**Notes:**

Create a VPC,

1 Public Subnet,

3 Private Subnets,

an Internet Gateway,

a NAT Gateway,

and lastly the Route Tables, in that order.

You can use 192.168.0.0/16 for the VPC and staring at 192.168.1.0/24 and so on for the subnets. The last Private Subnet must be in another availability zone so it can be used for the Subnet Group. Make sure to edit the DNS hostnames on the VPC and edit subnet settings by enabling the auto-assign public IPv4 addresses.

After that create the need security groups listed as such:

* **Bastion Server Security Group**
  + Inbound Rules
    - Rule: Type: SSH | Protocol: TCP | Port Range: 22 | Source: MyIP
    - Rule: Type: HTTP | Protocol: TCP | Port Range: 80 | Source: Anywhere – IPv4
    - Rule: Type: HTTPS | Protocol: TCP | Port Range: 443 | Source: Anywhere – IPv4
    - Rule: Type: HTTP | Protocol: TCP | Port Range: 80 | Source: Anywhere – IPv6
    - Rule: Type: HTTPS | Protocol: TCP | Port Range: 443 | Source: Anywhere – IPv6
    - Rule: Type: MYSQL/Aurora | Protocol: TCP | Port Range: 3306 | Source: MyDatabaseServerSG
* **Web Server Security Group**
  + Inbound Rules
    - Rule: Type: SSH | Protocol: TCP | Port Range: 22 | Source: MyIP
    - Rule: Type: HTTP | Protocol: TCP | Port Range: 80 | Source: Anywhere – IPv4
    - Rule: Type: HTTPS | Protocol: TCP | Port Range: 443 | Source: Anywhere – IPv4
    - Rule: Type: HTTP | Protocol: TCP | Port Range: 80 | Source: Anywhere – IPv6
    - Rule: Type: HTTPS | Protocol: TCP | Port Range: 443 | Source: Anywhere – IPv6
    - Rule: Type: All ICMP – IPv4 | Protocol: ICMP | Port Range 443 | Source: MyAppServerSG
* **App Server Security Group**
  + Inbound Rules
    - Rule: Type: SSH | Protocol: TCP | Port Range: 22 | Source: MyBastionServerSG
    - Rule: Type: HTTP | Protocol: TCP | Port Range: 80 | Source: Anywhere – IPv4
    - Rule: Type: HTTPS | Protocol: TCP | Port Range: 443 | Source: Anywhere – IPv4
    - Rule: Type: HTTP | Protocol: TCP | Port Range: 80 | Source: Anywhere – IPv6
    - Rule: Type: HTTPS | Protocol: TCP | Port Range: 443 | Source: Anywhere – IPv6
    - Rule: Type: All ICMP – IPv4 | Protocol: ICMP | Port Range 443 | Source: MyWebServerSG
    - Rule: Type: MYSQL/Aurora | Protocol: TCP | Port Range: 3306 | Source: MyDatabaseServerSG
* **Database Server Security Group**
  + Inbound Rules
    - Rule: Type: MYSQL/Aurora | Protocol: TCP | Port Range: 3306 | Source: MyBastionServerSG
    - Rule: Type: MYSQL/Aurora | Protocol: TCP | Port Range: 3306 | Source: MyAppServerSG

EC2

Then create an EC2 instance for a bastion host, a web server and an app server, in the AZs depicted above. You can use a UserData - Web.txt file to setup the web server and a UserData – App.txt file to setup the app server.

***Example for the UserData - Web.txt script:***

#!/bin/bash

# Install Apache Web Server

yum install -y httpd

# Turn on web server

systemctl enable httpd.service

systemctlstart httpd.service

# Setup Web Server

chmod 755 var/www/html

touch var/www/html/index.html

tee -a var/www/html/index.html << END

<!DOCTYPE html>

<html>

<body>

<h1>Welcome to my website.</h1>

<p>If you can see this then the force is with you.</p>

</body>

</html>

END

***Example for the UserData - App.txt script:***

#!/bin/bash

# Install MariaDB Server

yum install -y mariadb-server

# Turn on database server

service mariadbstart

**RDS**

Then create a subnet group and a RDS instance for the database server, in the AZs depicted above. Here are the specifications you can use:

* Subnet Group
  + Name: mydatabasesubnetgroup
  + Description: My Database Subnet Group
  + VPC: Demo VPC
  + Availability Zones:
    - us-west-2a
    - us-west-2b
  + Subnets:
    - Private Subnet 2: 192.168.3.0/24
    - Private Subnet 3: 192.168.4.0/24
* Database Server
  + Creation method: Standard Create
  + Engine: MariaDB
  + Templates: Free tier
  + DB Instance Identifier: DBInstance
  + Master Username: root
  + Master User Password: Re:Start!9
  + DB Instance Class: db.t2.micro
  + Allocated Storage: 20
  + VPC: Demo VPC
  + Subnet Group: mydatabasesubnetgroup
  + Public Access: No
  + VPC Security Group: MyDatabaseServerSG
  + Availability Zone: us-west-2a
  + Initial database name: mydb (You can add one just to help show the connection to this database server.)

To finish out the demo open a new browser tab with the Public IP/DNS of the Web Server and check the App Server’s connection to the Web Server and the Database Server by using the Bastion Host. Here are the commands one can use to show the connections:

* Setup
  + Load needed files for setup from your local device to the Bastion Server instance
    - Window Users:
      * pscp -scp -P 22 -i '.\Downloads\labsuser.ppk' -l user ec2-user '.\Downloads\labsuser.pem' [ec2-user@](about:blank)bastion-host-public-ip[:/home/ec2-user](about:blank)
    - Mac and Linux Users:
      * chmod 400 labsuser.pem
      * sudoscp -i '.\Downloads\labsuser.pem' -l user ec2-user '.\Downloads\labsuser.pem' ec2-user@bastion-host-public-ip:/home/ec2-user
  + Connect to the Bastion Server instance by using SSH
    - Window Users:
      * Use PUTTY to connect to the EC2 instance.
    - Mac and Linux Users:
      * ssh -ilabsuser.pem ec2-user@bastion-host-public-ip
  + Testing the connection to the App Server
    - chmod 400 labsuser.pem
    - ssh -ilabsuser.pem ec2-user@app-server-private-ip
    - ping web-server-private-ip
  + Testing the connection to the Database Server
    - mysql --user=root --password='Re:Start!9' --host=database-server-endpoint mydb
    - show databases;
    - exit (Use this command multiple times till the Bastion Host’s ssh connection is closed.)