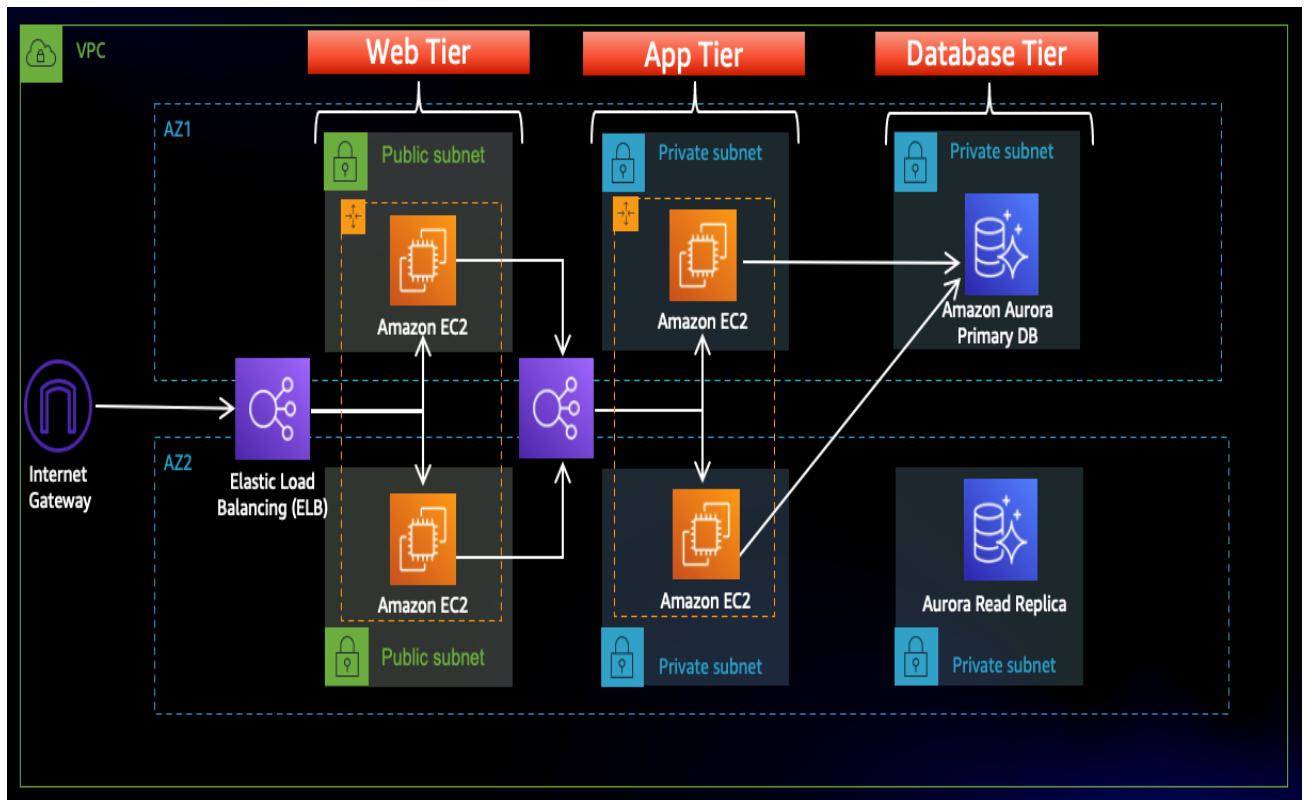


AWS Three Tier Web Architecture

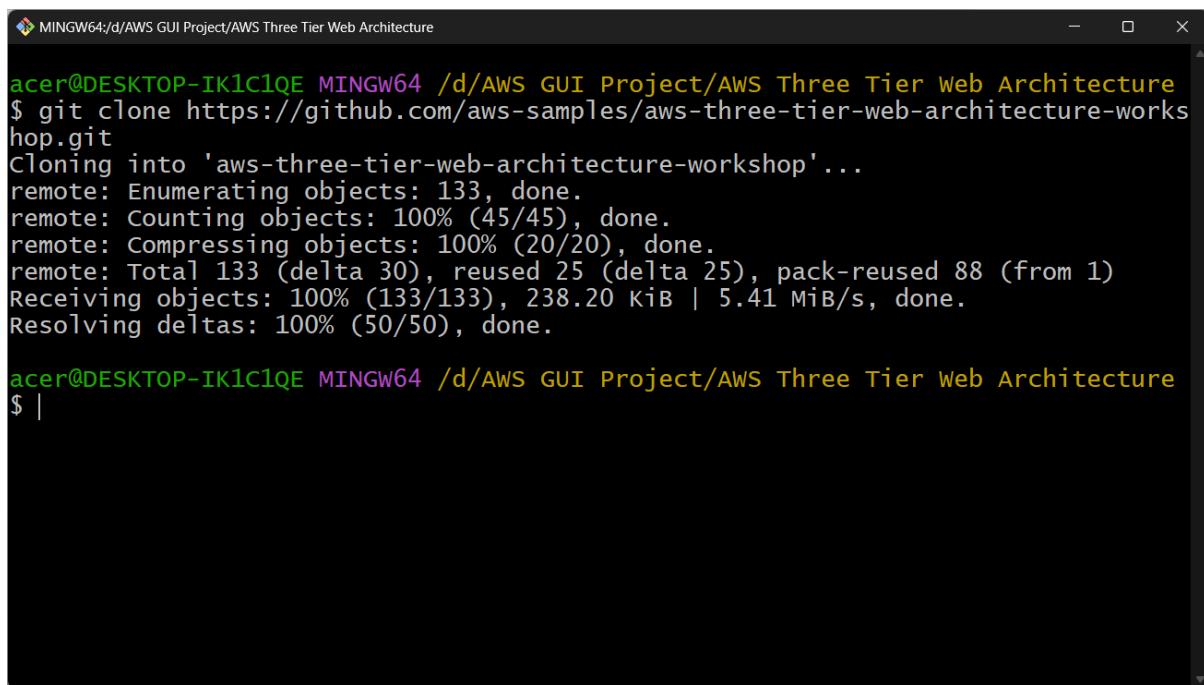


In this architecture, a public-facing Application Load Balancer forwards client traffic to our web tier EC2 instances. The web tier is running Nginx webservers that are configured to serve a React.js website and redirects our API calls to the application tier's internal facing load balancer. The internal facing load balancer then forwards that traffic to the application tier, which is written in Node.js. The application tier manipulates data in an Aurora MySQL multi-AZ database and returns it to our web tier. Load balancing, health checks and autoscaling groups are created at each layer to maintain the availability of this architecture.

1. Setup

- Clone the public repository for web app using below command

```
git clone https://github.com/aws-samples/aws-three-tier-web-architecture-workshop.git
```

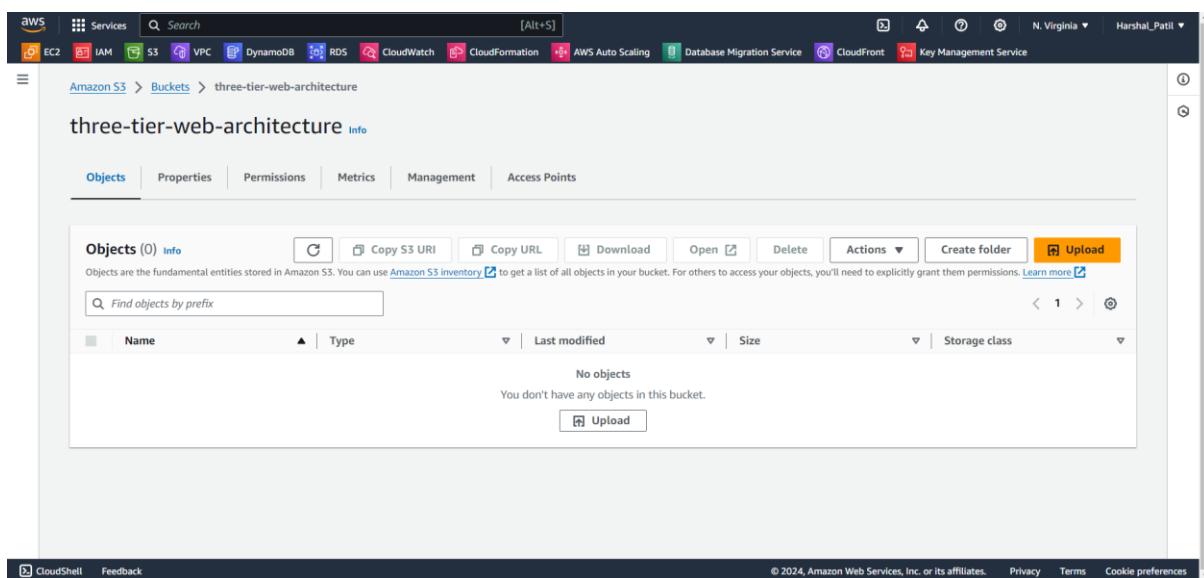


The screenshot shows a terminal window titled "MINGW64:/d/AWS GUI Project/AWS Three Tier Web Architecture". The command \$ git clone https://github.com/aws-samples/aws-three-tier-web-architecture-workshop.git is being run. The output shows the progress of cloning, including object enumeration, counting, compressing, and receiving objects, followed by a successful resolution of deltas.

```
acer@DESKTOP-IK1C1QE MINGW64 /d/AWS GUI Project/AWS Three Tier Web Architecture
$ git clone https://github.com/aws-samples/aws-three-tier-web-architecture-workshop.git
cloning into 'aws-three-tier-web-architecture-workshop'...
remote: Enumerating objects: 133, done.
remote: Counting objects: 100% (45/45), done.
remote: Compressing objects: 100% (20/20), done.
remote: Total 133 (delta 30), reused 25 (delta 25), pack-reused 88 (from 1)
Receiving objects: 100% (133/133), 238.20 KiB | 5.41 MiB/s, done.
Resolving deltas: 100% (50/50), done.

acer@DESKTOP-IK1C1QE MINGW64 /d/AWS GUI Project/AWS Three Tier Web Architecture
$ |
```

- Create S3 bucket with unique name in a region where we want to deploy application



➤ Create IAM role for EC2 with given policies

- AmazonSSMManagedInstanceCore
- AmazonS3ReadOnlyAccess

The screenshot shows the AWS Identity and Access Management (IAM) service in the AWS Management Console. The left sidebar navigation includes 'Dashboard', 'Access management' (with 'User groups', 'Users', and 'Roles' selected), 'Access reports' (with 'Access Analyzer', 'External access', 'Unused access', 'Analyzer settings', and 'Credential report'), and links to 'CloudShell' and 'Feedback'. The main content area is titled 'Roles (19) Info' and displays a table of existing roles. A green banner at the top says 'Roles deleted.' A red box highlights the 'Create role' button in the top right corner of the table header. The table columns are 'Role name', 'Trusted entities', and 'Last activity'. The roles listed are: AWSServiceRoleForAPIGateway, AWSServiceRoleForApplicationAutoScaling_DynamoDBTable, AWSServiceRoleForAutoScaling, AWSServiceRoleForElasticLoadBalancing, AWSServiceRoleForOrganizations, AWSServiceRoleForRDS, AWSServiceRoleForSSO, AWSServiceRoleForSupport, and AWSServiceRoleForTrustedAdvisor.

The screenshot shows the 'Create New Role' wizard, Step 2: Add permissions. The left sidebar shows 'Step 2: Add permissions' and 'Step 3: Name, review, and create'. The main content area has two sections: 'Trusted entity type' and 'Use case'. In 'Trusted entity type', the 'AWS service' option is selected and highlighted with a red box. In 'Use case', the 'Service or use case' dropdown is set to 'EC2' and highlighted with a red box. Below it, the 'Choose a use case for the specified service' section shows the 'EC2' option selected and highlighted with a red box. The 'EC2' use case description is: 'Allows EC2 instances to call AWS services on your behalf.'

Screenshot of the AWS IAM 'Create role' wizard - Step 1: Name, review, and create.

Role details

Role name: RoleForEC2toAccessS3

Description: Allows EC2 instances to call AWS services on your behalf.

Step 1: Select trusted entities

Trust policy:

```
1 "Version": "2012-10-17",
2 "Statement": [
```

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the AWS IAM 'Create role' wizard - Step 2: Add permissions.

Permissions policy summary

Policy name	Type	Attached as
AmazonS3ReadOnlyAccess	AWS managed	Permissions policy
AmazonSSMManagedInstanceCore	AWS managed	Permissions policy

Step 3: Add tags

Add tags - optional

No tags associated with the resource.

Add new tag

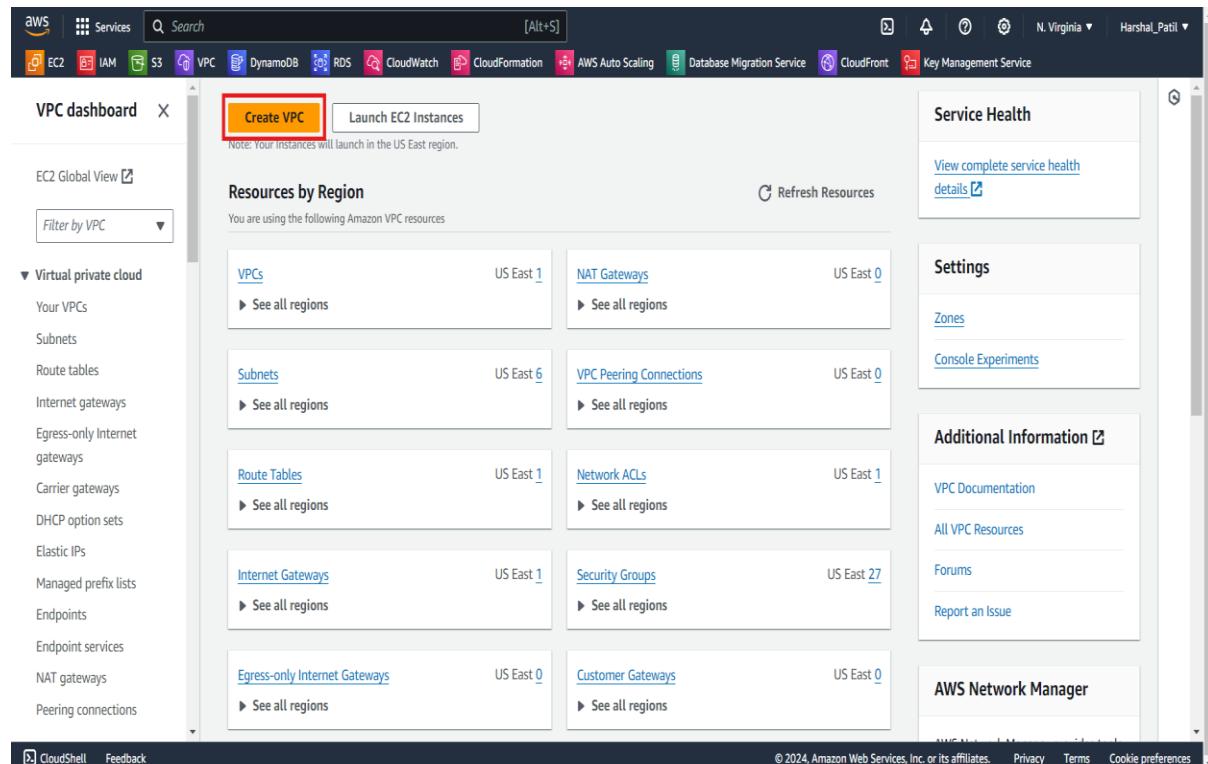
You can add up to 50 more tags.

Cancel Previous **Create role**

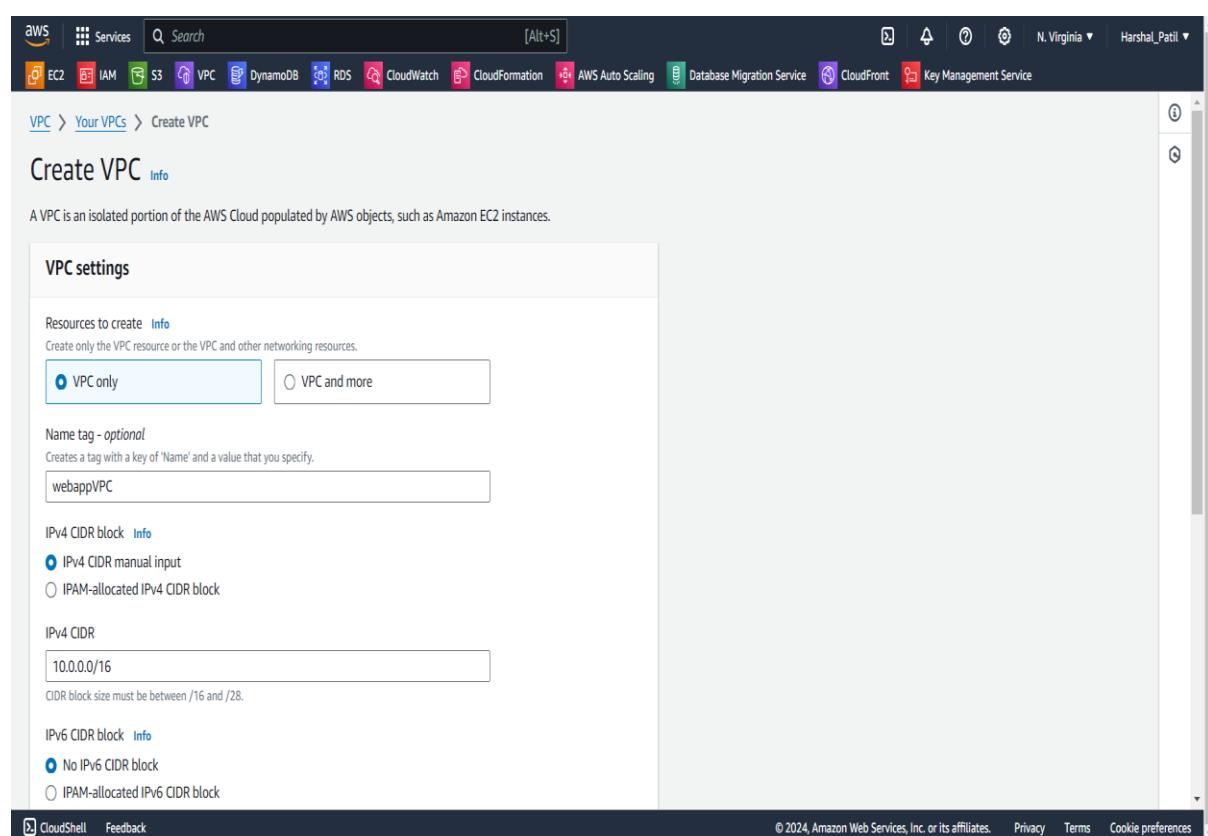
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

2. Networking and Security

➤ Create VPC



The screenshot shows the AWS VPC dashboard. At the top, there's a navigation bar with the AWS logo, services like EC2, IAM, S3, VPC, etc., and a search bar. Below the navigation bar is the main dashboard area. In the center, there's a section titled "Resources by Region" with various metrics for VPC components across the US East region. On the left, a sidebar lists "Virtual private cloud" resources such as Your VPCs, Subnets, Route tables, Internet gateways, Egress-only Internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, and Peering connections. At the bottom of the dashboard, there are links for CloudShell, Feedback, and copyright information.



The screenshot shows the "Create VPC" configuration page. At the top, there's a navigation bar with the AWS logo, services like EC2, IAM, S3, VPC, etc., and a search bar. Below the navigation bar is the "Create VPC" form. The form has several sections: "VPC settings" (with options for "VPC only" or "VPC and more"), "Name tag - optional" (with a text input field containing "webappVPC"), "IPv4 CIDR block" (with a dropdown set to "IPv4 CIDR manual input" and an input field containing "10.0.0.0/16"), and "IPv6 CIDR block" (with a dropdown set to "No IPv6 CIDR block"). At the bottom of the form, there are links for CloudShell, Feedback, and copyright information.

➤ Create 6 Subnets (2 Public & 4 Private) spread across 2 availability zones

The screenshot shows the AWS VPC dashboard with the following details:

- VPC dashboard** sidebar includes: EC2 Global View, Filter by VPC, Virtual private cloud, Your VPCs, Subnets, Route tables, Internet gateways, Egress-only Internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, and Peering connections.
- Subnets (6) Info** table:

Name	Subnet ID	State	VPC	IPv4 CIDR
-	subnet-0a104934d97135773	Available	vpc-004b68edb02c418b6	172.31.32.0/20
-	subnet-04b51bd77b241bef3	Available	vpc-004b68edb02c418b6	172.31.0.0/20
-	subnet-09fc30b206ed4a09a	Available	vpc-004b68edb02c418b6	172.31.64.0/20
-	subnet-070e40c11ffce896	Available	vpc-004b68edb02c418b6	172.31.80.0/20
-	subnet-06d76e89711e93528	Available	vpc-004b68edb02c418b6	172.31.48.0/20
-	subnet-08a92066e2b9f8e55	Available	vpc-004b68edb02c418b6	172.31.16.0/20
- Select a subnet** button.

The screenshot shows the 'Create subnet' wizard step 1: VPC selection. The VPC dropdown is set to 'vpc-0ca97ac97923efa57 (webappVPC)', which is highlighted with a red box.

The screenshot shows the 'Create subnet' wizard step 2: Subnet settings. The 'Subnet name' field contains 'Public-Web-Subnet-AZ1', which is highlighted with a red box. The 'Associated VPC CIDRs' section shows 'IPV4 CIDRs' set to '10.0.0.0/16'.

The screenshot shows the 'Create subnet' wizard step 3: Subnet configuration. The 'Subnet name' field contains 'Public-Web-Subnet-AZ1'. The 'Availability Zone' dropdown is set to 'US East (N. Virginia) / us-east-1a', which is highlighted with a red box. The 'IPv4 VPC CIDR block' dropdown is set to '10.0.0.0/16'. The 'IPv4 subnet CIDR block' dropdown is set to '10.0.0.0/24', which is highlighted with a red box. The 'Tags - optional' section shows a single tag 'Name: Public-Web-Subnet-AZ1'.

Subnet 2 of 2

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

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.

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.

IPv4 subnet CIDR block
 256 IPs

Tags - optional

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="Public-Web-Subnet-AZ2"/>

[Add new tag](#)
You can add 49 more tags.
[Remove](#)

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Subnet 3 of 3

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

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.

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.

IPv4 subnet CIDR block
 256 IPs

Tags - optional

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="Private-App-Subnet-AZ1"/>

[Add new tag](#)
You can add 49 more tags.
[Remove](#)

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Subnet 4 of 4

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

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.

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.

IPv4 subnet CIDR block
 256 IPs

Tags - optional

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="Private-App-Subnet-AZ2"/>

[Add new tag](#)
You can add 49 more tags.
[Remove](#)

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Subnet 5 of 5

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Private-DB-Subnet-AZ1

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 (N. Virginia) / us-east-1a

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
10.0.4.0/24 256 IPs

Tags - optional

Key	Value - optional
<input type="text" value="Name"/> X	<input type="text" value="Private-DB-Subnet-AZ1"/> X

[Add new tag](#)
You can add 49 more tags.
[Remove](#)

[CloudShell](#) [Feedback](#)

Subnet 6 of 6

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Private-DB-Subnet-AZ2

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 (N. Virginia) / us-east-1b

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
10.0.5.0/24 256 IPs

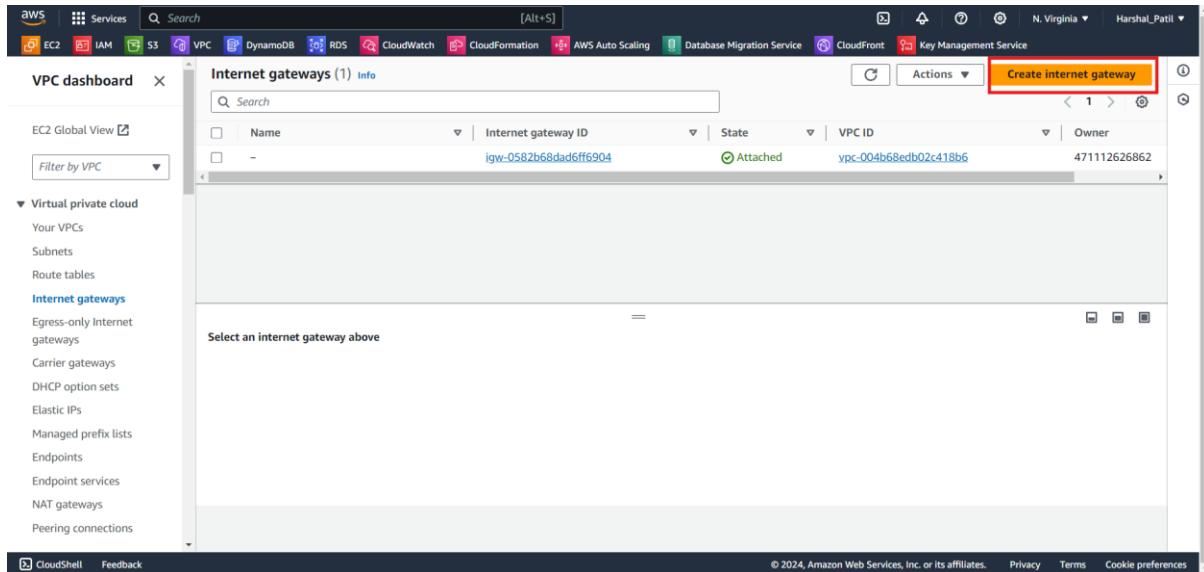
Tags - optional

Key	Value - optional
<input type="text" value="Name"/> X	<input type="text" value="Private-DB-Subnet-AZ2"/> X

[Add new tag](#)
You can add 49 more tags.
[Remove](#)

[CloudShell](#) [Feedback](#)

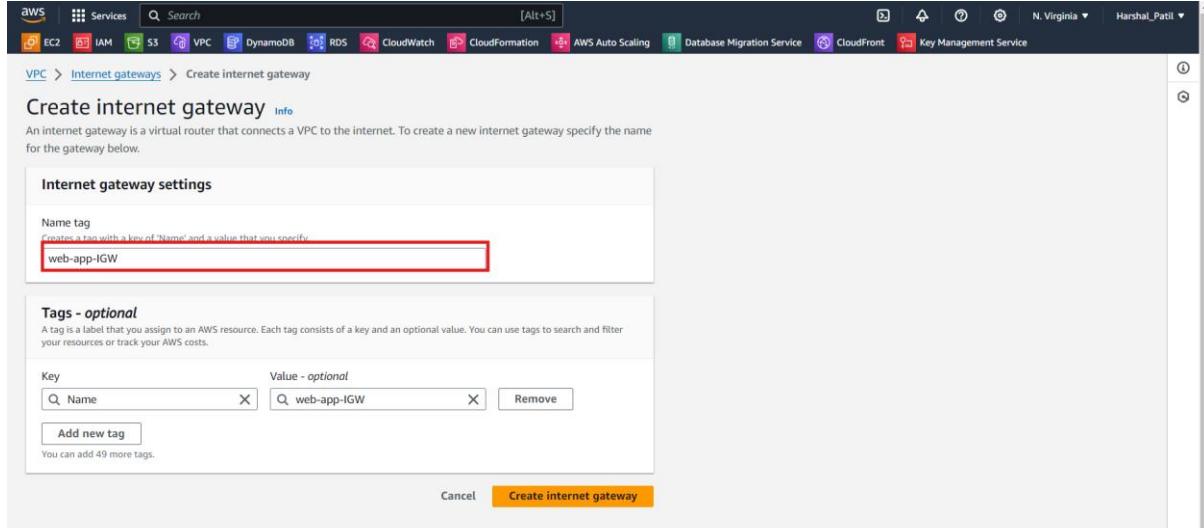
➤ Create Internet Gateway and attach to created VPC



The screenshot shows the AWS VPC dashboard with the 'Internet gateways' section selected. A single Internet gateway is listed:

Name	Internet gateway ID	State	VPC ID	Owner
-	igw-0582b68dad6ff6904	Attached	vpc-004b68edb02c418b6	471112626862

A red box highlights the 'Create internet gateway' button at the top right of the table.



The screenshot shows the 'Create internet gateway' wizard, Step 1: Internet gateway settings. The 'Name tag' field contains 'web-app-IGW', which is highlighted with a red box.

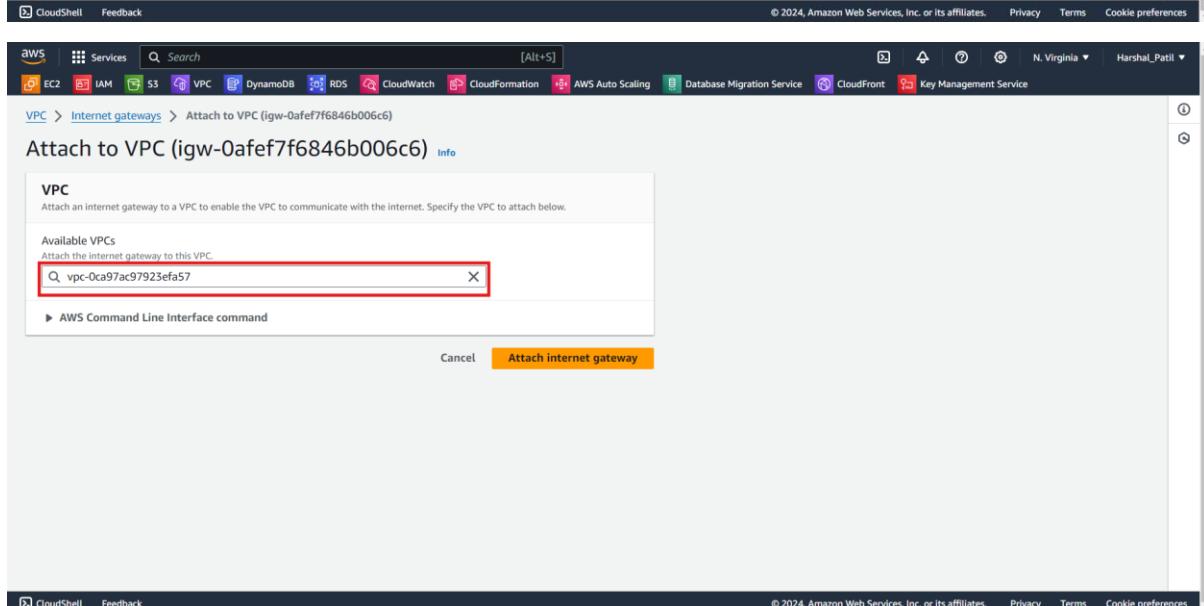
Internet gateway settings

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

Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="web-app-IGW"/>

You can add 49 more tags.



The screenshot shows the 'Attach to VPC' wizard, Step 1: Attach to VPC. The 'Available VPCs' dropdown menu is open, and the entry 'Q_vpc-0ca97ac97923efa57' is highlighted with a red box.

VPC
Attach an Internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

Available VPCs
Attach the Internet gateway to this VPC.

➤ Create 2 NAT Gateways for both Availability zones

The screenshot shows the AWS VPC service dashboard. On the left, there's a navigation pane with various VPC-related options like Subnets, Route tables, Internet gateways, and NAT gateways. The 'NAT gateways' option is selected and highlighted in blue. At the top right, there's a search bar and a 'Create NAT gateway' button, which is highlighted with a red box.

This screenshot shows the 'NAT gateway settings' configuration page. It includes fields for 'Name - optional' (containing 'NAT-GW-AZ1'), 'Subnet' (containing 'subnet-05ec9b87edcb484e2 (Public-Web-Subnet-AZ1)'), 'Connectivity type' (set to 'Public'), and 'Elastic IP allocation ID' (containing 'eipalloc-01ca53736e04f084d'). A red box highlights the 'Allocate Elastic IP' button. Below the form, there's a 'Tags' section with a note about labeling resources.

This screenshot shows the 'NAT gateway settings' configuration page for a second NAT gateway. It has similar fields: 'Name - optional' (containing 'NAT-GW-AZ2'), 'Subnet' (containing 'subnet-07f9cbcac168858d6f (Public-Web-Subnet-AZ2)'), 'Connectivity type' (set to 'Public'), and 'Elastic IP allocation ID' (containing 'eipalloc-090cf6683843ec9ba'). A red box highlights the 'Allocate Elastic IP' button. Like the first page, it includes a 'Tags' section at the bottom.

- Create 3 Route Tables (1 Public & 2 Private), add routes and attach to subnets

- Public Route Table

The screenshot shows the AWS VPC dashboard with the 'Route tables' section selected. The table lists three existing route tables:

Name	Route table ID	Explicit subnet associations	Main	VPC
rtb-0aea1791959697356	-	-	Yes	vpc-004b68edb02c418b6
rtb-05a2d88564b69ca45	-	-	Yes	vpc-0ca97ac97923efa57
-	-	-	-	-

Create route table

Create route table

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

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

VPC
The VPC to use for this route table.
vpc-0ca97ac97923efa57 (webappVPC)

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key Value - optional
Q Name **Q PublicRouteTable** **Remove**
Add new tag

You can add 49 more tags.

Create route table

Route table rtb-0c188049e37eb0692 | PublicRouteTable was created successfully.

rtb-0c188049e37eb0692 / PublicRouteTable

Details

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-0c188049e37eb0692	No	-	-
VPC	Owner ID		
vpc-0ca97ac97923efa57 webappVPC	471112626862		

Routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

Edit routes

Add following Route:

The screenshot shows the AWS VPC Route Tables page. In the 'Edit routes' section, a new route is being added. The destination is '0.0.0.0/0' and the target is 'Internet Gateway'. A red box highlights this row. Below it, there is a list of existing routes. At the bottom of the table, there is a button labeled 'Add route'.

Subnet association: Associate with public subnets

The screenshot shows the AWS VPC Route Tables page. In the 'rtb-0c188049e37eb0692 / PublicRouteTable' section, the 'Subnet associations' tab is selected. There is a button labeled 'Edit subnet associations' with a red box around it.

- Private Route Table

The screenshot shows the AWS VPC Route Tables page. In the 'rtb-052f20ee9e983adea / PrivateRouteTable-AZ1' section, the 'Routes' tab is selected. There is a button labeled 'Edit routes' with a red box around it.

Select NAT Gateway in AZ1

The screenshot shows the 'Edit routes' section of a route table. A new route is being added for destination 0.0.0.0/0, which is being directed to a 'NAT Gateway' target. The target is identified by its ID: nat-0b0b4b69224d1a250. This row is highlighted with a red box.

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	NAT Gateway	-	No

Buttons at the bottom include 'Add route', 'Cancel', 'Preview', and 'Save changes'.

Subnet Association:

The screenshot shows the 'Edit subnet associations' section. A private subnet, 'Private-App-Subnet-AZ1', is selected for association with the route table. This subnet is highlighted with a red box.

Available subnets (1/6)					
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID	
Public-Web-Subnet-AZ2	subnet-07f9cbcac168858d6f	10.0.1.0/24	-	rtb-0c188049e57eb0692 / PublicRouteTable	
Private-App-Subnet-AZ2	subnet-0afb07b016ae01cb9	10.0.3.0/24	-	Main (rtb-05a2d88564b69ca45)	
Private-DB-Subnet-AZ2	subnet-0a8f2b4beda0bbad	10.0.5.0/24	-	Main (rtb-05a2d88564b69ca45)	
Private-DB-Subnet-AZ1	subnet-097178bc26293ff0d	10.0.4.0/24	-	Main (rtb-05a2d88564b69ca45)	
Public-Web-Subnet-AZ1	subnet-05ec9b87edcb484e2	10.0.0.0/24	-	rtb-0c188049e57eb0692 / PublicRouteTable	
Private-App-Subnet-AZ1	subnet-0a3f0c9bd7b2bd059	10.0.2.0/24	-	Main (rtb-05a2d88564b69ca45)	

Buttons at the bottom include 'Cancel' and 'Save associations'.

Same Process for Private Route Table AZ-2

The screenshot shows the 'Route tables (5)' section of the VPC dashboard. A private route table, 'PrivateRouteTable-AZ2', is selected. This table is highlighted with a red box.

Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC
rtb-0aea1791959697356	-	-	-	Yes	vpc-004b686ed02c418b6
rtb-05a2d88564b69ca45	-	-	-	Yes	vpc-0ca97ac97923efaf57
PublicRouteTable	rtb-0c188049e57eb0692	2 subnets	-	No	vpc-0ca97ac97923efaf57
PrivateRouteTable-AZ1	rtb-052f20ee9e983adea	subnet-0a3f0c9bd7b2bd059	-	No	vpc-0ca97ac97923efaf57
PrivateRouteTable-AZ2	rtb-0bf03779c5e15a10e	subnet-0afb07b016ae01...	-	No	vpc-0ca97ac97923efaf57

Buttons at the bottom include 'Create route table'.

➤ Create Security Groups on Each Layer

- Internet Facing Load Balancer Security Group

The screenshot shows the AWS VPC Security Groups creation interface. In the 'Basic details' section, the security group name is 'Internet-Facing-lb-sg' and the description is 'Allows SSH access to developers'. The selected VPC is 'vpc-0ca97ac97923efa57 (webappVPC)'. In the 'Inbound rules' section, there is one rule: Type: HTTP, Protocol: TCP, Port range: 80, Source: Anywhere (0.0.0.0/0). A warning message at the bottom of the page says: '⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.' In the 'Outbound rules' section, the type is 'All traffic', protocol is 'All', port range is 'All', destination is 'Custom', and the source is '0.0.0.0/0'.

- Web Tier instances Security Group

Source Given: Internet Facing LB SG and Our IP for accessing instances

The screenshot shows the AWS VPC Security Groups console. A security group named "WebTier-sg" is being configured. The "Inbound rules" section is highlighted with a red box. It contains two rules:

- Rule 1:** Type: HTTP, Protocol: TCP, Port range: 80, Source: Custom (sg-04e410df73fee4ac), Description: optional.
- Rule 2:** Type: HTTP, Protocol: TCP, Port range: 80, Source: My IP (103.173.241.83/32), Description: optional.

At the bottom of the inbound rules table is a button labeled "Add rule".

- Internal Load Balancer Security Group

Source Given: Web Tier Instances SG

The screenshot shows the AWS VPC Security Groups console. A security group named "Internal-lb-sg" is being configured. The "Inbound rules" section is highlighted with a red box. It contains one rule:

- Rule 1:** Type: HTTP, Protocol: TCP, Port range: 80, Source: Custom (sg-0596682817a20e2), Description: optional.

At the bottom of the inbound rules table is a button labeled "Add rule".

- Private Servers Security Group

Source Given: Internal load balancer security group and our IP

Note: Backend server is running on port 4000

The screenshot shows the AWS VPC Security Groups console. The security group named "PrivateInstance-sg" is being configured. It has two inbound rules:

- A rule for TCP port 4000 from the security group "sg-02a537a87a0432a" (which is itself selected).
- A rule for TCP port 4000 from "My IP" (103.173.241.83/32).

- Database Security Group

Source Given: Private Server Security Group

Note: Here we used MYSQL/Aurora traffic with port 3306

The screenshot shows the AWS VPC Security Groups console. The security group named "DB-sg" is being configured. It has one inbound rule for MySQL/Aurora traffic on port 3306 from the security group "sg-068b42b5c0ed3a9" (which is itself selected).

3. Database Deployment

➤ Create Subnet Groups

The image consists of three vertically stacked screenshots from the AWS RDS console, showing the steps to create a new DB subnet group.

Screenshot 1: Subnet groups list

This screenshot shows the "Subnet groups" list page. A red box highlights the "Create DB subnet group" button at the top right. The table lists one existing subnet group:

Name	Description	Status	VPC
default-vpc-004b68edb02c418b6	Created from the RDS Management Console	Complete	vpc-004b68edb02c418b6

Screenshot 2: Subnet group details

This screenshot shows the "Subnet group details" configuration page. A red box highlights the "Name" field containing "Three-Tier-DB-Subnet-Group". Another red box highlights the "VPC" dropdown menu showing "webappVPC (vpc-0ca97ac97923efafa57)".

Screenshot 3: Add subnets

This screenshot shows the "Add subnets" step. A red box highlights the "Choose an availability zone" dropdown menu showing "us-east-1a" and "us-east-1b". Another red box highlights the "Select subnets" dropdown menu showing two subnets: "subnet-0a8f2b4beda0bbadec (10.0.5.0/24)" and "subnet-097178bc26293ffd0 (10.0.4.0/24)". A note at the bottom says: "For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones."

➤ Database Deployment

Screenshot 1: AWS RDS Dashboard - Databases

The screenshot shows the AWS RDS dashboard under the 'Databases' section. A success message at the top states 'Successfully created Three-Tier-DB-Subnet-Group. View subnet group'. A notification bar below it says 'Introducing Aurora I/O-Optimized' with a link to the Aurora User Guide. A modal window titled 'Consider creating a Blue/Green Deployment to minimize downtime during upgrades' also appears. In the main 'Databases (0)' table, there is a 'Create database' button highlighted with a red box.

Screenshot 2: Choose a database creation method

This screen allows selecting a database creation method. 'Standard create' is selected, while 'Easy create' is shown as an alternative. It also lists engine options: Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), MySQL, and MariaDB.

Screenshot 3: Engine options

Detailed view of engine options. 'Aurora (MySQL Compatible)' is selected. To the right, a sidebar for 'Aurora MySQL-Compatible Edition' provides details about its performance and features.

Screenshot 4: Aurora MySQL-Compatible Edition

This screen shows configuration for the Aurora MySQL-Compatible Edition. It includes a dropdown for 'Engine Version' set to 'Aurora (MySQL 5.7) 2.11.2'. A warning message about the end of standard support on October 31, 2024, is displayed. Another message notes that parallel query is off by default. The sidebar continues to describe the edition's capabilities.

Screenshot 5: Templates

Shows template selection for the database instance. 'Dev/Test' is selected, while 'Production' is another option.

Screenshot 6: Settings

The settings section at the bottom of the configuration page.

Aurora MySQL-Compatible Edition

Aura MySQL is Amazon's enterprise-class MySQL-compatible database.

Aurora MySQL offers:

- Up to five times the throughput of MySQL Community Edition
- Up to 128 TB of autoscaling SSD storage
- Six-way replication across three Availability Zones
- Up to 15 read replicas with replica lag under 10-ms
- Automatic monitoring with failover

DB instance class

Choose the DB instance type that allocates the computational, network, and memory capacity required by planned workloads of this DB instance.

Memory optimized classes
Memory optimized instances accelerate performance for workloads that process large data sets in memory.

Burstable classes
Burstable performance instances provide a baseline level of CPU performance with the ability to burst above the baseline.

Learn more

DB instance classes

Availability & durability

Multi-AZ deployment

- Create an Aurora Replica or Reader node in a different AZ (recommended for scaled availability)
- Creates an Aurora Replica for fast failover and high availability.
- Don't create an Aurora Replica

Connectivity

Compute resource

Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

- Don't connect to an EC2 compute resource
- Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.
- Connect to an EC2 compute resource
- Set up a connection to an EC2 compute resource for this database.

Network type

To use dual-stack mode, make sure that you associate an IPv6 CIDR block with a subnet in the VPC you specify.

- IPv4
- Your resources can communicate only over the IPv4 addressing protocol.
- Dual-stack mode
- Your resources can communicate over IPv4, IPv6, or both.

Virtual private cloud (VPC)

Choose the VPC. The VPC defines the virtual networking environment for this DB cluster.

Virtual private cloud (VPC) Info
Choose the VPC. The VPC defines the virtual networking environment for this DB cluster.

DB subnet group Info
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB cluster can use in the VPC that you selected.

Public access Info

- Yes
RDS assigns a public IP address to the cluster. Amazon EC2 instances and other resources outside of the VPC can connect to your cluster. Resources inside the VPC can also connect to the cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.
- No
RDS doesn't assign a public IP address to the cluster. Only Amazon EC2 instances and other resources inside the VPC can connect to your cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

VPC security group (firewall) Info
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

DB instance class
Choose the DB instance type that allocates the computational, network, and memory capacity required by planned workloads of this DB instance.

Memory optimized classes
Memory optimized instances accelerate performance for workloads that process large data sets in memory.

Burstable classes
Burstable performance instances provide a baseline level of CPU performance with the ability to burst above the baseline.

Learn more [DB instance classes](#)

No
RDS doesn't assign a public IP address to the cluster. Only Amazon EC2 instances and other resources inside the VPC can connect to your cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

VPC security group (firewall) Info
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

Create new
Create new VPC security group

Existing VPC security groups
Choose one or more options

DB-sg

RDS Proxy
RDS Proxy is a fully managed, highly available database proxy that improves application scalability, resiliency, and security.

Create an RDS Proxy Info
RDS automatically creates an IAM role and a Secrets Manager secret for the proxy. RDS Proxy has additional costs. For more information, see [Amazon RDS Proxy pricing](#).

Certificate authority - optional Info
Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.

rds-ca-rsa2048-g1 (default)
Expires: May 26, 2061

If you don't select a certificate authority, RDS chooses one for you.

DB instance class
Choose the DB instance type that allocates the computational, network, and memory capacity required by planned workloads of this DB instance.

Memory optimized classes
Memory optimized instances accelerate performance for workloads that process large data sets in memory.

Burstable classes
Burstable performance instances provide a baseline level of CPU performance with the ability to burst above the baseline.

Learn more [DB instance classes](#)

Add new tag
You can add up to 50 more tags.

Database authentication

Database authentication options Info

- Password authentication**
Authenticates using database passwords.
- Password and IAM database authentication**
Authenticates using the database password and user credentials through AWS IAM users and roles.
- Password and Kerberos authentication**
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

Monitoring

Enable Enhanced Monitoring
Enabling Enhanced Monitoring metrics are useful when you want to see how different processes or threads use the CPU.

Additional configuration

DB instance class
Choose the DB instance type that allocates the computational, network, and memory capacity required by planned workloads of this DB instance.

Memory optimized classes
Memory optimized instances accelerate performance for workloads that process large data sets in memory.

Burstable classes
Burstable performance instances provide a baseline level of CPU performance with the ability to burst above the baseline.

Learn more [DB instance classes](#)

Click on Create Database

Wait for 20 minutes till database gets provisioned

The screenshot shows the AWS RDS console with the 'Databases' tab selected. On the left, a sidebar lists various RDS management options like Dashboard, Databases, and Snapshots. The main area displays a table of databases. A callout box provides information about Aurora MySQL end-of-support and suggests a Blue/Green deployment. The table details three database instances under the cluster 'database-1':

DB identifier	Status	Role	Engine	Region & AZ	Size
database-1	Available	Regional cluster	Aurora MySQL	us-east-1	2 instances
database-1-instance-1	Available	Writer instance	Aurora MySQL	us-east-1a	db.t3.small
database-1-instance-1-us-east-1b	Available	Reader instance	Aurora MySQL	us-east-1b	db.t3.small

4. App Tier Instance Deployment

➤ App Instance Deployment

The screenshots illustrate the steps to launch a new instance:

- Screenshot 1: EC2 Instances Overview**
Shows the EC2 Instances dashboard with a search bar and filters. The "Launch instances" button is highlighted with a red box.
- Screenshot 2: Select an instance**
Shows the "Select an instance" step where users can choose their instance type. A summary panel on the right shows the configuration: 1 instance of t2.micro, Amazon Linux 2023 AMI, 1 volume (~8 GiB), and a new security group. A "Free tier" message is displayed.
- Screenshot 3: AMI Selection**
Shows the "Amazon Machine Image (AMI)" selection screen for Amazon Linux 2023. It details the AMI ID, virtualization type (hvm), and boot mode (uefi-preferred). A "Verified provider" badge is present.

Instance type [Info](#) | [Get advice](#)

Instance type

t2.micro Free tier eligible

Family: t2 - 1 vCPU 1 GiB Memory Current generation: true
 On-Demand Windows base pricing: 0.0162 USD per Hour
 On-Demand SUSE base pricing: 0.0116 USD per Hour
 On-Demand RHEL base pricing: 0.026 USD per Hour
 On-Demand Linux base pricing: 0.0116 USD per Hour

All generations

[Compare instance types](#)

Additional costs apply for AMIs with pre-installed software

Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Proceed without a key pair (Not recommended) Default value [Create new key pair](#)

Network settings [Info](#)

[Edit](#)

Summary

Number of instances [Info](#)
 1

Software Image (AMI)
 Amazon Linux 2023 AMI 2023.5.2... [read more](#)
 ami-0182f573e66f89c85

Virtual server type (instance type)
 t2.micro

Firewall (security group)
 New security group

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro) in the Regions in which you launch the instance.

[Launch instance](#)

VPC - required [Info](#)

vpc-0ca97ac97923efa57 (weappVPC)
 10.0.0.0/16

Subnet [Info](#)

subnet-0a3f0c9bd7b2bd059 Private-App-Subnet-AZ1
 VPC: vpc-0ca97ac97923efa57 Owner: 47112626862 Availability Zone: us-east-1a
 Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.2.0/24

[Create new subnet](#)

Auto-assign public IP [Info](#)

Disable

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

PrivateInstance-sg sg-068b42b5c0ed3a965 [X](#)
 VPC: vpc-0ca97ac97923efa57

Compare security group rules

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

Advanced network configuration

Summary

Number of instances [Info](#)
 1

Software Image (AMI)
 Amazon Linux 2023 AMI 2023.5.2... [read more](#)
 ami-0182f573e66f89c85

Virtual server type (instance type)
 t2.micro

Firewall (security group)
 PrivateInstance-sg

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro) in the Regions in which you launch the instance.

[Launch instance](#)

Advanced details [Info](#)

Domain join directory [Info](#)

Select [Create new directory](#)

IAM instance profile [Info](#)

RoleForEC2toAccessS3
 arn:aws:iam::47112626862:instance-profile/RoleForEC2toAccessS3 [X](#)
[Create new IAM profile](#)

Hostname type [Info](#)

IP name

DNS Hostname [Info](#)

Enable IP name (IPv4 (A record) DNS requests
 Enable resource-based IPv4 (A record) DNS requests
 Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery [Info](#)

Select

Shutdown behavior [Info](#)

Stop

Summary

Number of instances [Info](#)
 1

Software Image (AMI)
 Amazon Linux 2023 AMI 2023.5.2... [read more](#)
 ami-0182f573e66f89c85

Virtual server type (instance type)
 t2.micro

Firewall (security group)
 PrivateInstance-sg

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro) in the Regions in which you launch the instance.

[Launch instance](#)

Click on Launch Instance

➤ Connect to Instance

The screenshot shows the AWS EC2 Instances page. A single instance, 'myAppserver1' (Instance ID: i-03034a654cb7073ea), is listed as 'Running'. The 'Connect' button is highlighted in red.

Instances (1/1) info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
myAppserver1	i-03034a654cb7073ea	Running	t2.micro	Initializing	View alarms +	us-east-1a	-

i-03034a654cb7073ea (myAppserver1)

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

Instance summary

Instance ID i-03034a654cb7073ea (myAppserver1)	Public IPv4 address -	Private IPv4 addresses 10.0.2.31
IPv6 address -	Instance state Running	Public IPv4 DNS -
Hostname type IP name: ip-10-0-2-31.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-2-31.ec2.internal	

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The dialog box shows session options: EC2 Instance Connect, Session Manager (selected), SSH client, and EC2 serial console. It includes a note about Session Manager usage and a 'Connect' button.

Session Manager usage:

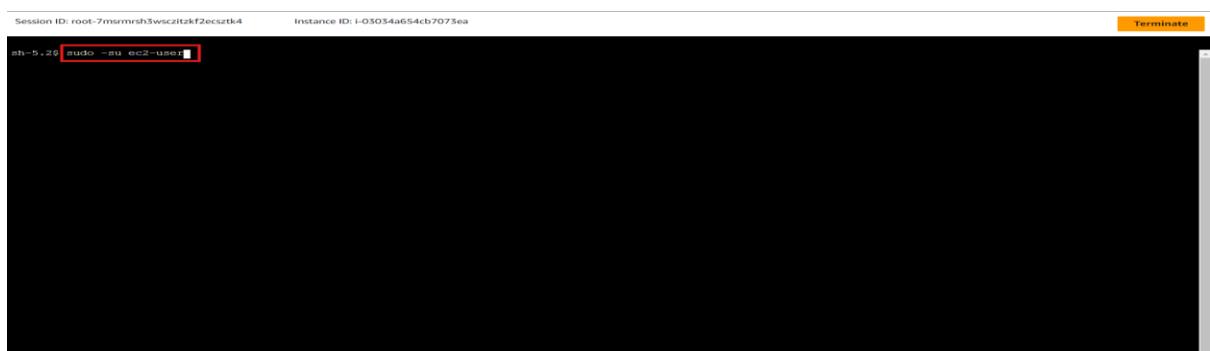
- Connect to your instance without SSH keys, a bastion host, or opening any inbound ports.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager [Preferences](#) page.

Cancel | **Connect**

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

When we first connect to our instance like this, we will be logged in as `ssm-user` which is the default user. Switch to `ec2-user` by executing the following command in the browser terminal:

```
sudo -su ec2-user
```



Check whether we are able to reach internet via NAT Gateway:

ping 8.8.8.8 OR
ping google.com

Session ID: root-7msmrsh3wsczitk2ecstik4 Instance ID: i-05034a654cb7073ea Terminate

```
sh-5.2$ sudo -su ec2-user
[ec2-user@ip-10-0-2-31 bin]$ ping google.com
PING google.com (142.251.167.101) 56(84) bytes of data.
64 bytes from www-in-f101.1e100.net (142.251.167.101): icmp_seq=1 ttl=57 time=2.94 ms
64 bytes from www-in-f101.1e100.net (142.251.167.101): icmp_seq=2 ttl=57 time=2.51 ms
64 bytes from www-in-f101.1e100.net (142.251.167.101): icmp_seq=3 ttl=57 time=2.10 ms
64 bytes from www-in-f101.1e100.net (142.251.167.101): icmp_seq=4 ttl=57 time=2.63 ms
64 bytes from www-in-f101.1e100.net (142.251.167.101): icmp_seq=5 ttl=57 time=2.57 ms
64 bytes from www-in-f101.1e100.net (142.251.167.101): icmp_seq=6 ttl=57 time=2.58 ms
64 bytes from www-in-f101.1e100.net (142.251.167.101): icmp_seq=7 ttl=57 time=2.35 ms
^C
--- google.com ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6008ms
rtt min/avg/max/mdev = 2.099/2.525/2.944/0.239 ms
[ec2-user@ip-10-0-2-31 bin]$
```

➤ Configure Database

```
Session ID: root-7msnmrsh3wscztkf2ecsztka
Instance ID: i-05034a654cb7073ea
Terminate
```

```
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=2 ttl=57 time=2.51 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=3 ttl=57 time=2.10 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=4 ttl=57 time=2.63 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=5 ttl=57 time=2.57 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=6 ttl=57 time=2.58 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=7 ttl=57 time=2.35 ms
^C
-- google.com ping statistics --
7 packets transmitted, 7 received, 0% packet loss, time 6008ms
rtt min/avg/max/mdev = 2.099/2.525/2.944/0.239 ms
[ec2-user@ip-10-0-2-31 bin]$ ^C
[ec2-user@ip-10-0-2-31 bin]$ ^C
[ec2-user@ip-10-0-2-31 bin]$ sudo yum install mysql
Last metadata expiration check: 0:06:53 ago on Mon Sep 9 16:33:42 2024.
No match for argument: mysql
Error: Unable to find a match: mysql
[ec2-user@ip-10-0-2-31 bin]$ sudo wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm^C
[ec2-user@ip-10-0-2-31 bin]$ sudo wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
--2024-09-09 16:46:06-- https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
Resolving dev.mysql.com (dev.mysql.com)... 23.218.131.64, 2600:1408:c400:188c::2e31
Connecting to dev.mysql.com (dev.mysql.com)|23.218.131.64|:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: https://repo.mysql.com/mysql57-community-release-el7-11.noarch.rpm (following)
--2024-09-09 16:46:07-- https://repo.mysql.com/mysql57-community-release-el7-11.noarch.rpm
Resolving repo.mysql.com (repo.mysql.com)... 23.62.123.104, 2600:1408:ec00:888::1:d68, 2600:1408:ec00:881::1:d68
Connecting to repo.mysql.com (repo.mysql.com)|23.62.123.104|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 25680 (25K) [application/x-redhat-package-manager]
Saving to: 'mysql57-community-release-el7-11.noarch.rpm'

mysql57-community-release-el7-11.noarch.rpm 100%[=====] 25.08K --.-KB/s in 0.007s

2024-09-09 16:46:07 (3.42 MB/s) - 'mysql57-community-release-el7-11.noarch.rpm' saved [25680/25680]

[ec2-user@ip-10-0-2-31 bin]$ 
```

```
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=2 ttl=57 time=2.51 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=3 ttl=57 time=2.10 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=4 ttl=57 time=2.63 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=5 ttl=57 time=2.57 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=6 ttl=57 time=2.58 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=7 ttl=57 time=2.35 ms
^C
-- google.com ping statistics --
7 packets transmitted, 7 received, 0% packet loss, time 6008ms
rtt min/avg/max/mdev = 2.099/2.525/2.944/0.239 ms
[ec2-user@ip-10-0-2-31 bin]$ ^C
[ec2-user@ip-10-0-2-31 bin]$ ^C
[ec2-user@ip-10-0-2-31 bin]$ sudo yum install mysql
Last metadata expiration check: 0:06:53 ago on Mon Sep 9 16:33:42 2024.
No match for argument: mysql
Error: Unable to find a match: mysql
[ec2-user@ip-10-0-2-31 bin]$ sudo wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm^C
[ec2-user@ip-10-0-2-31 bin]$ sudo wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
--2024-09-09 16:46:06-- https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
Resolving dev.mysql.com (dev.mysql.com)... 23.218.131.64, 2600:1408:c400:188c::2e31
Connecting to dev.mysql.com (dev.mysql.com)|23.218.131.64|:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: https://repo.mysql.com/mysql57-community-release-el7-11.noarch.rpm (following)
--2024-09-09 16:46:07-- https://repo.mysql.com/mysql57-community-release-el7-11.noarch.rpm
Resolving repo.mysql.com (repo.mysql.com)... 23.62.123.104, 2600:1408:ec00:888::1:d68, 2600:1408:ec00:881::1:d68
Connecting to repo.mysql.com (repo.mysql.com)|23.62.123.104|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 25680 (25K) [application/x-redhat-package-manager]
Saving to: 'mysql57-community-release-el7-11.noarch.rpm'

mysql57-community-release-el7-11.noarch.rpm 100%[=====] 25.08K --.-KB/s in 0.007s

2024-09-09 16:46:07 (3.42 MB/s) - 'mysql57-community-release-el7-11.noarch.rpm' saved [25680/25680]

[ec2-user@ip-10-0-2-31 bin]$ sudo rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2022 
```

```
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=3 ttl=57 time=2.10 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=4 ttl=57 time=2.63 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=5 ttl=57 time=2.57 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=6 ttl=57 time=2.58 ms
64 bytes from xx-in-f101.1e100.net (142.251.167.101): icmp_seq=7 ttl=57 time=2.35 ms
^C
-- google.com ping statistics --
7 packets transmitted, 7 received, 0% packet loss, time 6008ms
rtt min/avg/max/mdev = 2.099/2.525/2.944/0.239 ms
[ec2-user@ip-10-0-2-31 bin]$ ^C
[ec2-user@ip-10-0-2-31 bin]$ ^C
[ec2-user@ip-10-0-2-31 bin]$ sudo yum install mysql
Last metadata expiration check: 0:06:53 ago on Mon Sep 9 16:33:42 2024.
No match for argument: mysql
Error: Unable to find a match: mysql
[ec2-user@ip-10-0-2-31 bin]$ sudo wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm^C
[ec2-user@ip-10-0-2-31 bin]$ sudo wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
--2024-09-09 16:46:06-- https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
Resolving dev.mysql.com (dev.mysql.com)... 23.218.131.64, 2600:1408:c400:188c::2e31
Connecting to dev.mysql.com (dev.mysql.com)|23.218.131.64|:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: https://repo.mysql.com/mysql57-community-release-el7-11.noarch.rpm (following)
--2024-09-09 16:46:07-- https://repo.mysql.com/mysql57-community-release-el7-11.noarch.rpm
Resolving repo.mysql.com (repo.mysql.com)... 23.62.123.104, 2600:1408:ec00:888::1:d68, 2600:1408:ec00:881::1:d68
Connecting to repo.mysql.com (repo.mysql.com)|23.62.123.104|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 25680 (25K) [application/x-redhat-package-manager]
Saving to: 'mysql57-community-release-el7-11.noarch.rpm'

mysql57-community-release-el7-11.noarch.rpm 100%[=====] 25.08K --.-KB/s in 0.007s

2024-09-09 16:46:07 (3.42 MB/s) - 'mysql57-community-release-el7-11.noarch.rpm' saved [25680/25680]

[ec2-user@ip-10-0-2-31 bin]$ sudo rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2022
[ec2-user@ip-10-0-2-31 bin]$ sudo yum install https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm 
```

Use below command to download MySQL CLI

```
Verifying      : mysql57-community-release-el7-11.noarch          1/1
Installed:
  mysql57-community-release-el7-11.noarch

Complete!
[ec2-user@ip-10-0-2-31 bin]$ sudo yum install mysql -y

MySQL Connectors Community
MySQL Tools Community
MySQL 5.7 Community Server
Dependencies resolved.

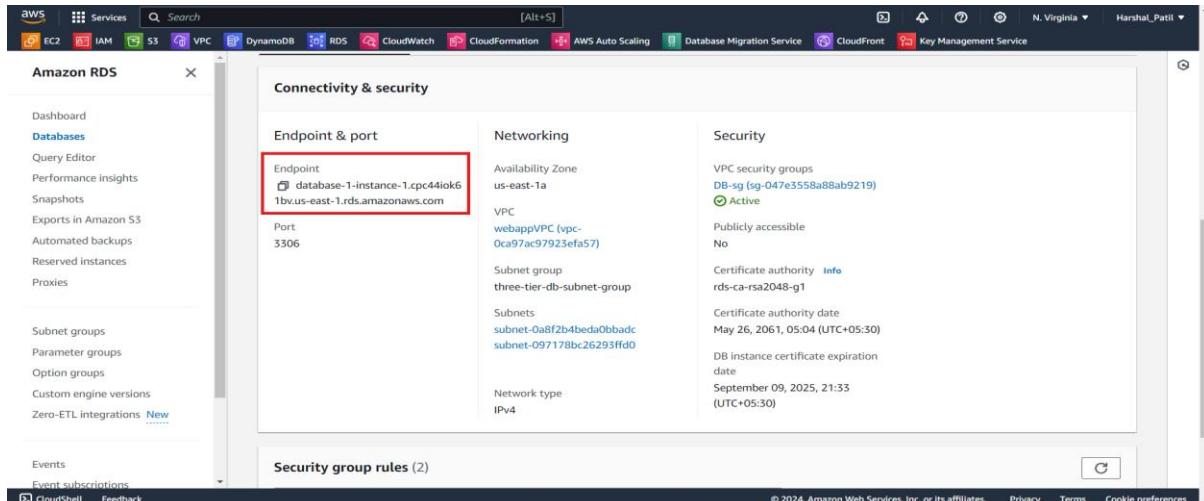
Transaction Summary
Install 4 Packages

Total download size: 35 M
Installed size: 135 M
Downloading Packages:
(1/4): mysql-community-common-5.7.44-1.el7.x86_64.rpm           5.5 MB/s | 313 kB   00:00
(2/4): ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64.rpm 2.7 MB/s | 323 kB   00:00
(3/4): mysql-community-libs-5.7.44-1.el7.x86_64.rpm             35 MB/s | 3.0 MB   00:00
(4/4): mysql-community-client-5.7.44-1.el7.x86_64.rpm            62 MB/s | 31 MB    00:00

Total
Running transaction check
61 MB/s | 35 MB   00:00
```

Initiate DB connection with Aurora RDS writer endpoint. In the following command, replace the RDS writer endpoint and the username

```
mysql -h CHANGE-TO-YOUR-RDS-ENDPOINT -u CHANGE-TO-USER-NAME -p
```



The screenshot shows the AWS Management Console with the RDS service selected. On the left, the 'Amazon RDS' navigation pane is visible with various options like Dashboard, Databases, Query Editor, etc. The main content area is titled 'Connectivity & security'. It displays information about the endpoint, networking (Availability Zone: us-east-1a, VPC: webappVPC, Subnet group: three-tier-db-subnet-group), and security (Security groups: DB-sg (sg-047e3558a88ab9219), Certificate authority: rds-ca-rsa2048-g1). Below this, there's a section for 'Security group rules'.

```
Install 4 Packages

Total download size: 35 M
Installed size: 135 M
Downloading Packages:
(1/4): mysql-community-common-5.7.44-1.el7.x86_64.rpm           5.5 MB/s | 313 kB   00:00
(2/4): ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64.rpm 2.7 MB/s | 323 kB   00:00
(3/4): mysql-community-libs-5.7.44-1.el7.x86_64.rpm             35 MB/s | 3.0 MB   00:00
(4/4): mysql-community-client-5.7.44-1.el7.x86_64.rpm            62 MB/s | 31 MB    00:00

Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing :
  Installing   : mysql-community-common-5.7.44-1.el7.x86_64          1/1
  Installing   : mysql-community-libs-5.7.44-1.el7.x86_64          1/4
  Running script: mysql-community-libs-5.7.44-1.el7.x86_64          2/4
  Installing   : ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64 2/4
  Installing   : mysql-community-client-5.7.44-1.el7.x86_64          3/4
  Running script: mysql-community-client-5.7.44-1.el7.x86_64          4/4
  Verifying    : ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64 1/4
  Verifying    : mysql-community-client-5.7.44-1.el7.x86_64          2/4
  Verifying    : mysql-community-common-5.7.44-1.el7.x86_64          3/4
  Verifying    : mysql-community-libs-5.7.44-1.el7.x86_64          4/4

Installed:
  mysql-community-client-5.7.44-1.el7.x86_64                      mysql-community-common-5.7.44-1.el7.x86_64
  ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64          mysql-community-libs-5.7.44-1.el7.x86_64

Complete!
[ec2-user@ip-10-0-2-31 bin]$ mysql -h database-1-instance-1.cpc44iok61bv.us-east-1.rds.amazonaws.com -u admin -p
```

Enter Password and log in to database

```
Running transaction test.
Transaction test succeeded.
Running transaction
Preparing      :
Installing   : mysql-community-common-5.7.44-1.el7.x86_64          1/1
Installing   : mysql-community-libs-5.7.44-1.el7.x86_64          1/4
Installing   : mysql-community-libs-5.7.44-1.el7.x86_64          2/4
Running scriptlet: mysql-community-libs-5.7.44-1.el7.x86_64          2/4
Installing   : ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64  3/4
Installing   : mysql-community-client-5.7.44-1.el7.x86_64          4/4
Running scriptlet: mysql-community-client-5.7.44-1.el7.x86_64          4/4
Verifying     : mysql-community-common-5.7.44-1.el7.x86_64          4/4
Verifying     : mysql-community-client-5.7.44-1.el7.x86_64          1/4
Verifying     : mysql-community-common-5.7.44-1.el7.x86_64          2/4
Verifying     : mysql-community-libs-5.7.44-1.el7.x86_64          3/4
Verifying     : mysql-community-client-5.7.44-1.el7.x86_64          4/4

Installed:
mysql-community-client-5.7.44-1.el7.x86_64          mysql-community-common-5.7.44-1.el7.x86_64          mysql-community-libs-5.7.44-1.el7.x86_64
ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64

Complete!
[ec2-user@ip-10-0-2-31 bin]$ mysql -h database-1-instance-1.cpc44iok6lbv.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 295
Server version: 5.7.12 MySQL Community Server (GPL)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

Create database with name webappdb

```
Preparing      :
Installing   : mysql-community-common-5.7.44-1.el7.x86_64          1/1
Installing   : mysql-community-libs-5.7.44-1.el7.x86_64          1/4
Installing   : mysql-community-libs-5.7.44-1.el7.x86_64          2/4
Running scriptlet: mysql-community-libs-5.7.44-1.el7.x86_64          2/4
Installing   : ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64  3/4
Installing   : mysql-community-client-5.7.44-1.el7.x86_64          4/4
Running scriptlet: mysql-community-client-5.7.44-1.el7.x86_64          4/4
Verifying     : ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64  1/4
Verifying     : mysql-community-client-5.7.44-1.el7.x86_64          2/4
Verifying     : mysql-community-common-5.7.44-1.el7.x86_64          3/4
Verifying     : mysql-community-libs-5.7.44-1.el7.x86_64          4/4

Installed:
mysql-community-client-5.7.44-1.el7.x86_64          mysql-community-common-5.7.44-1.el7.x86_64          mysql-community-libs-5.7.44-1.el7.x86_64
ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64

Complete!
[ec2-user@ip-10-0-2-31 bin]$ mysql -h database-1-instance-1.cpc44iok6lbv.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 295
Server version: 5.7.12 MySQL Community Server (GPL)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create DATABASE webappdb;
Query OK, 1 row affected (0.01 sec)

mysql> 
```

```
Installed:
mysql-community-client-5.7.44-1.el7.x86_64          mysql-community-common-5.7.44-1.el7.x86_64          mysql-community-libs-5.7.44-1.el7.x86_64
ncurses-compat-libs-6.2-4.20200222.amzn2023.0.6.x86_64

Complete!
[ec2-user@ip-10-0-2-31 bin]$ mysql -h database-1-instance-1.cpc44iok6lbv.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 295
Server version: 5.7.12 MySQL Community Server (GPL)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create DATABASE webappdb;
Query OK, 1 row affected (0.01 sec)

mysql> show databases;
+-----+-----+
| Database |
+-----+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
| webappdb |
+-----+-----+
5 rows in set (0.00 sec)

mysql> 
```

Navigate to webappdb and create **transactions** table and insert data into that table by using below commands

```
| information_schema |
| mysql |
| performance_schema |
| sys |
| webappdb |
+-----+
5 rows in set (0.00 sec)

mysql> use webappdb
Database changed
mysql> CREATE TABLE IF NOT EXISTS transactions(id INT NOT NULL
-> AUTO_INCREMENT, amount DECIMAL(10,2), description
-> VARCHAR(100), PRIMARY KEY(id));
Query OK, 0 rows affected (0.06 sec)

mysql> SHOW TABLES;
+-----+
| Tables_in_webappdb |
+-----+
| transactions |
+-----+
1 row in set (0.00 sec)

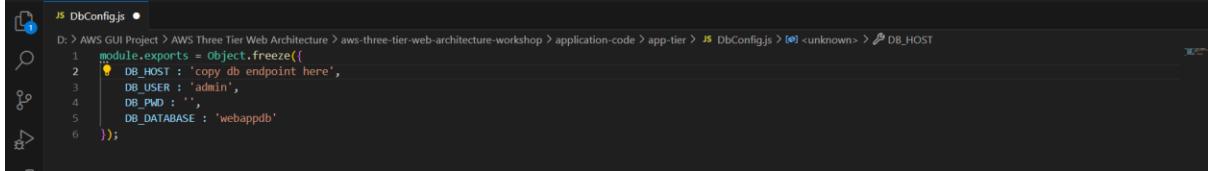
mysql> INSERT INTO transactions (amount,description) VALUES ('400','groceries');
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM transactions;
+----+-----+-----+
| id | amount | description |
+----+-----+-----+
| 1  | 400.00 | groceries |
+----+-----+-----+
1 row in set (0.01 sec)

mysql>
```

➤ Configure App Instance

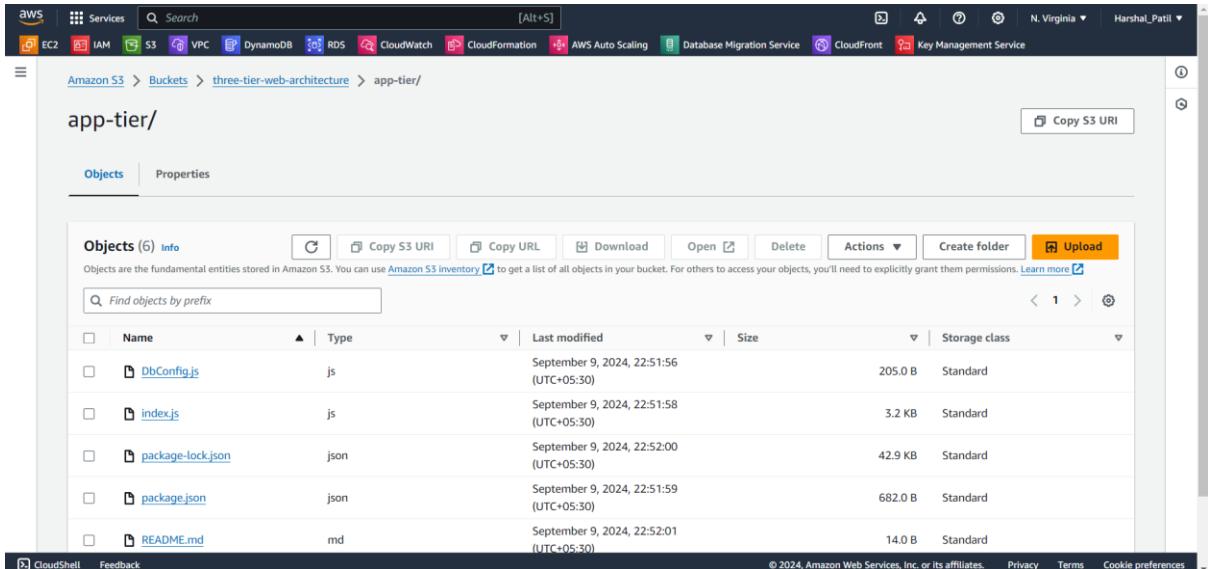
First we have update our database credentials for the app tier. Open **application-code/app-tier/DbConfig.js** and make the following changes



```
JS DbConfig.js ●
D: AWS GUI Project > AWS Three Tier Web Architecture > aws-three-tier-web-architecture-workshop > application-code > app-tier > JS DbConfig.js > [o] <unknown> > DB_HOST
1 module.exports = Object.freeze({
2   DB_HOST : 'copy db endpoint here',
3   DB_USER : 'admin',
4   DB_PWD : '',
5   DB_DATABASE : 'webappdb'
6 });

```

Upload the app-tier folder to the S3 bucket

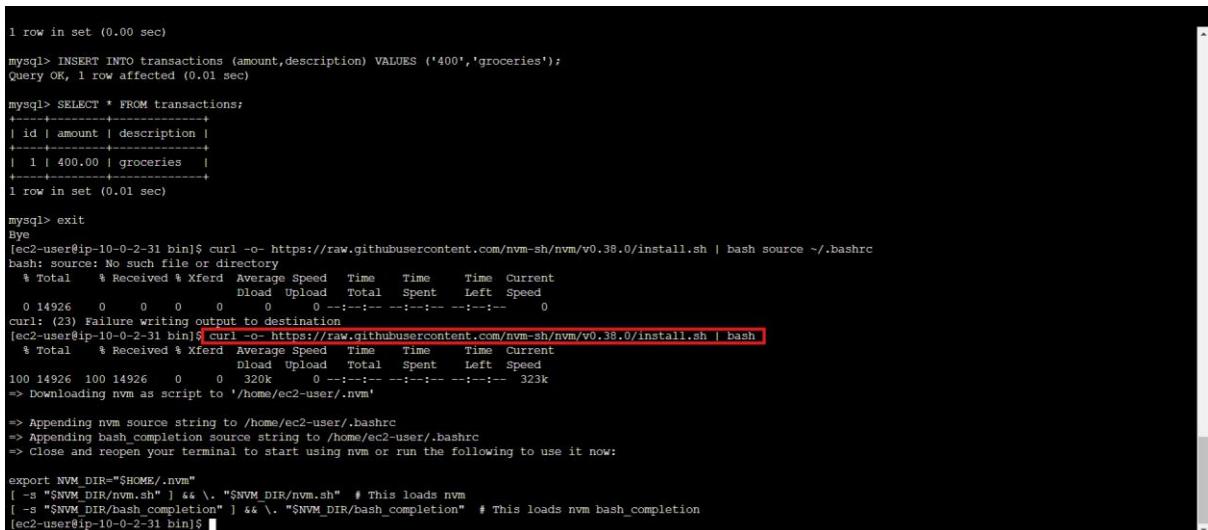


The screenshot shows the AWS S3 console with the path `Amazon S3 > Buckets > three-tier-web-architecture > app-tier/`. The `Objects` tab is selected, displaying a list of files:

Name	Type	Last modified	Size	Storage class
DbConfig.js	js	September 9, 2024, 22:51:56 (UTC+05:30)	205.0 B	Standard
index.js	js	September 9, 2024, 22:51:58 (UTC+05:30)	3.2 KB	Standard
package-lock.json	json	September 9, 2024, 22:52:00 (UTC+05:30)	42.9 KB	Standard
package.json	json	September 9, 2024, 22:51:59 (UTC+05:30)	682.0 B	Standard
README.md	md	September 9, 2024, 22:52:01 (UTC+05:30)	14.0 B	Standard

Go back to SSM session and run following commands which are need to install necessary components to run backend application

- `curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash`
- `source ~/.bashrc`



```
1 row in set (0.00 sec)

mysql> INSERT INTO transactions (amount,description) VALUES ('400','groceries');
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM transactions;
+----+-----+-----+
| id | amount | description |
+----+-----+-----+
| 1  | 400.00 | groceries  |
+----+-----+-----+
1 row in set (0.01 sec)

mysql> exit
Bye
[ec2-user@ip-10-0-2-31 bin]$ curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash source ~/.bashrc
bash: source: No such file or directory
  % Total    % Received % Xferd  Average Speed   Time   Time  Current
          Dload  Upload Total Spent   Left Speed
0 14926     0      0      0      0  --::-- --::-- 0
curl: (23) Failure writing output to destination
[ec2-user@ip-10-0-2-31 bin]$ curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash
  % Total    % Received % Xferd  Average Speed   Time   Time  Current
          Dload  Upload Total Spent   Left Speed
100 14926  100 14926     0      0  320k  --::--  --::-- 323k
=> Downloading nvm as script to '/home/ec2-user/.nvm'

=> Appending nvm source string to /home/ec2-user/.bashrc
=> Appending bash_completion source string to /home/ec2-user/.bashrc
=> Close and reopen your terminal to start using nvm or run the following to use it now:

export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && . "$NVM_DIR/nvm.sh" # This loads nvm
[ -s "$NVM_DIR/bash_completion" ] && . "$NVM_DIR/bash_completion" # This loads nvm bash_completion
[ec2-user@ip-10-0-2-31 bin]$
```

```

mysql> INSERT INTO transactions (amount,description) VALUES ('400','groceries');
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM transactions;
+----+-----+-----+
| id | amount | description |
+----+-----+-----+
| 1  | 400.00 | groceries |
+----+-----+-----+
1 row in set (0.01 sec)

mysql> exit
Bye
[ec2-user@ip-10-0-2-31 bin]$ curl -o https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash source ~/.bashrc
bash: source: No such file or directory
  % Total    % Received % Xferd  Average Speed   Time   Time Current
          Dload  Upload Total Spent   Left Speed
0 14926    0     0    0      0  --::-- --::-- 0
curl: (23) Failure writing output to destination
[ec2-user@ip-10-0-2-31 bin]$ curl -o https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash
  % Total    % Received % Xferd  Average Speed   Time   Time Current
          Dload  Upload Total Spent   Left Speed
100 14926  100 14926    0      0  --::-- --::-- 323k
=> Downloading nvm as script to '/home/ec2-user/.nvm'
=> Appending nvm source string to '/home/ec2-user/.bashrc'
=> Appending bash completion source string to '/home/ec2-user/.bashrc'
=> Close and reopen your terminal to start using nvm or run the following to use it now:

export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \."$NVM_DIR/nvm.sh" # This loads nvm
[ -s "$NVM_DIR/bash_completion" ] && \."$NVM_DIR/bash_completion" # This loads nvm bash_completion
[ec2-user@ip-10-0-2-31 bin]$ source ~/.bashrc
[ec2-user@ip-10-0-2-31 bin]$

```

Install a compatible version of Node.js and make sure it's being used

```
[ec2-user@ip-10-0-2-31 bin]$ nvm install 16
```

```

[ec2-user@ip-10-0-2-31 bin]$ nvm install 16
Downloading and installing node v16.20.2...
Downloaded https://nodejs.org/dist/v16.20.2/node-v16.20.2-linux-x64.tar.xz...
#####
Computing checksum with sha256sum
Checksums matched!
Now using node v16.20.2 (npm v8.19.4)
Creating default alias: default -> v16.20.2
[ec2-user@ip-10-0-2-31 bin]$ nvm use 16
Now using node v16.20.2 (npm v8.19.4)
[ec2-user@ip-10-0-2-31 bin]$

```

PM2 is a daemon process manager that will keep our node.js app running when we exit the instance or if it is rebooted. Install that as well.

```
[ec2-user@ip-10-0-2-31 bin]$ nvm install 16
Downloading and installing node v16.20.2...
Downloading https://nodejs.org/dist/v16.20.2/node-v16.20.2-linux-x64.tar.xz...
#####
Computing checksum with sha256sum
Checksums matched!
Now using node v16.20.2 (npm v8.19.4)
Creating default alias: default -> 16 (-> v16.20.2)
[ec2-user@ip-10-0-2-31 bin]$ nvm use 16
Now using node v16.20.2 (npm v8.19.4)
[ec2-user@ip-10-0-2-31 bin]$ npm install -g pm2
added 138 packages, and audited 139 packages in 6s

13 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice New major version of npm available! 8.19.4 -> 10.8.3
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.3
npm notice Run npm install -g npm@10.8.3 to update!
npm notice
[ec2-user@ip-10-0-2-31 bin]$
```

Now we need to download our code from our s3 buckets into our instance.

```
[ec2-user@ip-10-0-2-31 bin]$ nvm install 16
Downloading and installing node v16.20.2...
Downloading https://nodejs.org/dist/v16.20.2/node-v16.20.2-linux-x64.tar.xz...
#####
Computing checksum with sha256sum
Checksums matched!
Now using node v16.20.2 (npm v8.19.4)
Creating default alias: default -> 16 (-> v16.20.2)
[ec2-user@ip-10-0-2-31 bin]$ nvm use 16
Now using node v16.20.2 (npm v8.19.4)
[ec2-user@ip-10-0-2-31 bin]$ npm install -g pm2
added 138 packages, and audited 139 packages in 6s

13 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice New major version of npm available! 8.19.4 -> 10.8.3
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.3
npm notice Run npm install -g npm@10.8.3 to update!
npm notice
[ec2-user@ip-10-0-2-31 bin]$ cd ..
[ec2-user@ip-10-0-2-31 ~]$ aws s3 cp s3://three-tier-web-architecture/app-tier/ app-tier --recursive
download: s3://three-tier-web-architecture/app-tier/DBConfig.js to app-tier/DBConfig.js
download: s3://three-tier-web-architecture/app-tier/package.json to app-tier/package.json
download: s3://three-tier-web-architecture/app-tier/README.md to app-tier/README.md
download: s3://three-tier-web-architecture/app-tier/TransactionService.js to app-tier/TransactionService.js
download: s3://three-tier-web-architecture/app-tier/package-lock.json to app-tier/package-lock.json
download: s3://three-tier-web-architecture/app-tier/index.js to app-tier/index.js
[ec2-user@ip-10-0-2-31 ~]$
```

Navigate to the app directory, install dependencies, and start the app with pm2.

```
[ec2-user@ip-10-0-2-31 app-tier]$ cd ~/app-tier
[ec2-user@ip-10-0-2-31 app-tier]$ npm install
added 68 packages, and audited 69 packages in 1s

2 packages are looking for funding
  run `npm fund` for details

3 high severity vulnerabilities

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
[ec2-user@ip-10-0-2-31 app-tier]$ pm2 start index.js
-----

Runtime Edition
PM2 is a Production Process Manager for Node.js applications
with a built-in Load Balancer.
Start and daemonize any application:
```

To make sure the app is running correctly run the following:

```
[ec2-user@ip-10-0-2-31 app-tier]$ pm2 list
[ec2-user@ip-10-0-2-31 app-tier]$ pm2 logs
[ec2-user@ip-10-0-2-31 app-tier]$ pm2 save
[ec2-user@ip-10-0-2-31 app-tier]$ pm2 start index
```

id	name	namespace	version	mode	pid	uptime	⌚	status	cpu	mem	user	watching
0	index	default	1.0.0	[redacted]	28344	0s	0	online	0%	25.4mb	ec2-user	disabled

id	name	namespace	version	mode	pid	uptime	⌚	status	cpu	mem	user	watching
0	index	default	1.0.0	[redacted]	28344	57s	0	online	0%	52.9mb	ec2-user	disabled

Right now, pm2 is just making sure our app stays running when we leave the SSM session. However, if the server is interrupted for some reason, we still want the app to start and keep running. This is also important for the AMI we will create. After PM2 startup command to setup the startup script run long command highlighted in red box.

```
[ec2-user@ip-10-0-2-31 app-tier]$ pm2 startup
[PM2] Init System found: systemd
[PM2] To setup the Startup Script, copy/paste the following command:
sudo env PATH=$PATH:/home/ec2-user/.nvm/versions/node/v16.20.2/bin /home/ec2-user/.nvm/versions/node/v16.20.2/lib/node_modules/pm2/bin/pm2 startup systemd -u ec2-user --hp /home/ec2-user
[ec2-user@ip-10-0-2-31 app-tier]$ sudo env PATH=$PATH:/home/ec2-user/.nvm/versions/node/v16.20.2/bin /home/ec2-user/.nvm/versions/node/v16.20.2/lib/node_modules/pm2/bin/pm2 startup systemd -u ec2-user --hp /home/ec2-user
-----  

Runtime Edition  

PM2 is a Production Process Manager for Node.js applications
with a built-in Load Balancer.
```

After you run it, save the current list of node processes with the following command:

```
[Install]
WantedBy=multi-user.target

[Target path
/etc/systemd/system/pm2-ec2-user.service
Command list
[+] systemctl enable pm2-ec2-user
[PM2] Enabling init configuration in /etc/systemd/system/pm2-ec2-user.service
[PM2] Making script booting at startup...
[PM2] [+] Executing: systemctl enable pm2-ec2-user...
Created symlink /etc/systemd/system/multi-user.target.wants/pm2-ec2-user.service → /etc/systemd/system/pm2-ec2-user.service.
[PM2] [+] Command successfully executed.
+-----+
[PM2] Freeze a process list on reboot via:
$ pm2 save

[PM2] Remove init script via:
$ pm2 unstartup system
[ec2-user@ip-10-0-2-31 app-tier]$ pm2 save
[PM2] Saving current process list...
[PM2] Successfully saved in /home/ec2-user/.pm2/dump.pm2
[ec2-user@ip-10-0-2-31 app-tier]$
```

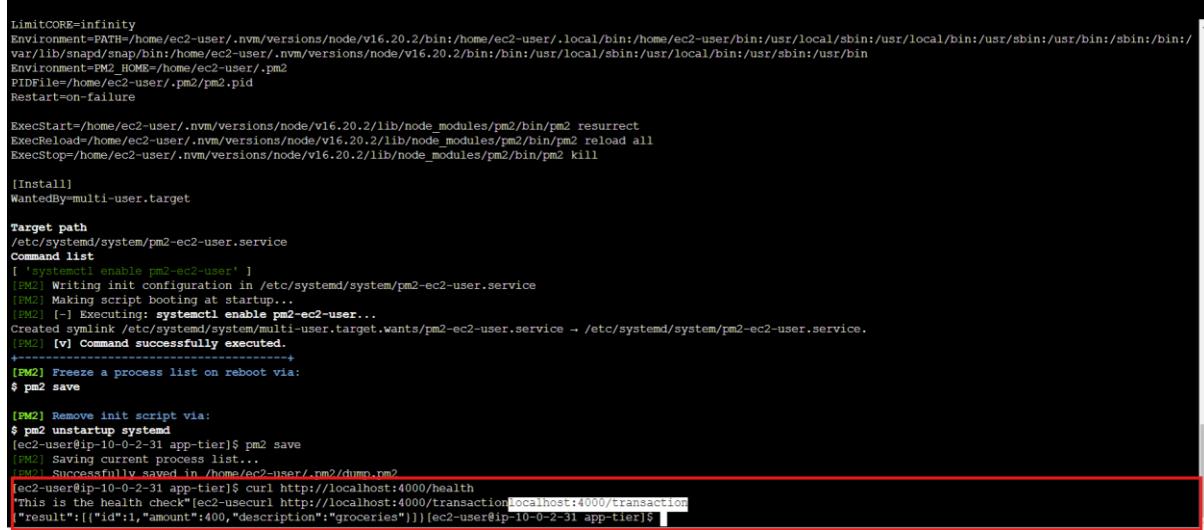
➤ Test App Tier

To hit out health check endpoint, run below command. This is our simple health check endpoint that tells us if the app is simply running.

```
curl http://localhost:4000/health
```

Next, test your database connection

```
curl http://localhost:4000/transaction
```



A terminal session showing the configuration of the pm2 process manager and the execution of curl commands. The session starts with the configuration of a pm2 service, followed by enabling it, saving the current process list, and finally executing a curl request to the health check endpoint. The transaction response is highlighted with a red box.

```
LimitCORE=infinity
Environment=PATH=/home/ec2-user/.nvm/versions/node/v16.20.2/bin:/home/ec2-user/.local/bin:/home/ec2-user/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/var/lib/snapd/snap/bin:/home/ec2-user/.nvm/versions/node/v16.20.2/bin:/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin
Environment=PM2_HOME=/home/ec2-user/.pm2
PIDFile=/home/ec2-user/.pm2/pm2.pid
Restart-on-failure

ExecStart=/home/ec2-user/.nvm/versions/node/v16.20.2/lib/node_modules/pm2/bin/pm2 resurrect
ExecReload=/home/ec2-user/.nvm/versions/node/v16.20.2/lib/node_modules/pm2/bin/pm2 reload all
ExecStop=/home/ec2-user/.nvm/versions/node/v16.20.2/lib/node_modules/pm2/bin/pm2 kill

[Install]
WantedBy=multi-user.target

[Service]
Target path
/etc/systemd/system/pm2-ec2-user.service
Command list
[ 'systemctl enable pm2-ec2-user' ]
[EC2] Writing init configuration in /etc/systemd/system/pm2-ec2-user.service
[EC2] Making script booting at startup...
[EC2] (-) Executing: systemctl enable pm2-ec2-user...
Created symlink /etc/systemd/system/multi-user.target.wants/pm2-ec2-user.service → /etc/systemd/system/pm2-ec2-user.service.
[EC2] [V] Command successfully executed.
+-----+
[PM2] Freeze a process list on reboot via:
$ pm2 save

[PM2] Remove init script via:
$ pm2 unstartup systemd
[ec2-user@ip-10-0-2-31 app-tier]$ pm2 save
[PM2] Saving current process list...
[PM2] Successfully saved in /home/ec2-user/.pm2/dump.pm2
[ec2-user@ip-10-0-2-31 app-tier]$ curl http://localhost:4000/health
This is the health check![ec2-user@ip-10-0-2-31 app-tier]$ curl http://localhost:4000/transaction
"result":[{"id":1,"amount":400,"description":"groceries"}])[ec2-user@ip-10-0-2-31 app-tier]$
```

This sets up app tier layer.

5. Internal Load Balancer and Auto Scaling Group

➤ App Tier AMI

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with sections like EC2 Dashboard, EC2 Global View, Events, and Instances. Under Instances, there are sub-sections for Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations. Below that is the Images section with AMIs and AMI Catalog. At the bottom of the sidebar are CloudShell and Feedback links.

The main content area displays a table titled "Instances (1/1) Info". It shows one instance named "myAppserver1" with the ID "i-03034a654cb7073ea". The instance is listed as "Running" with an "t2.micro" instance type and "2/2 checks passed" status. A context menu is open over this instance, with the "Actions" dropdown expanded. The "Create image" option is highlighted with a red box. Other options in the Actions menu include Connect, View details, Manage instance state, Instance settings, Networking, Security, Image and templates, and Monitor and troubleshoot.

Below the table, a detailed view for "i-03034a654cb7073ea (myAppserver1)" is shown. It includes tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The Details tab is selected. Under Instance summary, it shows the Instance ID (i-03034a654cb7073ea), Public IPv4 address (10.0.2.31), Instance state (Running), Private IP DNS name (ip-10-0-2-31.ec2.internal), and Public IPv4 DNS (ip-10-0-2-31.ec2.internal).

At the bottom right of the page, there are links for © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

➤ Target Group

Screenshot of the AWS EC2 Target Groups page:

The page shows a table with columns: Name, ARN, Port, Protocol, Target type, and Load balancer. A message at the top states: "No target groups You don't have any target groups in us-east-1". A prominent orange button labeled "Create target group" is located in the top right corner.

Below the table, a message says: "0 target groups selected Select a target group above." A "Create target group" button is also present here.

Left sidebar navigation includes: Instances, Images, and Elastic Block Store.

Bottom footer: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the "Step 2 Register targets" configuration page:

The title is "Basic configuration". It says: "Settings in this section can't be changed after the target group is created."

Choose a target type

- Instances
 - Supports load balancing to instances within a specific VPC.
 - Facilitates the use of Amazon EC2 Auto Scaling [?] to manage and scale your EC2 capacity.
- IP addresses
 - Supports load balancing to VPC and on-premises resources.
 - Facilitates routing to multiple IP addresses and network interfaces on the same instance.
 - Offers flexibility with microservice based architectures, simplifying inter-application communication.
 - Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.
- Lambda function
 - Facilitates routing to a single Lambda function.
 - Accessible to Application Load Balancers only.
- Application Load Balancer
 - Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
 - Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Bottom footer: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the "AppTierTG" target group creation page:

The title is "AppTierTG". A note says: "A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen."

Protocol : Port

Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation.

Protocol dropdown: HTTP Port: 4000 Subnets: 1-65555

IP address type

Only targets with the indicated IP address type can be registered to this target group.

IPv4

- Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.

IPv6

- Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). [Learn more \[?\]](#)

VPC

Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

VPC dropdown: webappVPC
vpc-0ca97ae97923efaf57
IPv4 VPC CIDR: 10.0.0.0/16

Protocol version

HTTP1

- Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.

Bottom footer: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the 'Health checks' configuration page in the AWS CloudFront console. At the top, there's a navigation bar with various AWS services like EC2, IAM, S3, VPC, DynamoDB, RDS, CloudWatch, CloudFormation, AWS Auto Scaling, Database Migration Service, CloudFront, and Key Management Service. Below the navigation bar, the main content area is titled 'Health checks'. It contains several configuration sections:

- Health check protocol:** Set to 'HTTP'.
- Health check path:** Set to '/health'. A note indicates a maximum length of 1024 characters.
- Advanced health check settings:** A section with a 'Restore defaults' button. It includes:
 - Health check port:** A note explaining it's the same as the target group's traffic port. Options are 'Traffic port' (selected) and 'Override'.
 - Healthy threshold:** Set to '2' (range 2-10).
 - Unhealthy threshold:** (Not visible in the screenshot)

At the bottom of the page, there are links for CloudShell, Feedback, and legal notices: © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

Do not select any target instances, directly create a target group

➤ Internal Load Balancer

The screenshot shows the AWS Management Console interface for managing load balancers.

Top Navigation: Services > EC2 > Load balancers

Left Sidebar:

- Elastic Block Store: Volumes, Snapshots, Lifecycle Manager
- Network & Security: Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces
- Load Balancing: Load Balancers (selected), Target Groups, Trust Stores (New)
- Auto Scaling: Auto Scaling Groups
- Settings

Main Content:

Load balancers (No load balancers)

0 load balancers selected

Select a load balancer above.

Comparison Diagram:

Application Load Balancer Info	Network Load Balancer Info	Gateway Load Balancer Info
Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.	Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second.	Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.
Create		

How Application Load Balancers work:

Basic configuration:

Load balancer name: app-tier-internal-lb (Maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.)

Scheme: **Internal** (An internal load balancer routes requests from clients to targets using private IP addresses. Compatible with the IPv4 and Dualstack IP address types.)

Load balancer IP address type: **IPv4** (Includes only IPv4 addresses.)

Dualstack (Includes IPv4 and IPv6 addresses.)

Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.

Availability Zones

us-east-1a (use1-az2)

Subnet

subnet-0a3foc9bd7b2bd059
IPv4 subnet CIDR: 10.0.2.0/24

Private-App-Subnet-AZ1

us-east-1b (use1-az4)

Subnet

subnet-0afb07b016ae01cb9
IPv4 subnet CIDR: 10.0.3.0/24

Private-App-Subnet-AZ2

Security groups Info

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

Security groups

Select up to 5 security groups

Internal-lb-sg
sg-02a537a87a0432afe VPC: vpc-0ca97ac97923efa57

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Internal-lb-sg
sg-02a537a87a0432afe VPC: vpc-0ca97ac97923efa57

Listeners and routing Info

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener HTTP:80

Remove

Protocol

HTTP : 80

Port

Default action [Info](#)

Forward to

AppTierTG

Target type: Instance, IPv4

HTTP

▼

C

[Create target group](#)

Listener tags - optional

Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add listener tag

You can add up to 50 more tags.

Add listener

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Hit Create Load Balancer

➤ App Tier Launch Template

Before we configure Auto Scaling, we need to create a Launch template with the AMI we created earlier

The screenshots illustrate the step-by-step process of creating an App Tier Launch Template:

- Screenshot 1: EC2 Launch Templates Overview**

This screenshot shows the EC2 Launch Templates page. It features a main heading "EC2 launch templates" and a sub-section "Streamline, simplify and standardize instance launches". Below this, there's a brief description of what launch templates do, followed by a "New launch template" button with a red border and a "Create launch template" button.
- Screenshot 2: Create launch template wizard - Step 1**

This screenshot shows the "Create launch template" wizard starting with the "Launch template name and description" step. The "Launch template name - required" field contains "AppTier-LaunchTemplate". A tooltip for "Free tier" is visible, stating: "In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB".
- Screenshot 3: Create launch template wizard - Step 2**

This screenshot shows the "Application and OS Images (Amazon Machine Image)" step. It lists an "AppTierImage" AMI (ami-04327ca5bf553a84d). A tooltip for "Free tier" is visible, stating: "In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB".

Instance type

Software Image (AMI)
AppTierImage
ami-04527ca5bf355a84d

Virtual server type (instance type)
t2.micro

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

Key pair (login)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name
Don't include in launch template

Network settings

Subnet
Don't include in launch template

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.

Select existing security group
PrivateInstance-sg

Security groups
Select security groups
PrivateInstance-sg sg-068b42b5c0ed3a965 X
VPC: vpc-0ca07ac97925efaf5

Storage (volumes)

Advanced details

IAM instance profile
RoleForEC2toAccessSSS
arn:aws:iam::471112626862:instance-profile/RoleForEC2toAccessSSS

Hostname type
Info
Don't include in launch template

DNS Hostname
 Enable resource-based IPv4 (A record) DNS requests
 Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery
Info
Don't include in launch template

Shutdown behavior
Info
Don't include in launch template

Stop - Hibernate behavior
Info
Don't include in launch template

Summary

Software Image (AMI)
AppTierImage
ami-04527ca5bf355a84d

Virtual server type (instance type)
t2.micro

Firewall (security group)

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB

Create launch template

Cancel

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Network settings

Subnet
Don't include in launch template

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.

Select existing security group
PrivateInstance-sg

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

Storage (volumes)

Advanced details

IAM instance profile
RoleForEC2toAccessSSS
arn:aws:iam::471112626862:instance-profile/RoleForEC2toAccessSSS

Hostname type
Info
Don't include in launch template

DNS Hostname
 Enable resource-based IPv4 (A record) DNS requests
 Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery
Info
Don't include in launch template

Shutdown behavior
Info
Don't include in launch template

Stop - Hibernate behavior
Info
Don't include in launch template

Summary

Software Image (AMI)
AppTierImage
ami-04527ca5bf355a84d

Virtual server type (instance type)
t2.micro

Firewall (security group)

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB

Create launch template

Cancel

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

➤ App Tier Auto Scaling Group

Amazon EC2 Auto Scaling helps maintain the availability of your applications

Auto Scaling groups are collections of Amazon EC2 instances that enable automatic scaling and fleet management features. These features help you maintain the health and availability of your applications.

Create Auto Scaling group

Get started with EC2 Auto Scaling by creating an Auto Scaling group.

Create Auto Scaling group

How it works

Pricing

Amazon EC2 Auto Scaling features have no additional fees beyond the service fees for Amazon EC2, CloudWatch (for scaling policies), and the other AWS resources that you use. Visit the pricing page of each service to learn more.

Name

Auto Scaling group name
Enter a name to identify the group.
 Must be unique to this account in the current Region and no more than 255 characters.

Launch template [Info](#)

For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.
 [Create a launch template](#)

Version
 [Create a launch template version](#)

Network [Info](#)

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.
 [Create a VPC](#)

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.
 [Create a subnet](#)

us-east-1a | subnet-0a3f0c9bd7b2bd059 (Private-App-Subnet-A21)
10.0.2.0/24

us-east-1b | subnet-0afb07b016ae01cb9 (Private-App-Subnet-A22)
10.0.3.0/24

Next

Screenshot of Step 4: Configure group size and scaling.

Load balancing Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

Attach to an existing load balancer
Choose from your existing load balancers.

Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to an existing load balancer

Select the load balancers that you want to attach to your Auto Scaling group.

Choose from your load balancer target groups
This option allows you to attach Application, Network, or Gateway Load Balancers.

Choose from Classic Load Balancers

Existing load balancer target groups

Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

Select target groups

AppTierTG HTTP
Application Load Balancer: app-tier-internal-lb

Screenshot of Step 5: Configure group size and scaling.

Step 2 Choose instance launch options

Step 3 - optional Configure advanced options

Step 4 - optional Configure group size and scaling

Step 5 - optional Add notifications

Step 6 - optional Add tags

Step 7 Review

Group size Info

Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

Desired capacity type

Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances)

Desired capacity

Specify your group size.

2

Scaling Info

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits

Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity	Max desired capacity
2	2
Equal or less than desired	Equal or greater than desired

Screenshot of Step 6: Add notifications and Step 7: Create Auto Scaling group.

Instance scale-in protection

Instance scale-in protection

Enable instance protection from scale in

Step 5: Add notifications

Notifications

No notifications

Step 6: Add tags

Tags (0)

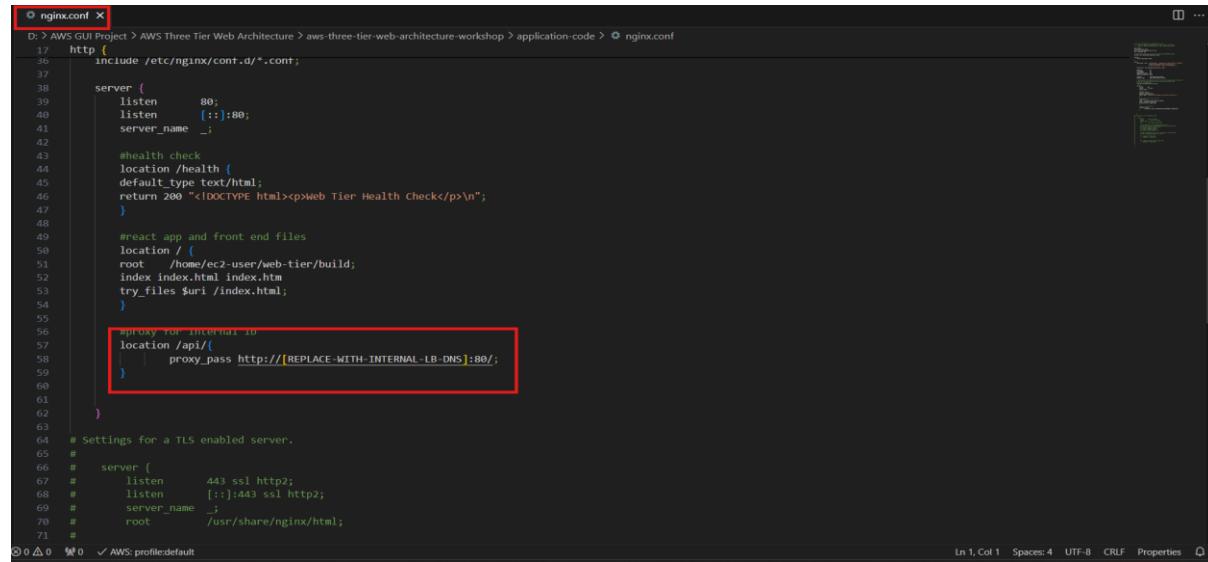
Key	Value	Tag new instances

No tags

Create Auto Scaling group

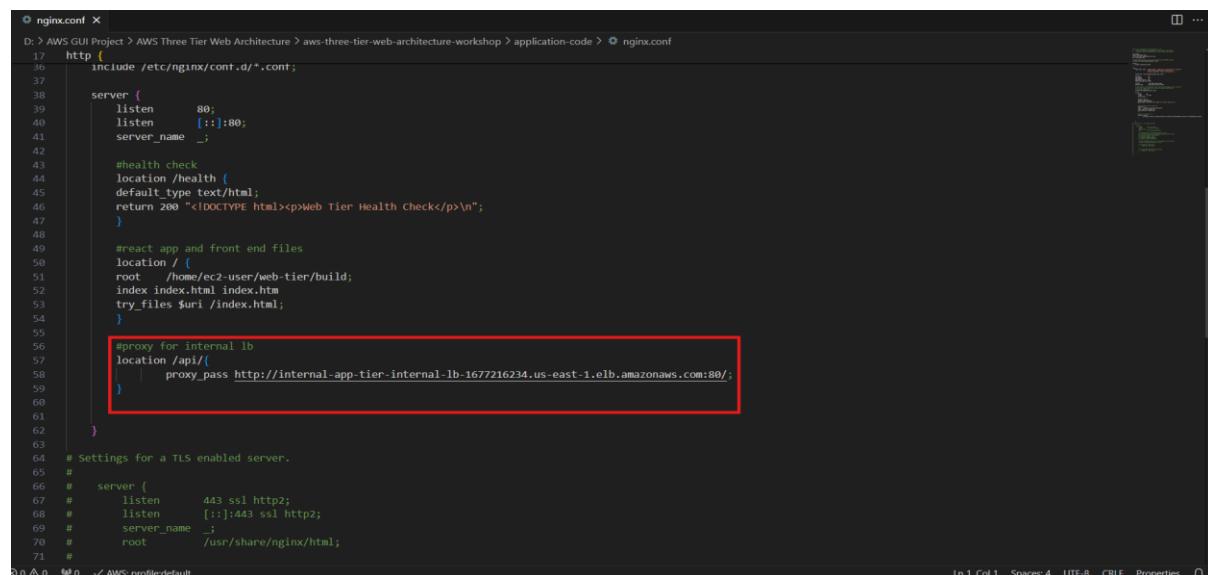
6. Web Tier Instance Deployment

➤ Update nginx Config file



```
nginx.conf
D: > AWS GUI Project > AWS Three Tier Web Architecture > aws-three-tier-web-architecture-workshop > application-code > nginx.conf

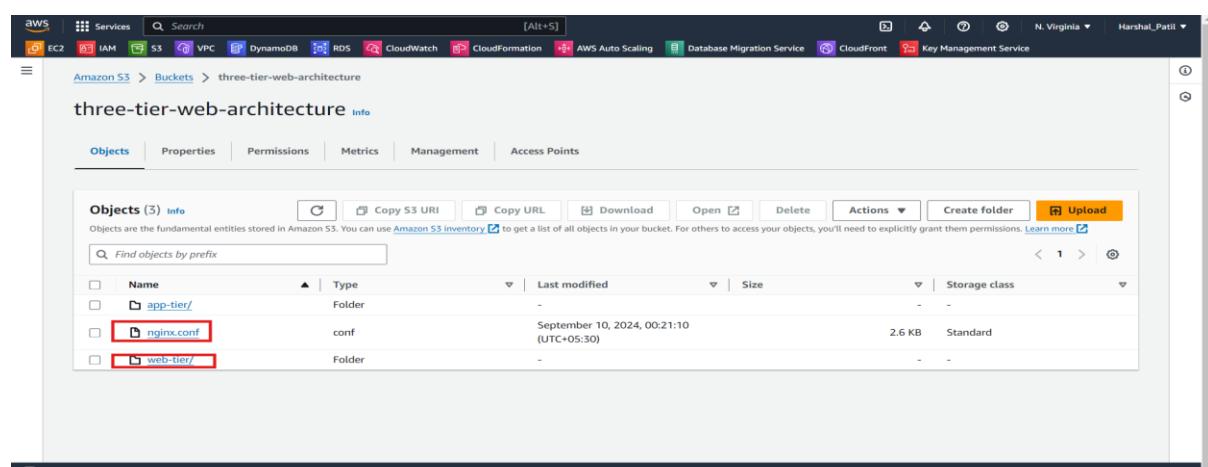
17 http {
18     include /etc/nginx/conf.d/*.conf;
19
20     server {
21         listen      80;
22         listen      [::]:80;
23         server_name _;
24
25         #health check
26         location /health {
27             default_type text/html;
28             return 200 "<!DOCTYPE html><p>Web Tier Health Check</p>\n";
29         }
30
31         #react app and front end files
32         location / {
33             root   /home/ec2-user/web-tier/build;
34             index index.html index.htm;
35             try_files $uri /index.html;
36         }
37
38         #proxy for internal lb
39         location /api/ {
40             proxy_pass http://[REPLACE-WITH-INTERNAL-LB-DNS]:80/;
41         }
42
43     }
44
45     # Settings for a TLS enabled server.
46     #
47     #   server {
48     #       listen      443 ssl http2;
49     #       listen      [::]:443 ssl http2;
50     #       server_name _;
51     #       root       /usr/share/nginx/html;
52     #   }
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71 }
```



```
nginx.conf
D: > AWS GUI Project > AWS Three Tier Web Architecture > aws-three-tier-web-architecture-workshop > application-code > nginx.conf

17 http {
18     include /etc/nginx/conf.d/*.conf;
19
20     server {
21         listen      80;
22         listen      [::]:80;
23         server_name _;
24
25         #health check
26         location /health {
27             default_type text/html;
28             return 200 "<!DOCTYPE html><p>Web Tier Health Check</p>\n";
29         }
30
31         #react app and front end files
32         location / {
33             root   /home/ec2-user/web-tier/build;
34             index index.html index.htm;
35             try_files $uri /index.html;
36         }
37
38         #proxy for internal lb
39         location /api/ {
40             proxy_pass http://internal-app-tier-internal-lb-1677216234.us-east-1.elb.amazonaws.com:80/;
41         }
42
43     }
44
45     # Settings for a TLS enabled server.
46     #
47     #   server {
48     #       listen      443 ssl http2;
49     #       listen      [::]:443 ssl http2;
50     #       server_name _;
51     #       root       /usr/share/nginx/html;
52     #   }
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71 }
```

Then, upload this file and the application-code/web-tier folder to the s3 bucket



Amazon S3 > Buckets > three-tier-web-architecture

three-tier-web-architecture [Info](#)

Objects (3) [Info](#)

Name	Type	Last modified	Size	Storage class
app-tier/	Folder		-	-
nginx.conf	conf	September 10, 2024, 00:21:10 (UTC+05:30)	2.6 KB	Standard
web-tier/	Folder		-	-

➤ Web Instance deployment

The following screenshots show the AWS EC2 console interface for launching a new instance.

Screenshot 1: Launch an instance - Step 1

Screenshot 2: Launch an instance - Step 2

Screenshot 3: Launch an instance - Step 3

Screenshot 4: Launch an instance - Step 4

Screenshot 5: Launch an instance - Step 5

VPC - required [Info](#)

vpc-0ca97ac97923efa57 (webappVPC)
10.0.0.0/16

Subnet [Info](#)

subnet-05ec9b87edcb484e2 Public-Web-Subnet-AZ1
VPC: vpc-0ca97ac97923efa57 Owner: 471112626862 Availability Zone: us-east-1a
Zone type: Availability Zone IP addresses available: 250 CIDR: 10.0.0.0/24

[Create new subnet](#)

Auto-assign public IP [Info](#)

Enable

Additional charges apply when outside of **free tier allowance**

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

WebTier-sg sg-0596682817a20e2d7 X
VPC: vpc-0ca97ac97923efa57

Compare security group rules

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

Advanced network configuration

[CloudShell](#) [Feedback](#)

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.5.2...[read more](#)
ami-0182f573e66f89c85

Virtual server type (instance type)
t2.micro

Firewall (security group)
WebTier-sg

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which

[Launch instance](#)

[Review commands](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Advanced details [Info](#)

Domain join directory [Info](#)
Select [Create new directory](#)

IAM instance profile [Info](#)

RoleForEC2toAccessS3
arn:aws:iam:471112626862:instance-profile/RoleForEC2toAccessS3

[Create new IAM profile](#)

Hostname type [Info](#)

IP name

DNS Hostname [Info](#)

Enable IP name IPv4 (A record) DNS requests
 Enable resource-based IPv4 (A record) DNS requests
 Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery [Info](#)

Select

Shutdown behavior [Info](#)

Stop [Create new shutdown behavior](#)

[CloudShell](#) [Feedback](#)

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.5.2...[read more](#)
ami-0182f573e66f89c85

Virtual server type (instance type)
t2.micro

Firewall (security group)
WebTier-sg

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which

[Launch instance](#)

[Review commands](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

➤ Connect to instance

The screenshot shows the AWS EC2 Instances page. A modal window is open for the instance **i-099c112422c8f211c (WebServer)**. At the top of this modal, there is a red box around the **Connect** button. Below the modal, the main EC2 Instances list shows three instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
WebServer	i-099c112422c8f211c	Running	t2.micro	Initializing	View alarms +	us-east-1a
App-Tier-Instance-AZ1	i-052318fe0ff06f20a	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a
App-Tier-Instance-AZ2	i-01d5dc0e0fcf5d51e	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b

```
sh-5.2$ sudo -u ec2-user
[ec2-user@ip-10-0-0-132 bin]$ ping google.com
PING google.com (142.251.163.130) 56(84) bytes of data.
64 bytes from wv-in-f138.1e100.net (142.251.163.130): icmp_seq=1 ttl=56 time=2.03 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.130): icmp_seq=2 ttl=56 time=1.66 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.130): icmp_seq=3 ttl=56 time=1.74 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.130): icmp_seq=4 ttl=56 time=1.69 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.130): icmp_seq=5 ttl=56 time=1.97 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.130): icmp_seq=6 ttl=56 time=2.11 ms
^C
--- google.com ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5008ms
rtt min/avg/max/mdev = 1.658/1.865/2.105/0.175 ms
[ec2-user@ip-10-0-0-132 bin]$
```

➤ Configure Web Instance

We now need to install all of the necessary components needed to run our front-end application

```
sh-5.2$ sudo -su ec2-user
[ec2-user@ip-10-0-0-132 bin]$ ping google.com
PING google.com (142.251.163.138) 56(84) bytes of data.
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=1 ttl=56 time=2.03 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=2 ttl=56 time=1.66 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=3 ttl=56 time=1.74 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=4 ttl=56 time=1.69 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=5 ttl=56 time=1.97 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=6 ttl=56 time=2.11 ms
```
-- google.com ping statistics --
6 packets transmitted, 6 received, 0% packet loss, time 5008ms
rtt min/avg/max/mdev = 1.658/1.865/2.105/0.175 ms
[ec2-user@ip-10-0-0-132 bin]$ curl -o https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash
 % Total % Received % Xferd Average Speed Time Time Current
 Dload Upload Total Spent Left Speed
100 14926 100 14926 0 0 313k 0:--:--:--:--:--:--: 316k
=> Downloading nvm as script to '/home/ec2-user/.nvm'
=> Appending nvm source string to /home/ec2-user/.bashrc
=> Appending bash_completion source string to /home/ec2-user/.bashrc
=> Close and reopen your terminal to start using nvm or run the following to use it now:

export NVM_DIR="$HOME/.nvm"
[-s "$NVM_DIR/nvm.sh"] && \."$NVM_DIR/nvm.sh" # This loads nvm
[-s "$NVM_DIR/bash_completion"] && \."$NVM_DIR/bash_completion" # This loads nvm bash_completion
[ec2-user@ip-10-0-0-132 bin]$ source ~/.bashrc
[ec2-user@ip-10-0-0-132 bin]$
```

```
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=1 ttl=56 time=2.03 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=2 ttl=56 time=1.66 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=3 ttl=56 time=1.74 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=4 ttl=56 time=1.69 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=5 ttl=56 time=1.97 ms
64 bytes from wv-in-f138.1e100.net (142.251.163.138): icmp_seq=6 ttl=56 time=2.11 ms
```
-- google.com ping statistics --
6 packets transmitted, 6 received, 0% packet loss, time 5008ms
rtt min/avg/max/mdev = 1.658/1.865/2.105/0.175 ms
[ec2-user@ip-10-0-0-132 bin]$ curl -o https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash
  % Total    % Received % Xferd  Average Speed   Time   Time  Current
                                 Dload  Upload  Total  Spent   Left  Speed
100 14926  100 14926    0     0  313k  0:--:--:--:--:--:--: 316k
=> Downloading nvm as script to '/home/ec2-user/.nvm'
=> Appending nvm source string to /home/ec2-user/.bashrc
=> Appending bash_completion source string to /home/ec2-user/.bashrc
=> Close and reopen your terminal to start using nvm or run the following to use it now:

export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \."$NVM_DIR/nvm.sh" # This loads nvm
[ -s "$NVM_DIR/bash_completion" ] && \."$NVM_DIR/bash_completion" # This loads nvm bash_completion
[ec2-user@ip-10-0-0-132 bin]$ source ~/.bashrc
[ec2-user@ip-10-0-0-132 bin]$ nvm install 16
Downloading and installing node v16.20.2...
Downloaded https://nodejs.org/dist/v16.20.2/node-v16.20.2-linux-x64.tar.xz...
#####
Computing checksum with sha256sum
Checksums matched!
Now using node v16.20.2 (npm v8.19.4)
Creating default alias: default --> 16 (> v16.20.2)
[ec2-user@ip-10-0-0-132 bin]$ nvm use 16
Now using node v16.20.2 (npm v8.19.4)
[ec2-user@ip-10-0-0-132 bin]$
```

Now we need to download our web tier code from our s3 bucket:

```
[ec2-user@ip-10-0-0-132 bin]$ cd
[ec2-user@ip-10-0-0-132 ~]$ aws s3 cp s3://three-tier-web-architecture/web-tier/ web-tier --recursive
download: s3://three-tier-web-architecture/web-tier/README.md to web-tier/README.md
download: s3://three-tier-web-architecture/web-tier/src/App.test.js to web-tier/src/App.test.js
download: s3://three-tier-web-architecture/web-tier/src/App.js to web-tier/src/App.js
download: s3://three-tier-web-architecture/web-tier/src/components/DatabaseDemo/DatabaseDemo.js to web-tier/src/components/DatabaseDemo/DatabaseDemo.js
download: s3://three-tier-web-architecture/web-tier/package.json to web-tier/package.json
download: s3://three-tier-web-architecture/web-tier/src/components/Menu/Menu.js to web-tier/src/components/Menu/Menu.js
download: s3://three-tier-web-architecture/web-tier/src/components/Menu/index.js to web-tier/src/components/Menu/index.js
download: s3://three-tier-web-architecture/web-tier/public/index.html to web-tier/public/index.html
download: s3://three-tier-web-architecture/web-tier/src/components/Home/Home.js to web-tier/src/components/Home/Home.js
download: s3://three-tier-web-architecture/web-tier/src/components/DS_Store to web-tier/src/components/DS_Store
download: s3://three-tier-web-architecture/web-tier/src/components/Burger/Burger.styled.js to web-tier/src/components/Burger/Burger.styled.js
download: s3://three-tier-web-architecture/web-tier/src/components/Burger/Index.js to web-tier/src/components/Burger/Index.js
download: s3://three-tier-web-architecture/web-tier/src/components/Menu/Menu.styled.js to web-tier/src/components/Menu/Menu.styled.js
download: s3://three-tier-web-architecture/web-tier/src/components/index.js to web-tier/src/components/index.js
download: s3://three-tier-web-architecture/web-tier/src/theme.js to web-tier/src/theme.js
download: s3://three-tier-web-architecture/web-tier/src/index.css to web-tier/src/index.css
download: s3://three-tier-web-architecture/web-tier/src/global.js to web-tier/src/global.js
download: s3://three-tier-web-architecture/web-tier/src/setupTests.js to web-tier/src/setupTests.js
download: s3://three-tier-web-architecture/web-tier/src/components/Burger/Burger.js to web-tier/src/components/Burger/Burger.js
download: s3://three-tier-web-architecture/web-tier/src/hooks.js to web-tier/src/hooks.js
download: s3://three-tier-web-architecture/web-tier/public/robots.txt to web-tier/public/robots.txt
download: s3://three-tier-web-architecture/web-tier/src/components/DatabaseDemo/DatabaseDemo.css to web-tier/src/components/DatabaseDemo/DatabaseDemo.css
download: s3://three-tier-web-architecture/web-tier/src/App.css to web-tier/src/App.css
download: s3://three-tier-web-architecture/web-tier/src/DS_Store to web-tier/src/DS_Store
download: s3://three-tier-web-architecture/web-tier/src/reportWebVitals.js to web-tier/src/reportWebVitals.js
download: s3://three-tier-web-architecture/web-tier/src/assets/3TierArch.png to web-tier/src/assets/3TierArch.png
[ec2-user@ip-10-0-0-132 ~]$
```

Navigate to the web-layer folder and create the build folder for the react app so we can serve our code:

```
download: s3://three-tier-web-architecture/web-tier/src/index.js to web-tier/src/index.js
download: s3://three-tier-web-architecture/web-tier/src/theme.js to web-tier/src/theme.js
download: s3://three-tier-web-architecture/web-tier/src/index.css to web-tier/src/index.css
download: s3://three-tier-web-architecture/web-tier/src/global.js to web-tier/src/global.js
download: s3://three-tier-web-architecture/web-tier/src/setupTests.js to web-tier/src/setupTests.js
download: s3://three-tier-web-architecture/web-tier/src/components/Burger/Burger.js to web-tier/src/components/Burger/Burger.js
download: s3://three-tier-web-architecture/web-tier/src/hooks.js to web-tier/src/hooks.js
download: s3://three-tier-web-architecture/web-tier/public/robots.txt to web-tier/public/robots.txt
download: s3://three-tier-web-architecture/web-tier/src/components/databaseDemo.css to web-tier/src/components/DatabaseDemo/DatabaseDemo.css
download: s3://three-tier-web-architecture/web-tier/src/App.css to web-tier/src/App.css
download: s3://three-tier-web-architecture/web-tier/.DS_Store to web-tier/.DS_Store
download: s3://three-tier-web-architecture/web-tier/src/reportWebVitals.js to web-tier/src/reportWebVitals.js
download: s3://three-tier-web-architecture/web-tier/src/assets/3TierArch.png to web-tier/src/assets/3TierArch.png
[ec2-user@ip-10-0-0-132 ~]$ cd ~/web-tier
[ec2-user@ip-10-0-0-132 web-tier]$ npm install
npm warn EBADENGINE Unsupported engine {
npm warn   package: 'testing-library-dom@10.4.0',
npm warn   required: { node: '>v18' },
npm warn   current: { node: 'v16.20.2', npm: '8.19.4' }
npm warn   )
npm warn deprecated sourcemap-codec@1.4.8: Please use @jridgewell/sourcemap-codec instead
npm warn deprecated rollup-plugin-terser@7.0.2: This package has been deprecated and is no longer maintained. Please use @rollup/plugin-terser
npm warn deprecated workbox-cacheable-response@6.6.0: workbox-background-sync@6.6.0
npm warn deprecated workbox-google-analytics@6.6.0: It is not compatible with newer versions of GA starting with v4, as long as you are using GAv3 it should be ok, but the
package is not longer being maintained
npm warn deprecated stable@0.1.8: Modern JS already guarantees Array#sort() is a stable sort, so this library is deprecated. See the compatibility table on MDN: https://de
veloper.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/sort#browser_compatibility
npm warn deprecated qe15.1.1: You or someone you depend on is using Q, the JavaScript Promise library that gave JavaScript developers strong feelings about promises. They c
an almost certainly migrate to the native JavaScript promise now. Thank you literally everyone for joining me in this bet against the odds. Be excellent to each other.
npm warn deprecated
npm warn deprecated (For a CapTP with native promises, see @endo/eventual-send and @endo/capt)
npm warn deprecated w3c-hr-time@1.0.2: Use your platform's native performance.now() and performance.timeOrigin.
npm warn deprecated domexception@2.0.1: Use your platform's native DOMException instead
npm warn deprecated abab@2.0.6: Use your platform's native atob() and btoa() methods instead
npm warn deprecated inflight@1.0.6: This module is not supported, and leaks memory. Do not use it. Check out lru-cache if you want a good and tested way to coalesce async
```

```
8 vulnerabilities (2 moderate, 6 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
npm notice New major version of npm available! 8.19.4 -> 10.8.3
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.3
npm notice Run npm install -g npm@10.8.3 to update!
[ec2-user@ip-10-0-0-132 web-tier]$ npm run build
> aws-3tier-web-layer@0.1.0 build
> react-scripts build

Creating an optimized production build...
One of your dependencies, babel-preset-react-app, is importing the
"@babel/plugin-proposal-private-property-in-object" package without
declaring it in its dependencies. This is currently working because
"@babel/plugin-proposal-private-property-in-object" is already in your
node_modules folder for unrelated reasons, but it may break at any time.
babel-preset-react-app is part of the create-react-app project, which
is not maintained anymore. It is thus unlikely that this bug will
ever be fixed. Add "@babel/plugin-proposal-private-property-in-object" to
your devDependencies to work around this error. This will make this message
go away.

Compiled successfully.

File sizes after gzip:

  74.96 kB  build/static/js/main.7d879652.js
```

Install nginx web server

```
1.79 kB  build/static/js/453.a4ec9c9e.chunk.js
493 B    build/static/css/main.b20b6ac4.css

The project was built assuming it is hosted at ./.
You can control this with the homepage field in your package.json.

The build folder is ready to be deployed.

Find out more about deployment here:
  https://cra.link/deployment

[ec2-user@ip-10-0-0-132 web-tier]$ sudo amazon-linux-extras install nginx1 -y
sudo: amazon-linux-extras: command not found
[ec2-user@ip-10-0-0-132 web-tier]$ Sudo yum install nginx
Last metadata expiration check: 0:14:36 ago on Mon Sep 9 10:57:58 2024.
Dependencies resolved.

=====
Package          Architecture Version       Repository      Size
=====
Installing:
nginx           x86_64     1:1.24.0-1.amzn2023.0.3  amazonlinux   33 k
Installing dependencies:
generic-logs-nginx          noarch    18.0.0-12.amzn2023.0.3  amazonlinux   19 k
gperf-tools-libs             x86_64    2.9.1-1.amzn2023.0.3  amazonlinux   308 k
libnumwind                x86_64    1.4.0-5.amzn2023.0.2   amazonlinux   66 k
nginx-core                 x86_64    1:1.24.0-1.amzn2023.0.3  amazonlinux   586 k
nginx-filesystem            noarch    1:1.24.0-1.amzn2023.0.3  amazonlinux   9.6 k
nginx-mimetypes             noarch    2.1.49-3.amzn2023.0.3  amazonlinux   21 k

Transaction Summary
=====
Install 7 Packages
```

Delete already present nginx.conf file and use the one which we have uploaded in S3

```
Installing      : nginx-1:1.24.0-1.amzn2023.0.3.x86_64          7/7
Running scriptlet: nginx-1:1.24.0-1.amzn2023.0.3.x86_64          7/7
Verifying       : generic-logos-https-18.0.0-12.amzn2023.0.3.noarch 1/7
Verifying       : gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64        2/7
Verifying       : libumwind-1.4.0-5.amzn2023.0.2.x86_64           3/7
Verifying       : nginx-1:1.24.0-1.amzn2023.0.3.x86_64           4/7
Verifying       : nginx-core-1:1.24.0-1.amzn2023.0.3.x86_64           5/7
Verifying       : nginx-fs-1:1.24.0-1.amzn2023.0.3.noarch          6/7
Verifying       : nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch          7/7

Installed:
generic-logos-https-18.0.0-12.amzn2023.0.3.noarch          7/7
nginx-1:1.24.0-1.amzn2023.0.3.x86_64                   7/7
nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch          7/7

Complete!
[ec2-user@ip-10-0-0-132 web-tier]$ cd /etc/nginx
[ec2-user@ip-10-0-0-132 nginx]$ ls
conf.d    fastcgi.conf    fastcgi_params    koi-utf    mime.types    nginx.conf    scgi_params    uwsgi_params    win-utf
default.d fastcgi.conf.default fastcgi_params.default koi-win    mime.types.default nginx.conf.default scgi_params.default uwsgi_params.default
[ec2-user@ip-10-0-0-132 nginx]$ sudo rm nginx.conf
[ec2-user@ip-10-0-0-132 nginx]$ sudo aws s3 cp s3://three-tier-web-architecture/nginx.conf

usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

aws help
aws <command> help
aws <command> <subcommand> help

aws: error: the following arguments are required: paths

[ec2-user@ip-10-0-0-132 nginx]$ sudo aws s3 cp s3://three-tier-web-architecture/nginx.conf .
download: s3://three-tier-web-architecture/nginx.conf to ./nginx.conf
[ec2-user@ip-10-0-0-132 nginx]$
```

Restart nginx, give permission to access our files and make sure the service stars on boot

```
Verifying      : nginx-core-1:1.24.0-1.amzn2023.0.3.x86_64          5/7
Verifying      : nginx-fs-1:1.24.0-1.amzn2023.0.3.noarch          6/7
Verifying      : nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch          7/7

Installed:
generic-logos-https-18.0.0-12.amzn2023.0.3.noarch          5/7
nginx-1:1.24.0-1.amzn2023.0.3.x86_64                   6/7
nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch          7/7

Complete!
[ec2-user@ip-10-0-0-132 web-tier]$ cd /etc/nginx
[ec2-user@ip-10-0-0-132 nginx]$ ls
conf.d    fastcgi.conf    fastcgi_params    koi-utf    mime.types    nginx.conf    scgi_params    uwsgi_params    win-utf
default.d fastcgi.conf.default fastcgi_params.default koi-win    mime.types.default nginx.conf.default scgi_params.default uwsgi_params.default
[ec2-user@ip-10-0-0-132 nginx]$ sudo rm nginx.conf
[ec2-user@ip-10-0-0-132 nginx]$ sudo aws s3 cp s3://three-tier-web-architecture/nginx.conf

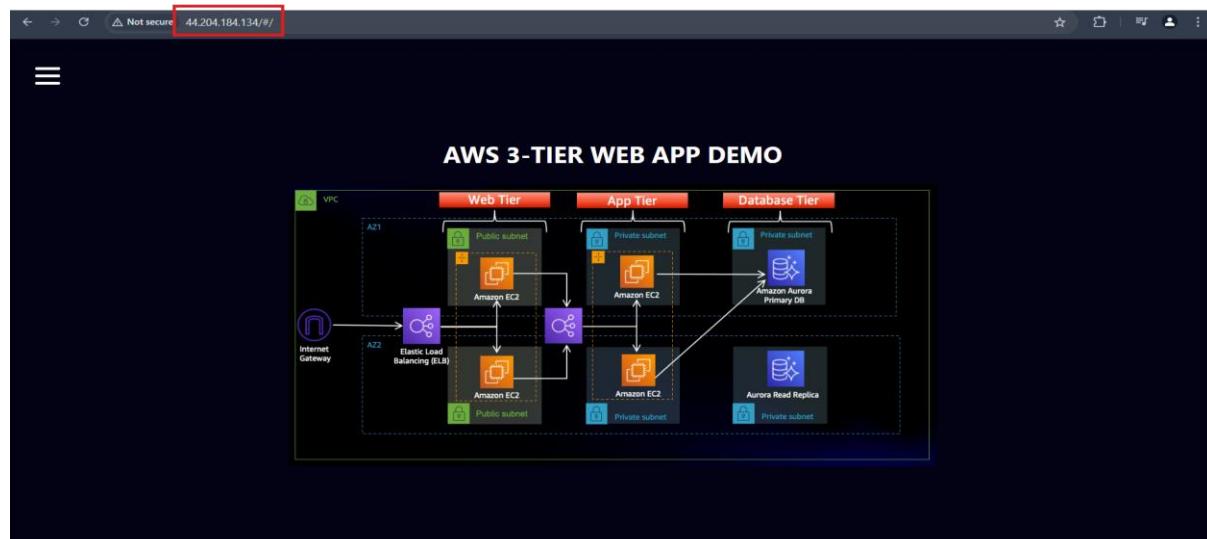
usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

aws help
aws <command> help
aws <command> <subcommand> help

aws: error: the following arguments are required: paths

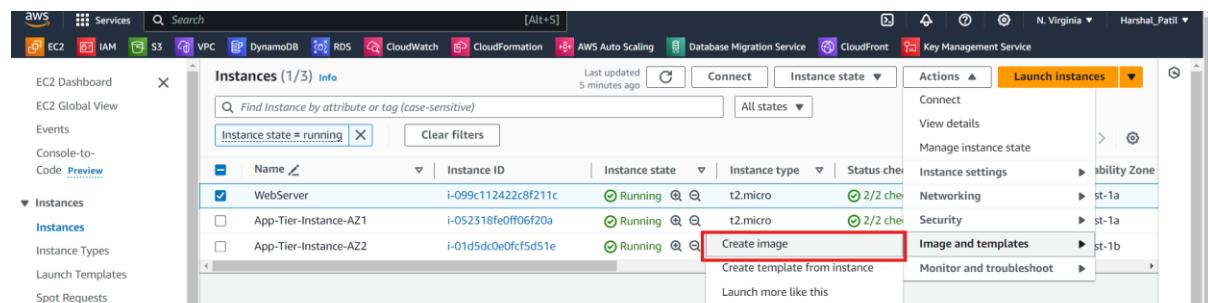
[ec2-user@ip-10-0-0-132 nginx]$ sudo aws s3 cp s3://three-tier-web-architecture/nginx.conf .
download: s3://three-tier-web-architecture/nginx.conf to ./nginx.conf
[ec2-user@ip-10-0-0-132 nginx]$ sudo service nginx restart
Redirecting to /bin/systemctl restart nginx.service
[ec2-user@ip-10-0-0-132 nginx]$ chmod -R 755 /home/ec2-user
[ec2-user@ip-10-0-0-132 nginx]$ sudo chkconfig nginx on
Note: Forwarding request to 'systemctl enable nginx.service'.
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /usr/lib/systemd/system/nginx.service.
[ec2-user@ip-10-0-0-132 nginx]$
```

Now we can access the website by using public IP of instance



7. External Load Balancer and Auto Scaling Group

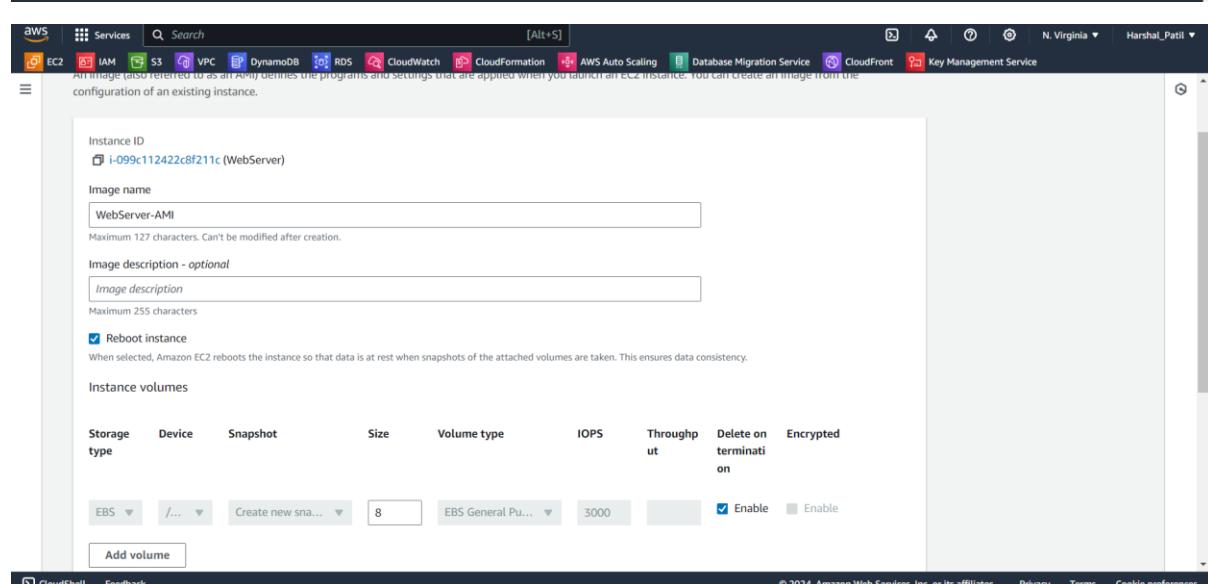
➤ Web Tier AMI



The screenshot shows the AWS EC2 Instances page. There are three instances listed:

- WebServer**: Instance ID i-099c112422c8f211c, Running, t2.micro, 2/2 checked.
- App-Tier-Instance-AZ1**: Instance ID i-052318fe0ff0f6f20a, Running, t2.micro, 2/2 checked.
- App-Tier-Instance-AZ2**: Instance ID i-01d5dc0e0fcf5d51e, Running, t2.micro, 2/2 checked.

In the Actions menu for the WebServer instance, the "Create image" option is highlighted with a red box. Other options in the menu include Connect, View details, Manage instance state, Instance settings, Networking, Security, Image and templates, and Monitor and troubleshoot.



The screenshot shows the "Create image" dialog for the instance i-099c112422c8f211c (WebServer). The dialog fields are as follows:

- Instance ID:** i-099c112422c8f211c (WebServer)
- Image name:** WebServer-AMI
- Image description - optional:** (empty)
- Reboot instance:** (Description: When selected, Amazon EC2 reboots the instance so that data is at rest when snapshots of the attached volumes are taken. This ensures data consistency.)
- Instance volumes:** A table showing volume details:

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev/sda1	Create new snapshot	8	EBS General Purpose (SSD)	3000	1000	<input checked="" type="checkbox"/> Enable	<input type="checkbox"/> Enable

➤ Target Group

The screenshot shows three sequential steps in the AWS CloudFront console for creating a new target group:

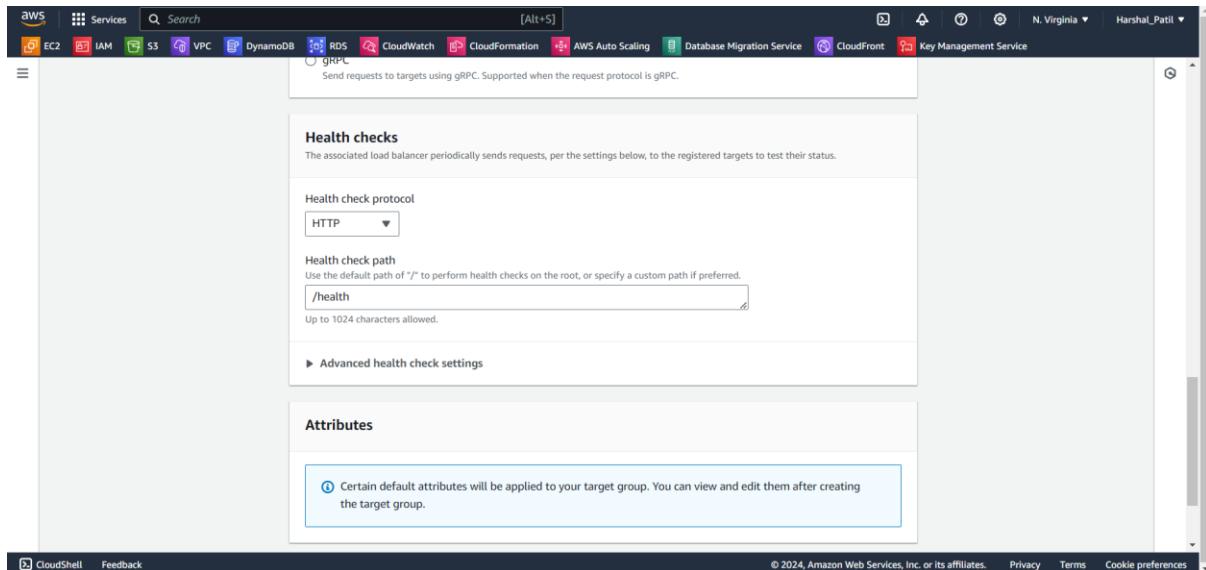
- Step 1: Target groups**

The "Create target group" button is highlighted in orange.

Name	ARN	Port	Protocol	Target type	Load balancer
AppTierTG	arn:aws:elasticloadbalanc...	4000	HTTP	Instance	app-tier-internal-lb
- Step 2: Register targets**

The "Basic configuration" section is shown. The "Instances" target type is selected, which supports load balancing to instances within a specific VPC and manages EC2 capacity via Auto Scaling.
- Step 3: Configure target group**

The "Target group name" is set to "WebTier-TG". The "Protocol : Port" dropdown shows "HTTP" with port "80". The "IP address type" is set to "IPv4". The "VPC" dropdown lists "webappVPC" with ARN "vpc-0ca97ac97925efaf57" and CIDR "10.0.0.0/16".



Create target group without selecting any instances

➤ Internet facing Load Balancer

Screenshot of the AWS Management Console showing the creation of an Internet-facing Load Balancer.

Top Navigation Bar:

- AWS Services
- Search
- [Alt+S]
- N. Virginia
- Harshal_Patil

Left Sidebar:

- AMIs
- AMI Catalog
- Elastic Block Store
 - Volumes
 - Snapshots
 - Lifecycle Manager
- Network & Security
 - Security Groups
 - Elastic IPs
 - Placement Groups
 - Key Pairs
 - Network Interfaces
- Load Balancing
 - Load Balancers** (selected)
 - Target Groups
 - Trust Stores [New](#)
- Auto Scaling
 - Auto Scaling Groups

Actions Bar:

- Actions
- Create load balancer** (highlighted with a red box)

Table View:

Name	DNS name	State	VPC ID	Availability Zones	Type
app-tier-internal-lb	internal-app-tier-internal-l...	Active	vpc-0ca97ac97923efa57	2 Availability Zones	application

Message: 0 load balancers selected

Select a load balancer above.

Basic Configuration Step:

Load balancer name: WebTier-lb

Scheme: Internet-facing (An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet.)

Internal: An internal load balancer routes requests from clients to targets using private IP addresses. Compatible with the IPv4 and Dualstack IP address types.

Load balancer IP address type: IPv4 (Includes only IPv4 addresses.)

Dualstack (Includes IPv4 and IPv6 addresses.)

Dualstack without public IPv4 (Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with internet-facing load balancers only.)

Network mapping: Info

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC Info: The load balancer will exist and scale within the selected VPC. The selected VPC is also where the load balancer targets must be hosted unless routing to Lambda or on-premises targets, or if using VPC peering. To confirm the VPC for your targets, view target groups. For a new VPC, create a VPC.

Selected VPC: webappVPC (vpc-0ca97ac97923efa57, IPv4 VPC CIDR: 10.0.0.0/16)

Mappings: Info

Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

Availability Zones:

- us-east-1a (use1-az2) Subnet: subnet-05ec9b87edcb484e2 (IPv4 subnet CIDR: 10.0.0.0/24)
- us-east-1b (use1-az4) Subnet: subnet-07f9bac168858d6f (IPv4 subnet CIDR: 10.0.1.0/24)

The screenshot shows the AWS CloudFront console with the following interface elements:

- Top Navigation Bar:** Includes the AWS logo, Services menu, a search bar, and account information for N. Virginia and Harshal_Patil.
- Left Sidebar:** Shows a tree view of resources under "CloudFront".
- Main Content Area:**
 - Security groups:** A dropdown menu titled "Select up to 5 security groups" contains one item: "Internet-Facing-lb-sg" (sg-04e410df73fee4acd, VPC: vpc-0ca97ac97923efa57).
 - Listeners and routing:** A section with a sub-section for "Listener HTTP:80".
 - Protocol:** HTTP (dropdown), Port: 80.
 - Default action:** Forward to "WebTier-TG" (Info button). Target type: Instance, IPv4.
 - Buttons:** "Remove" (right), "Create target group" (bottom right).
 - Listener tags - optional:** A note about adding tags to your listener.
 - Bottom Buttons:** "add Listener tag" (button), "CloudShell" (link), "Feedback" (link), and copyright information: © 2024, Amazon Web Services, Inc. or its affiliates.

Click on create load balancer

➤ Launch Template

The screenshot illustrates the process of creating a launch template in the AWS Management Console.

Top Navigation: Services menu showing EC2, IAM, S3, VPC, DynamoDB, RDS, CloudWatch, CloudFormation, AWS Auto Scaling, Database Migration Service, CloudFront, Key Management Service. Region: N. Virginia. User: Harshal_Patil.

Left Sidebar:

- EC2 Dashboard
- EC2 Global View
- Events
- Console-to-Code **Preview**
- Instances
 - Instances
 - Instance Types
 - Launch Templates**
 - Spot Requests
 - Savings Plans
 - Reserved Instances
 - Dedicated Hosts
 - Capacity
 - Reservations **New**
- Images
 - AMIs
 - AMI Catalog
- Elastic Block Store
 - Volumes

CloudShell Feedback

Launch Templates (1) Info

Launch Template ID	Launch Template Name	Default Version	Latest Version	Create Time	Created ...
lt-03020ef7e62c64f3b	AppTier-LaunchTemplate	1	1	2024-09-09T18:08:44.000Z	arn:aws:...

Select a launch template

Bottom Navigation: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Second Window: Create a launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

Launch template name and description

Launch template name - required
WebTier-ASG
Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.

Template version description
A prod webserver for MyApp
Max 255 chars

Auto Scaling guidance **Info**
Select this if you intend to use this template with EC2 Auto Scaling
 Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

► Template tags
► Source template

Launch template contents
Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

Third Window: My AMIs

Search our full catalog including 1000s of application and OS images

Recents | **My AMIs** | Quick Start

Don't include in launch template | Owned by me | Shared with me

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

WebServer-AMI
ami-0b075980b40e52a73
2024-09-09T19:27:58.000Z Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Architecture x86_64 AMI ID ami-0b075980b40e52a73

Instance type **Info** | Get advice Advanced

Bottom Navigation: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Instance type

Software Image (AMI)
WebServer-AMI
ami-0b075980b40e52a73

Virtual server type (instance type)
t2.micro

Firewall (security group)

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

Key pair (login)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name
Don't include in launch template

Network settings

Subnet

Virtual server type (instance type)
t2.micro

Firewall (security group)
WebTier-sg

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

Create launch template

Network settings

Subnet

Virtual server type (instance type)
t2.micro

Firewall (security group)
WebTier-sg

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

Create launch template

No resource tags are currently included in this template. Add a resource tag to include it in the launch template.

Advanced details

IAM instance profile

RoleForEC2toAccessS3
arn:aws:iam::471112626862:instance-profile/RoleForEC2toAccessS3

Hostname type

DNS Hostname

Enable resource-based IPv4 (A record) DNS requests

Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery

Shutdown behavior

Create launch template

➤ Web Tier Auto Scaling Group

The following screenshots show the step-by-step process of creating a Web Tier Auto Scaling Group (ASG) in the AWS Management Console.

Screenshot 1: EC2 Auto Scaling Groups Overview

Screenshot 2: Step 4 - optional: Configure group size and scaling

Screenshot 3: Step 5 - optional: Add notifications

Screenshot 4: Step 6 - optional: Add tags

Screenshot 5: Step 7: Network info

Screenshot 6: Step 7: Summary and Next button

Load balancing

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

Attach to an existing load balancer
Choose from your existing load balancers.

Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to an existing load balancer

Select the load balancers that you want to attach to your Auto Scaling group.

Choose from your load balancer target groups
This option allows you to attach Application, Network, or Gateway Load Balancers.

Choose from Classic Load Balancers

Existing load balancer target groups

Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

Select target groups

WebTier-TG | HTTP
Application Load Balancer: WebTier-lb

Step 2: Choose instance launch options

Step 3 - optional
[Configure advanced options](#)

Step 4 - optional
[Configure group size and scaling](#)

Step 5 - optional
[Add notifications](#)

Step 6 - optional
[Add tags](#)

Step 7
[Review](#)

Group size

Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

Desired capacity type
Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances)
▼

Desired capacity
Specify your group size.
2

Scaling

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity	Max desired capacity
2	2
Equal or less than desired	Equal or greater than desired

Step 5: Add notifications

Notifications

No notifications

Step 6: Add tags

Tags (0)

Key	Value	Tag new instances
No tags		

Cancel Previous **Create Auto Scaling group**

Public and Private instances created by Auto Scaling Groups

The screenshot shows the AWS EC2 Instances page with the following details:

- Instances (5) Info**
- Last updated: less than a minute ago
- Actions: Connect, Instance state, Actions, Launch instances
- Table headers: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone
- Instances listed:
 - Web-Tier-Instance-AZ2: i-006124e4fa5f82ca, Running, t2.micro, 2/2 checks passed, us-east-1b
 - Web-Tier-Instance-AZ1: i-0e6640f2084b7a79c, Running, t2.micro, 2/2 checks passed, us-east-1a
 - App-Tier-Instance-AZ1: i-052318fe0ff06f20a, Running, t2.micro, 2/2 checks passed, us-east-1a
 - App-Tier-Instance-AZ2: i-01d5dc0e0fcf5d51e, Running, t2.micro, 2/2 checks passed, us-east-1b

A modal window titled "Select an instance" is open at the bottom.

To access the application copy internet facing load balancer DNS

The screenshot shows the AWS Load Balancers page for the "WebTier-lb" load balancer:

- EC2 > Load balancers > WebTier-lb**
- WebTier-lb**
- Details** section:
 - Load balancer type: Application
 - Status: Active
 - VPC: vpc-0ca97ac97923efa57
 - Load balancer IP address type: IPv4
 - Scheme: Internet-facing
 - Hosted zone: Z355XD0TRQ7X7K
 - Availability Zones:
 - subnet-079c9bc168858def us-east-1b (use1-az4)
 - subnet-05ec9b87edcb484e2 us-east-1a (use1-az2)
 - Date created: September 10, 2024, 01:03 (UTC+05:30)
- Listeners and rules (1) Info**
- Listeners and rules** tab: Manage rules, Manage listener, Add listener

And paste in search bar

The screenshot shows a web browser displaying the "AWS 3-TIER WEB APP DEMO". The URL in the address bar is "webtier-lb-493095210.us-east-1.elb.amazonaws.com/". The page content includes:

AWS 3-TIER WEB APP DEMO

The diagram illustrates the architecture:

- Web Tier:** Contains two Amazon EC2 instances in a Public subnet.
- App Tier:** Contains three Amazon EC2 instances in a Private subnet.
- Database Tier:** Contains an Amazon RDS Primary DB and an Aurora Read Replica in a Private subnet.
- Networking:** An Internet Gateway connects to the Public subnets via AZ1 and AZ2. An Elastic Load Balancing (ELB) layer sits between the Web Tier and App Tier subnets.
- Subnets:** Public subnets for Web and App tiers, and Private subnets for App and Database tiers.

A screenshot of a web browser window titled "AURORA DATABASE DEMO PAGE". The page has a dark theme with white text. On the left side, there are two buttons: "HOME" and "DB DEMO", where "DB DEMO" is highlighted with a red border. The main content area displays a table with three columns: ID, AMOUNT, and DESC. The table contains one row with ID 1, AMOUNT 400, and DESC groceries. There is a red "ADD" button in the first column of the table, and a "DEL" button in the top right corner of the table.

ID	AMOUNT	DESC
1	400	groceries

A screenshot of the same web browser window after an item has been added. The "DB DEMO" button is still highlighted with a red border. The table now contains four rows: Row 1 (ID 1, AMOUNT 400, DESC groceries), Row 2 (ID 2, AMOUNT 20, DESC Buckets), and Row 3 (ID 3, AMOUNT 10, DESC Pens). The "ADD" button in the first column of the table is also highlighted with a red border. The "DEL" button remains in the top right corner.

ID	AMOUNT	DESC
1	400	groceries
2	20	Buckets
3	10	Pens

Our app is working successfully!!!

8. To Clean Infrastructure refer to following steps

- 1) First delete Auto Scaling Groups
- 2) Then delete listeners for both load balancers and then delete both load balancers
- 3) Delete target groups
- 4) Delete launch templates
- 5) Delete both AMI's
- 6) Terminate remaining EC2 instances if you terminated already then skip this step
- 7) To delete database make sure your database delete protection is off, then go on and delete writer instance followed by reader instance and for the last one pop-up will ask to create snapshot, uncheck that and delete.
- 8) Delete NAT Gateways
- 9) Release Elastic IP's
- 10) Delete all subnet association in route tables and delete created route tables at once
- 11) Detach VPC from Internet Gateway and delete internet gateway
- 12) Lastly, Delete VPC