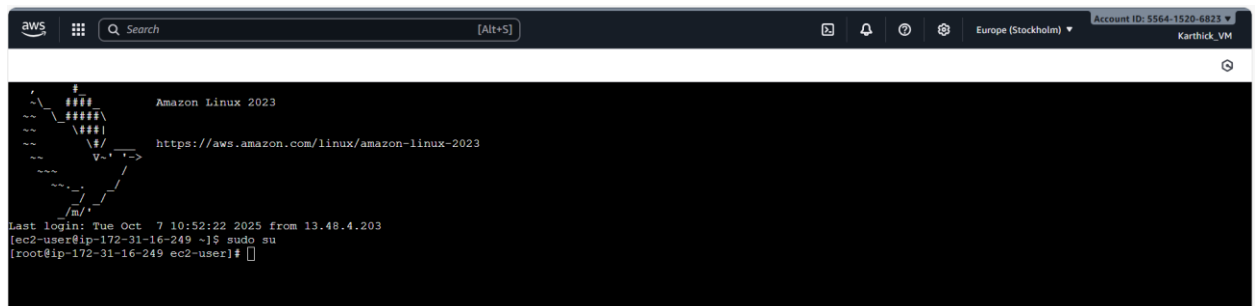
AWS Management Console "Launch an instance" page, showing the "Network settings" section. The "VPC" dropdown is set to "vpc-0ad09efaba5c4696 (vpc-01)". The "Subnet" dropdown is set to "subnet-0b3275dbd1314ca8a". The "Auto-assign public IP" dropdown is set to "Disable". The "Firewall (security group)" section shows the "Create security group" button selected. The "Common security groups" section shows the "default" security group selected. The "Launch instance" button is visible.



- You can see that the elastic ip address i.e public ip is not assigned to this instance so we want to allocate the elastic ip address to this instance.



- Now you can connect to the ec2 instance .



4. Configure a NAT Gateway in a public subnet and allow private instances to download updates from the internet.
 - Create a vpc and create a public subnet

This screenshot shows the 'Create VPC' page in the AWS Management Console. The page is titled 'VPC settings' and includes a 'Resources to create' section with two radio buttons: 'VPC only' (selected) and 'VPC and more'. Below this is a 'Name tag - optional' section with a text input field containing 'vpc-01'. The 'IPv4 CIDR block' section has two radio buttons: 'IPv4 CIDR manual input' (selected) and 'IPAM-allocated IPv4 CIDR block'. The 'IPv4 CIDR' text input field contains '10.0.0.0/16'. The 'IPv6 CIDR block' section has four radio buttons: 'No IPv6 CIDR block' (selected), 'IPAM-allocated IPv6 CIDR block', 'Amazon-provided IPv6 CIDR block', and 'IPv6 CIDR owned by me'. The page footer includes the AWS logo, a search bar, and account information for 'Europe (Stockholm)' and 'Karthick_VM'.

This screenshot shows the 'Create subnet' page in the AWS Management Console. The page is titled 'Subnet 1 of 1' and includes a 'Subnet name' section with a text input field containing 'sub-pub-1'. The 'Availability Zone' section has a dropdown menu showing 'Europe (Stockholm) / eu-north-1a'. The 'IPv4 VPC CIDR block' section has a dropdown menu showing '10.0.0.0/16'. The 'IPv4 subnet CIDR block' section has a text input field containing '10.0.1.0/24'. The 'Tags - optional' section includes a 'Key' input field with 'Name' and a 'Value - optional' input field with 'sub-pub-1'. The page footer includes the AWS logo, a search bar, and account information for 'Europe (Stockholm)' and 'Karthick_VM'.

- Create a internet gateway and route table , then associate the route table with the subnet and edit the route with the internet gateway.

This screenshot shows the 'Create internet gateway' page in the AWS Management Console. The page is titled 'Create internet gateway' and includes an 'Internet gateway settings' section with a 'Name tag' text input field containing 'igw-1'. The 'Tags - optional' section includes a 'Key' input field with 'Name' and a 'Value - optional' input field with 'igw-1'. The page footer includes the AWS logo, a search bar, and account information for 'Europe (Stockholm)' and 'Karthick_VM'.

aws [Search] [Alt+S] Europe (Stockholm) Account ID: 5564-1520-6823 Karthick_VM

VPC > Route tables > rtb-0a6aeeded2121dbf4 > Edit routes

Edit routes

Destination	Target	Status	Propagated	Route Origin
10.0.0.0/16	local	Active	No	CreateRouteTable
0.0.0.0/0	Internet Gateway	-	No	CreateRoute

- Create a private subnet

aws [Search] [Alt+S] Europe (Stockholm) Account ID: 5564-1520-6823 Karthick_VM

VPC > Subnets > Create subnet

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
sub-private-1
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.
No preference

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.2.0/24 256 IPs

Tags - optional

Key	Value - optional
Name	sub-private-1

You can add 49 more tags.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Now create a NAT Gateway and select the public subnet (so that it can access the internet)

aws [Search] [Alt+S] Europe (Stockholm) Account ID: 5564-1520-6823 Karthick_VM

VPC > NAT gateways > Create NAT gateway

Elastic IP address 13.51.140.42 (eipalloc-08e3a72176c1e7a43) allocated.

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.
nat-1
The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.
subnet-0b3275dbd1314ca8a (sub-pub-1)

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.
eipalloc-08e3a72176c1e7a43

Additional settings [Info](#)

Tags

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Now create another route table and edit the route with NAT gateway and associate this route table with private subnet so it can access the internet using the NAT gateway.

aws Search [Alt+S] Europe (Stockholm) Account ID: 5564-1520-6323 Karthick_VM

VPC > Route tables > rtb-02c3433893c6ce395 > Edit routes

Edit routes

Destination	Target	Status	Propagated	Route Origin
10.0.0.0/16	local	Active	No	CreateRouteTable
0.0.0.0/0	NAT Gateway	-	No	CreateRoute

[Add route](#) [Cancel](#) [Preview](#) [Save changes](#)

You have successfully updated subnet associations for rtb-02c3433893c6ce395 / route-2.

rtb-02c3433893c6ce395 / route-2

[Actions](#)

Details [Info](#)

Route table ID
rtb-02c3433893c6ce395

VPC
vpc-0adf09efaba5c4696 | vpc-01

Main
No

Owner ID
556415206823

Explicit subnet associations
subnet-06a01337108a16f79 / sub-private-1

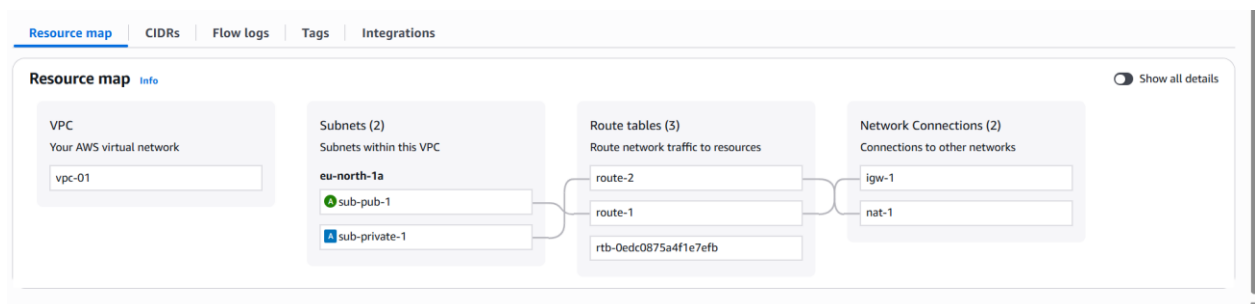
Edge associations
-

[Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

Routes (2)

[Filter routes](#) [Both](#) [Edit routes](#)

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	nat-08e4444b22173c2b1	Active	No	Create Route
10.0.0.0/16	local	Active	No	Create Route Table



5. Use IAM to create a user with programmatic access and restrict them to only read S3 buckets.

The screenshot shows the 'Specify user details' step of the AWS IAM 'Create user' wizard. The 'User name' field is set to 'User-01'. The 'Provide user access to the AWS Management Console - optional' checkbox is checked. Under the 'Are you providing console access to a person?' section, the 'I want to create an IAM user' radio button is selected. The 'Console password' section shows the 'Autogenerated password' option selected.

Specify user details

User details

User name
User-01

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , _ @ - (hyphen)

☒ Provide user access to the AWS Management Console - optional
If you're providing console access to a person, it's a best practice to manage their access in IAM Identity Center.

Are you providing console access to a person?

User type

☐ Specify a user in Identity Center - Recommended
We recommend that you use Identity Center to provide console access to a person. With Identity Center, you can centrally manage user access to their AWS accounts and cloud applications.

☒ I want to create an IAM user
We recommend that you create IAM users only if you need to enable programmatic access through access keys, service-specific credentials for AWS CodeCommit or Amazon Keyspaces, or a backup credential for emergency account access.

Console password

☐ Autogenerated password
You can view the password after you create the user.

The screenshot shows the 'Set permissions' step of the AWS IAM 'Create user' wizard. The 'Attach policies directly' option is selected under 'Permissions options'. In the 'Permissions policies' section, the search filter is set to 's3' and the 'AmazonS3ReadOnlyAccess' policy is selected.

Set permissions

Permissions options

☐ Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

☐ Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.

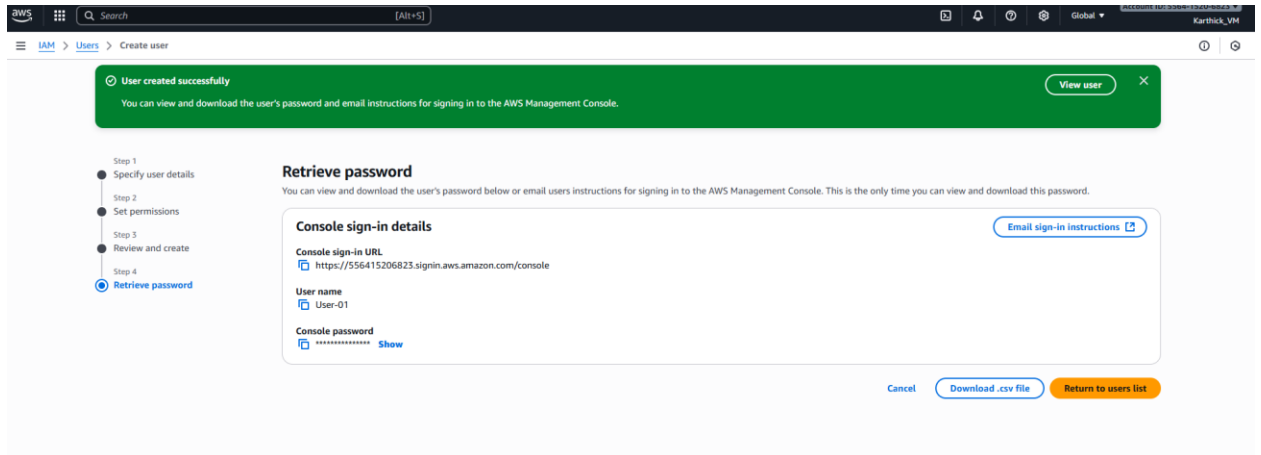
☒ Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Permissions policies (1/1393)

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

Filter by Type: All types (17 matches)

Policy name	Type	Attached entities
<input type="checkbox"/> AmazonDMSRedshiftS3Role	AWS managed	0
<input type="checkbox"/> AmazonS3FullAccess	AWS managed	0
<input type="checkbox"/> AmazonS3ObjectLambdaExecutionRolePol...	AWS managed	0
<input type="checkbox"/> AmazonS3OutpostsFullAccess	AWS managed	0
<input type="checkbox"/> AmazonS3OutpostsReadOnlyAccess	AWS managed	0
<input checked="" type="checkbox"/> AmazonS3ReadOnlyAccess	AWS managed	0
<input type="checkbox"/> AmazonS3TablesFullAccess	AWS managed	0
<input type="checkbox"/> AmazonS3TablesLakeFormationServiceRole	AWS managed	0

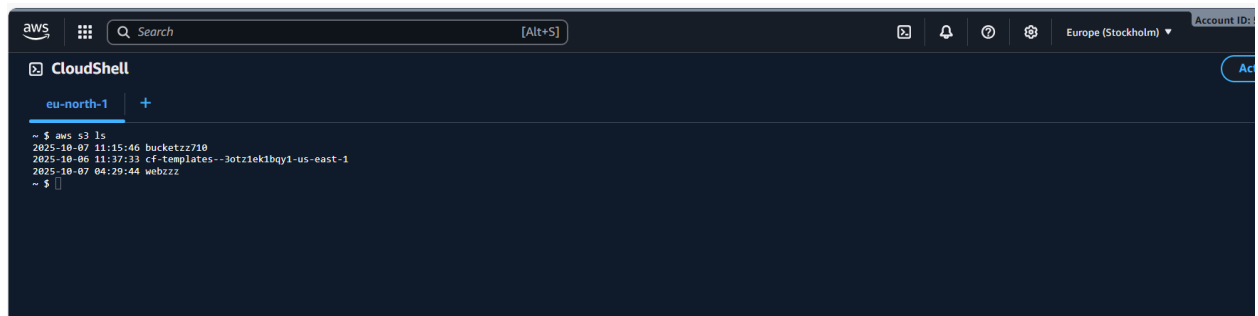


6. Run AWS CLI commands to list EC2 instances, S3 buckets, and IAM users in your account.

- List EC2 Instances



- List S3 buckets



- List IAM Users



The screenshot shows the AWS CloudShell interface. At the top, there's a navigation bar with the AWS logo, a search bar, and a region dropdown set to 'Europe (Stockholm)'. Below the navigation bar, the CloudShell title bar shows 'eu-north-1' and a '+' icon. The terminal window displays the command `~ $ aws iam list-users` and its output, which is a JSON array containing one user object.

```
~ $ aws iam list-users
{
  "Users": [
    {
      "Path": "/",
      "UserName": "User-01",
      "UserId": "AIDAYDDHOIMTUE67VM4MB",
      "Arn": "arn:aws:iam::556415206823:user/User-01",
      "CreateDate": "2025-10-07T11:40:26+00:00"
    }
  ]
}
```