

Design and configure a high available 3-tier Architecture on AWS

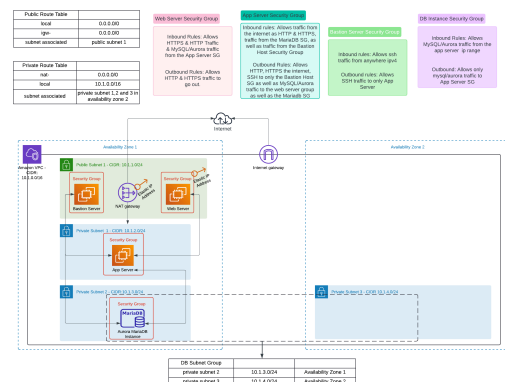
Type	AWS ARCHITECTURE
Tasks Achieved	<ul style="list-style-type: none"> • Tier 1 - User/Presentation Tier • Tier 2 - Application Tier • Tier 3 - Data Tier

Tasks Achieved

▼ VPC - CIDR Block be 10.1.0.0/16

- 4 subnets (1 public, 3 private)
- Enable in subnet settings public IP addresses
- Make it highly available (use 2 availability zones, the final private subnet can be the only one in a different subnet)
- Allocate an Elastic IP to EC2 servers
- Create a NAT gateway to allow internet access to the private subnets
- Create an internet gateway and attach it to your VPC to allow internet access for the VPC
- Make route tables for your public and private subnets and attach an internet gateway and

▼ Built Architecture



▼ POCs

Using username "ec2-user".
 Authenticating with public key "bastion host key pair"

```

, #_
~\ ####_ Amazon Lin
ux 2023
~~ \#####\
~~ \###|
~~ \#/ __ https://aws.a
  
```

NAT gateway to them respectively

- Make security groups for Bastion Host, web server, app server, and database to allow only necessary access to users
- Make sure to go back to security groups after making them and adding security groups to link them together, for example in the app server security group adding a rule for the database security group after creating the database security group.
- If you want your DB instance in the VPC to be publicly accessible, you must enable the VPC attributes DNS hostnames and DNS resolution.

▼ EC2 Instances

Bastion Host

- Amazon Linux 2 ami
- T2 Micro

Web Server

- Amazon Linux 2 ami
- T2 Micro
- User Data - to install and enable HTTPD

```
#!/bin/bash
sudo yum update -y
sudo amazon-linux-extras in
```

mazon.com/linux/amazon-linux-2023

```
~~      V~' '→
~~~~    /
~~~.~.  _/
  _/ _/
  _/m/'
```

```
[ec2-user@ip-10-1-1-155 ~]$
ls
[ec2-user@ip-10-1-1-155 ~]$
touch appserver.pem
[ec2-user@ip-10-1-1-155 ~]$
chmod 400 appserver.pem
[ec2-user@ip-10-1-1-155 ~]$
ls -la appserver.pem
-r-----. 1 ec2-user ec2-user 0 Oct  6 11:51 appserver.pem
[ec2-user@ip-10-1-1-155 ~]$
chmod +w appserver.pem
[ec2-user@ip-10-1-1-155 ~]$
sudo vi appserver.pem
[ec2-user@ip-10-1-1-155 ~]$
sudo ssh -i appserver.pem ec2-user@10.1.2.222
```

The authenticity of host '10.1.2.222 (10.1.2.222)' can't be established.

ED25519 key fingerprint is SHA256:FVW12hvfOTsFvxCOflmKVST38qUjmCnIPrgPRJKZIMU.

This key is not known by any other names

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Warning: Permanently added

```
sudo yum install -y httpd
sudo systemctl start httpd
sudo systemctl enable httpd
```

App Server

- Amazon Linux 2 ami
- T2 Micro
- Use VPC & public subnet
- User Data - to allow access to the RDS Database

```
#!/bin/bash
sudo yum install -y mariadb
sudo service mariadb start
```

Create DB Instance

- Created a subnet group
- DB Instance
 - Standard create
 - Mariadb
 - Free tier

```
user = <username>
password = <password>
initial Database: mydb
```

'10.1.2.222' (ED25519) to the list of known hosts.

```
[ec2-user@ip-10-1-2-222 ~]
$ mysql -u root -h <hostname> -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 51
Server version: 10.11.8-MariaDB
DB managed by https://aws.amazon.com/rds/
```

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation AB and others.

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

```
MariaDB [(none)]> show databases;
```

```
+-----+
| Database          |
+-----+
| information_schema |
| innodb             |
| mysql              |
| performance_schema |
| sys                |
+-----+
5 rows in set (0.001 sec)
```

```
MariaDB [(none)]>
```

Web Server HTTP Connection

