Project Report: Lift and Shift Application Workload

Title: Lift and Shift Application Workload

Overview:

The project focuses on lifting and shifting a multi-tier web application stack (VPROFILE) to the AWS Cloud for production. This strategy involves moving application services running on physical or virtual machines in a data center to the cloud.

Scenario:

Currently, the application services are running on physical/virtual machines with all workloads managed in your data center. The existing setup requires involvement from various teams, including virtualization, DC OPS, monitoring, and system administration.

Problems:

1. **Complex Management:** Managing the current infrastructure is complex and challenging.
2. **Scale Up/Down Complexity:** Scaling the infrastructure up or down as per demand is complicated.
3. **Upfront CapEx and Regular OpEx:** There are significant upfront capital expenditures and ongoing operational expenses.
4. **Manual Processes:** Many processes are manual, leading to inefficiencies.
5. **Difficult to Manage:** The overall system is hard to manage.
6. **Time Consuming:** Managing the infrastructure is time-consuming.

Solution:

Cloud Setup:

1. **Automation:** Automate processes to reduce manual intervention.
2. **Pay-As-You-Go:** Only pay for the resources used.
3. **Infrastructure as a Service (IAAS):** Utilize IAAS for flexible infrastructure.
4. **Flexibility:** Easily adapt to changing requirements.
5. **Ease to Manage:** Simplify management with cloud solutions.

AWS Solution:

AWS Services:

- **EC2 Instances:** Virtual machines for Tomcat, RabbitMQ, Memcached, and MySQL.
- **ELB (Elastic Load Balancer):** Replacement for Nginx Load Balancer.
- **Auto Scaling:** Automate the scaling of virtual machines.
- **S3/EFS Storage:** Use S3 or EFS for shared storage.
- **Route 53:** Private DNS service.

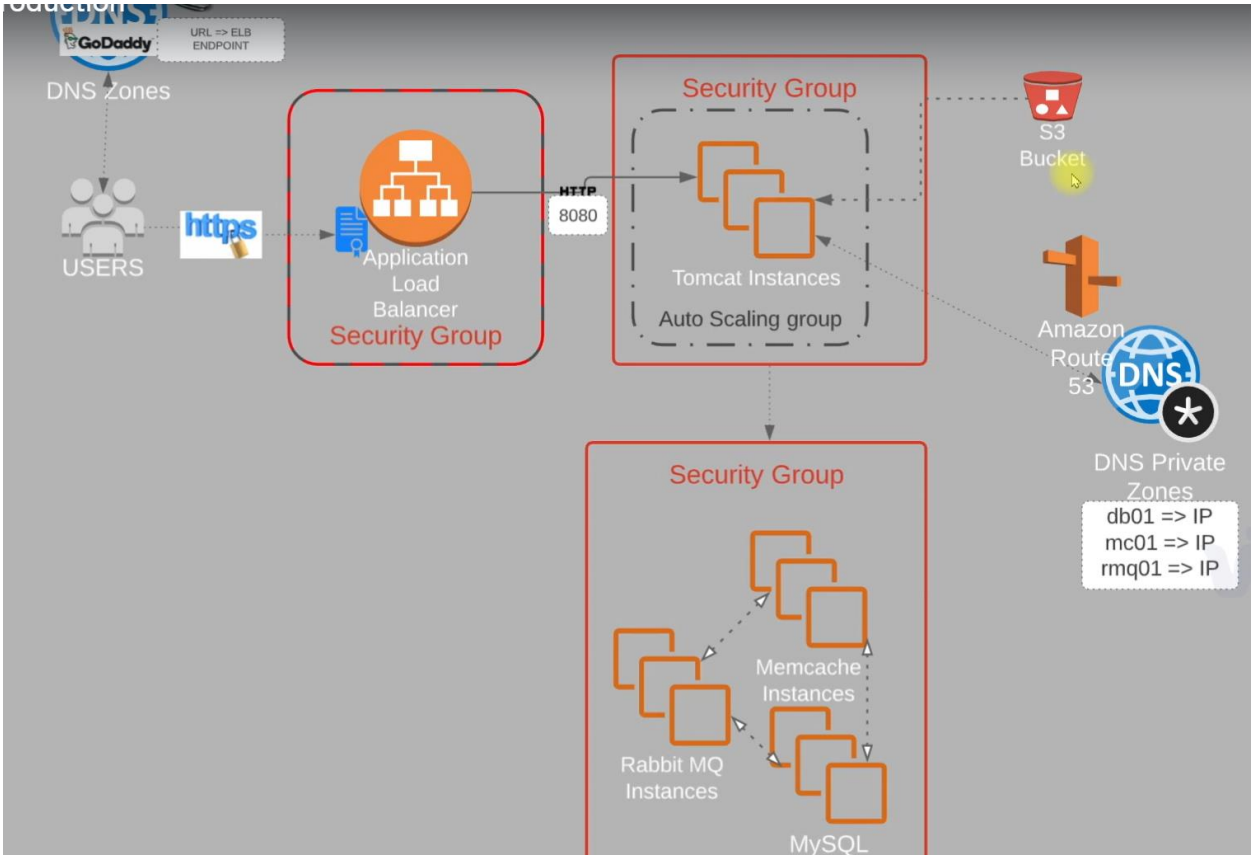
Objective:

1. **Flexible Infrastructure:** Create a flexible and scalable infrastructure.
2. **No Upfront Cost:** Eliminate upfront capital expenditures.
3. **Modernize Effectively:** Update and modernize the application infrastructure effectively.
4. **Infrastructure as Code (IAAC):** Implement IAAC for managing infrastructure.

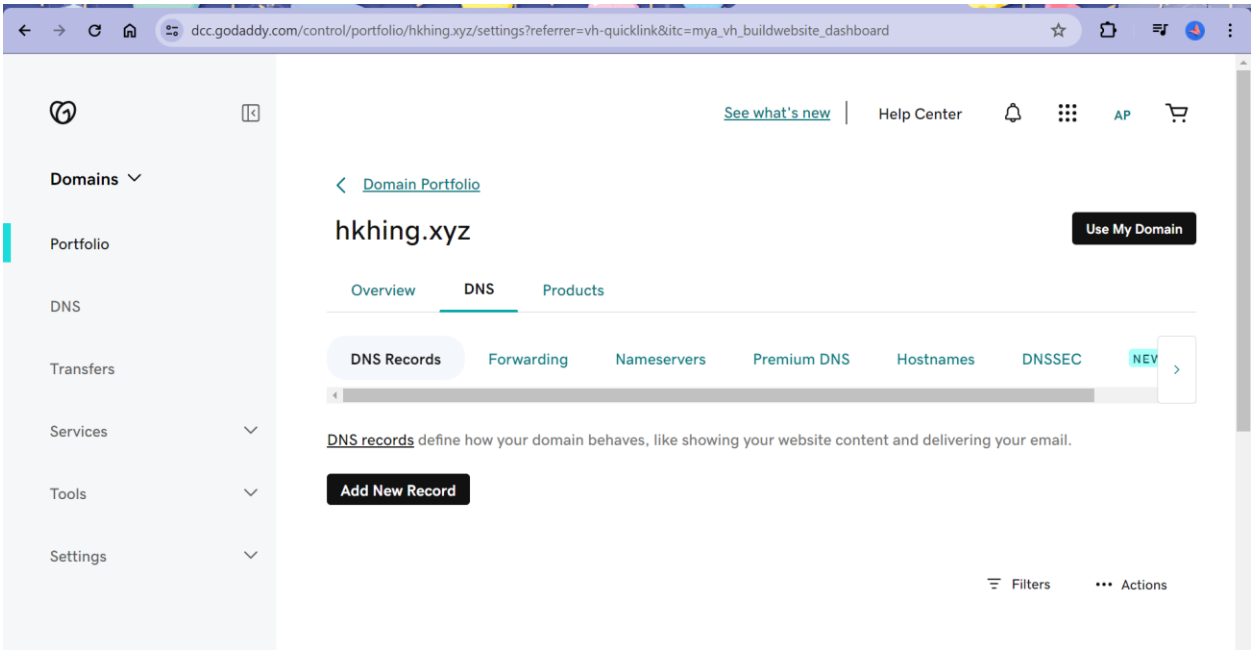
Flow:

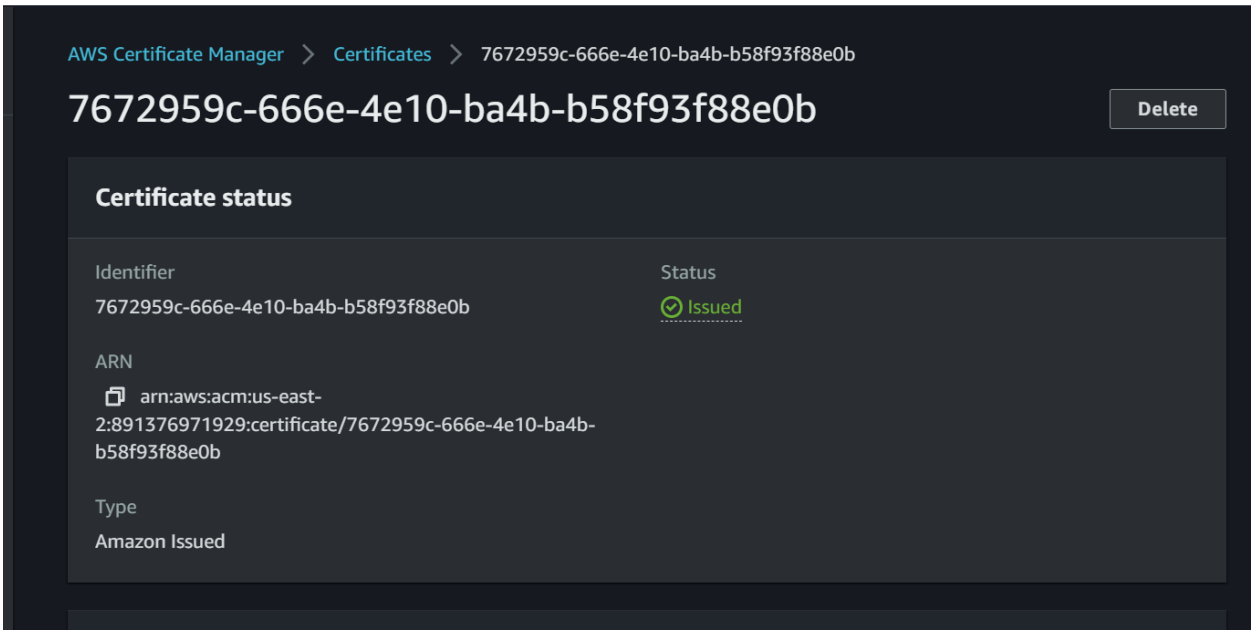
1. **Login to AWS Account:** Access the AWS Management Console.
2. **Create Key Pairs:** Generate key pairs for secure access to instances.
3. **Create Security Groups:** Define security groups to control access to instances.
4. **Launch Instances with User Data (Bash Scripts):** Launch EC2 instances and execute initialization scripts.
5. **Update IP to Name Mapping in Route 53:** Configure DNS settings in Route 53.
6. **Build Application from Source Code:** Compile the application from the source code.
7. **Upload to S3 Bucket:** Upload the application artifacts to an S3 bucket.
8. **Download Artifact to Tomcat EC2 Instance:** Retrieve the application artifacts on the Tomcat instance.
9. **Setup ELB with HTTPS (Cert from Amazon Certificate Manager):** Configure the Elastic Load Balancer with HTTPS using a certificate from Amazon Certificate Manager.
10. **Map ELB Endpoint to Website Name in GoDaddy DNS:** Update the DNS settings in GoDaddy to point to the ELB endpoint.
11. **Verify:** Ensure that the setup works as expected and verify the application is accessible.

Pics:

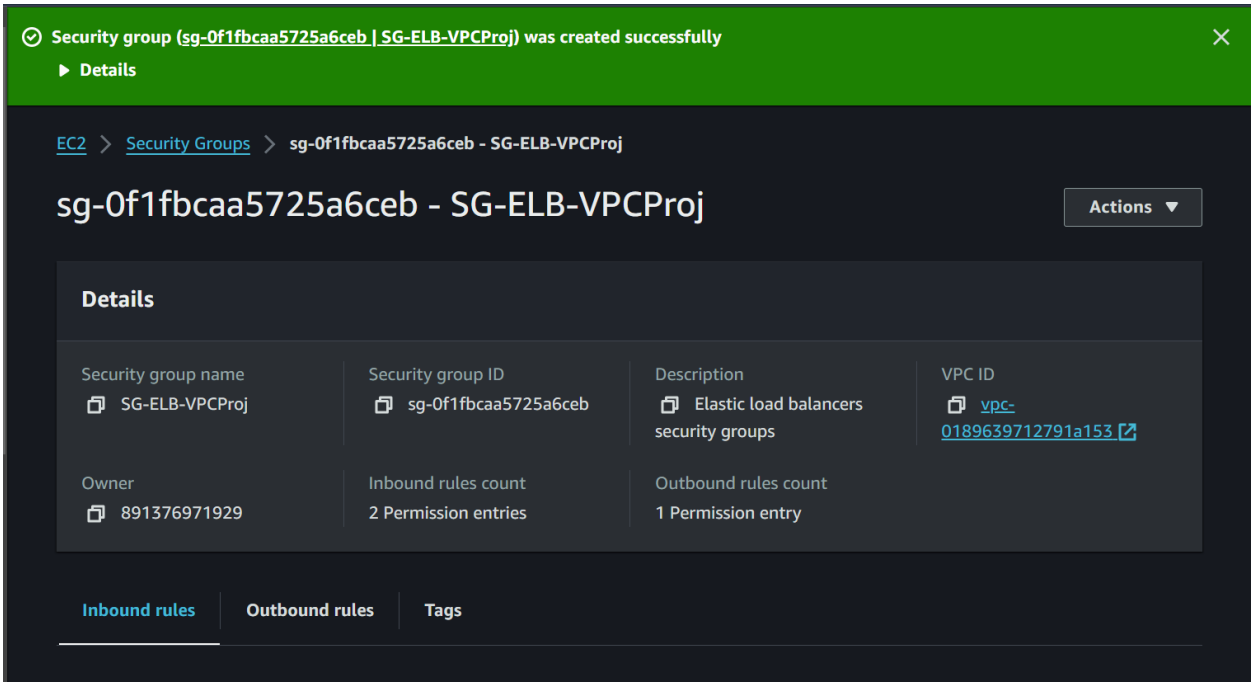

GoDaddy
URL => ELB ENDPOINT
DNS Zones
USERS
https
Application Load Balancer
Security Group
HTTP
8080
Security Group
Tomcat Instances
Auto Scaling group
S3 Bucket
Amazon Route 53
DNS
DNS Private Zones
db01 => IP
mc01 => IP
rmq01 => IP
Security Group
Memcache Instances
Rabbit MQ Instances
MySQL

1) Get AWS Certificate and validate it using GoDaddy.

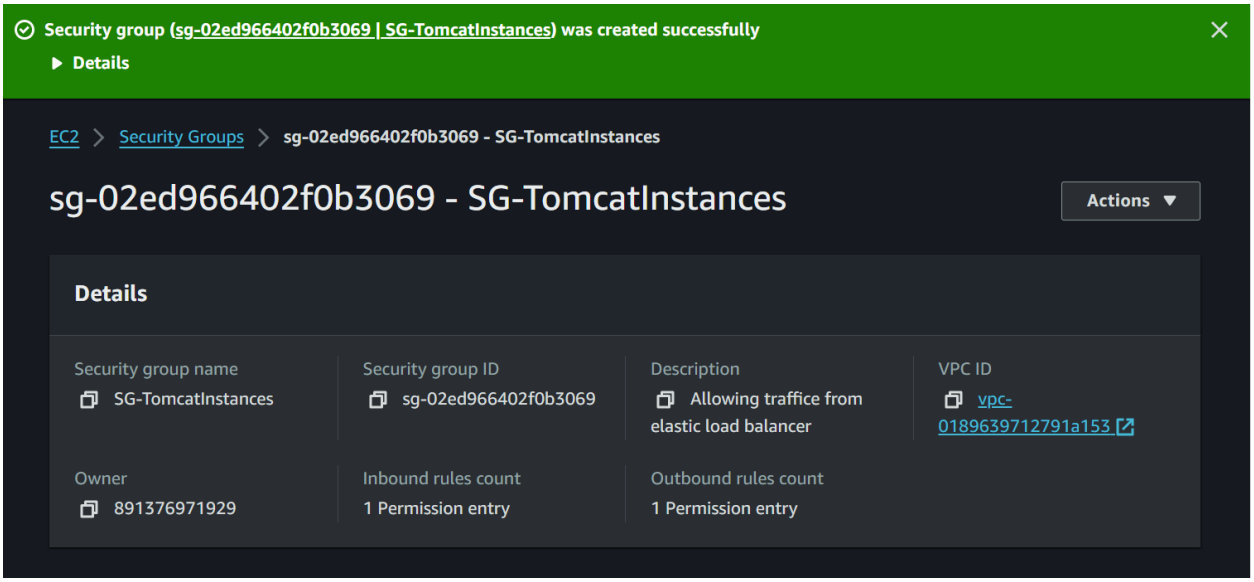




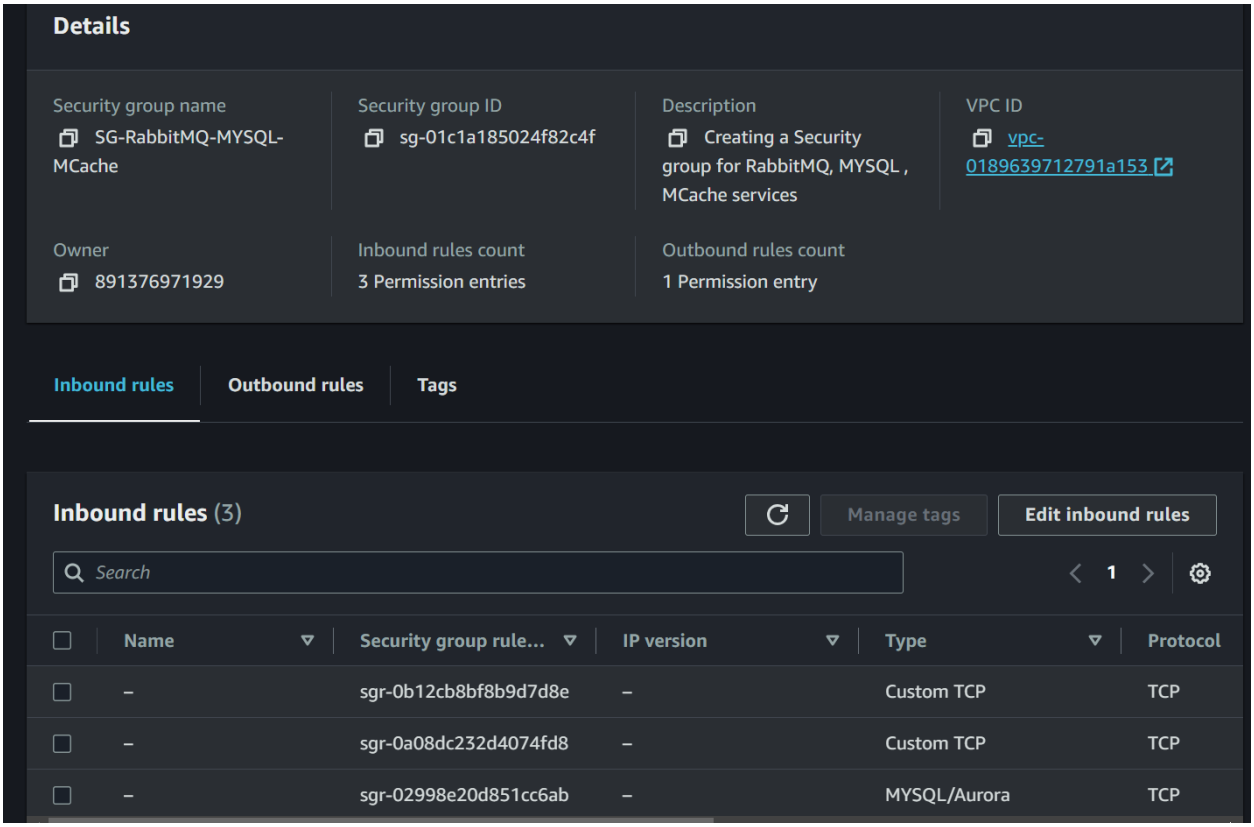
2) Create Security groups for load balanacers

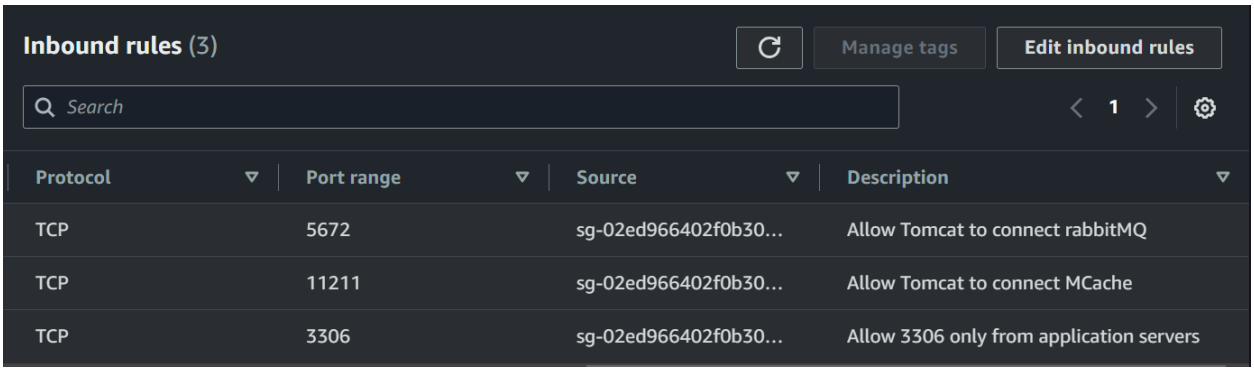


3) Create security group for tomcat instances

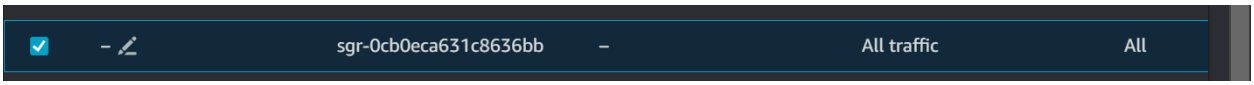


4) Create security groups for MySql, RabbitMQ and MCache

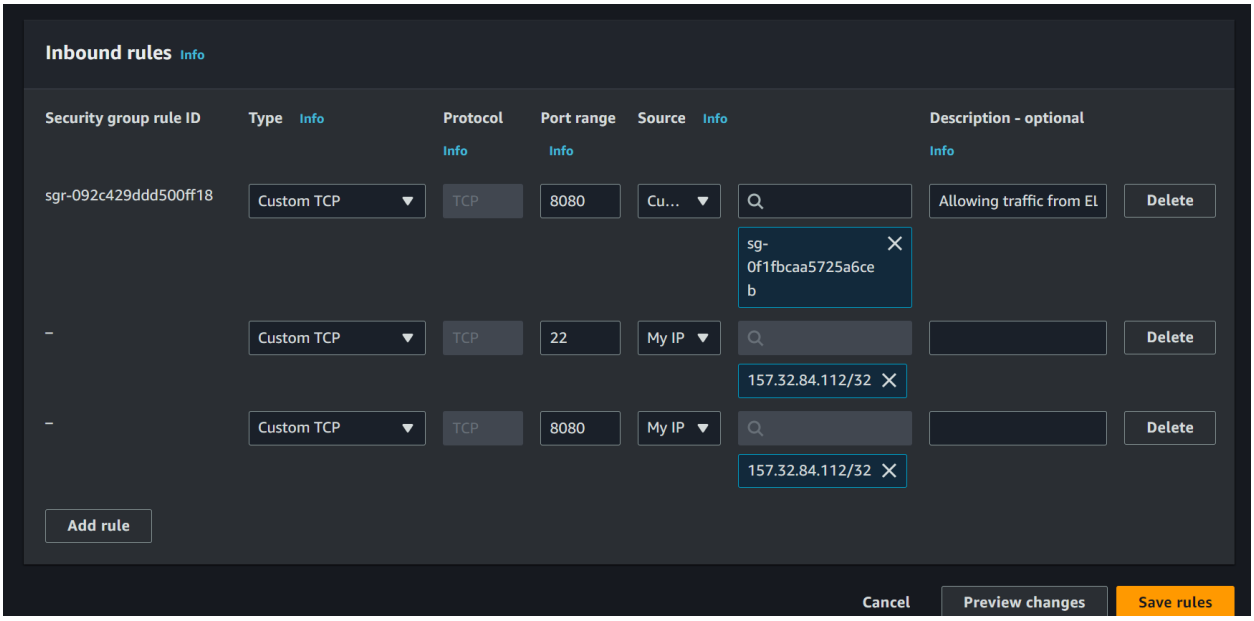


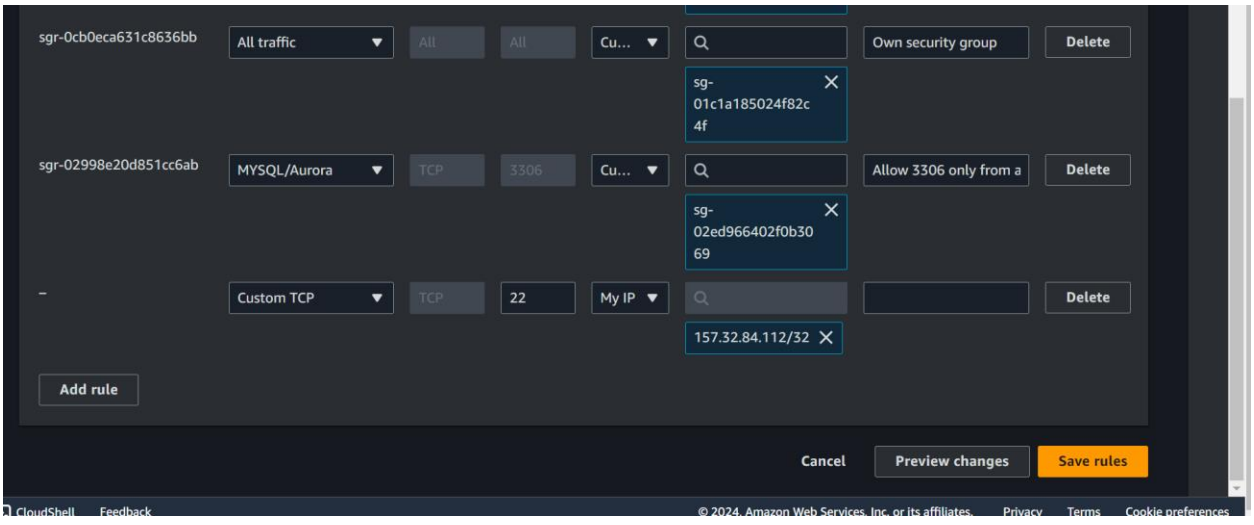


5) In our backend all three services need to connect, so let us create a port which allow them to connect

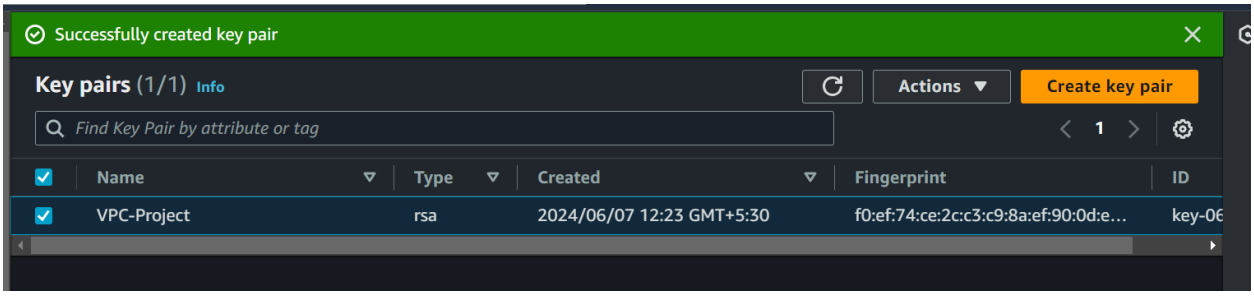


6) Add 22 from my IP in both security groups for SSH

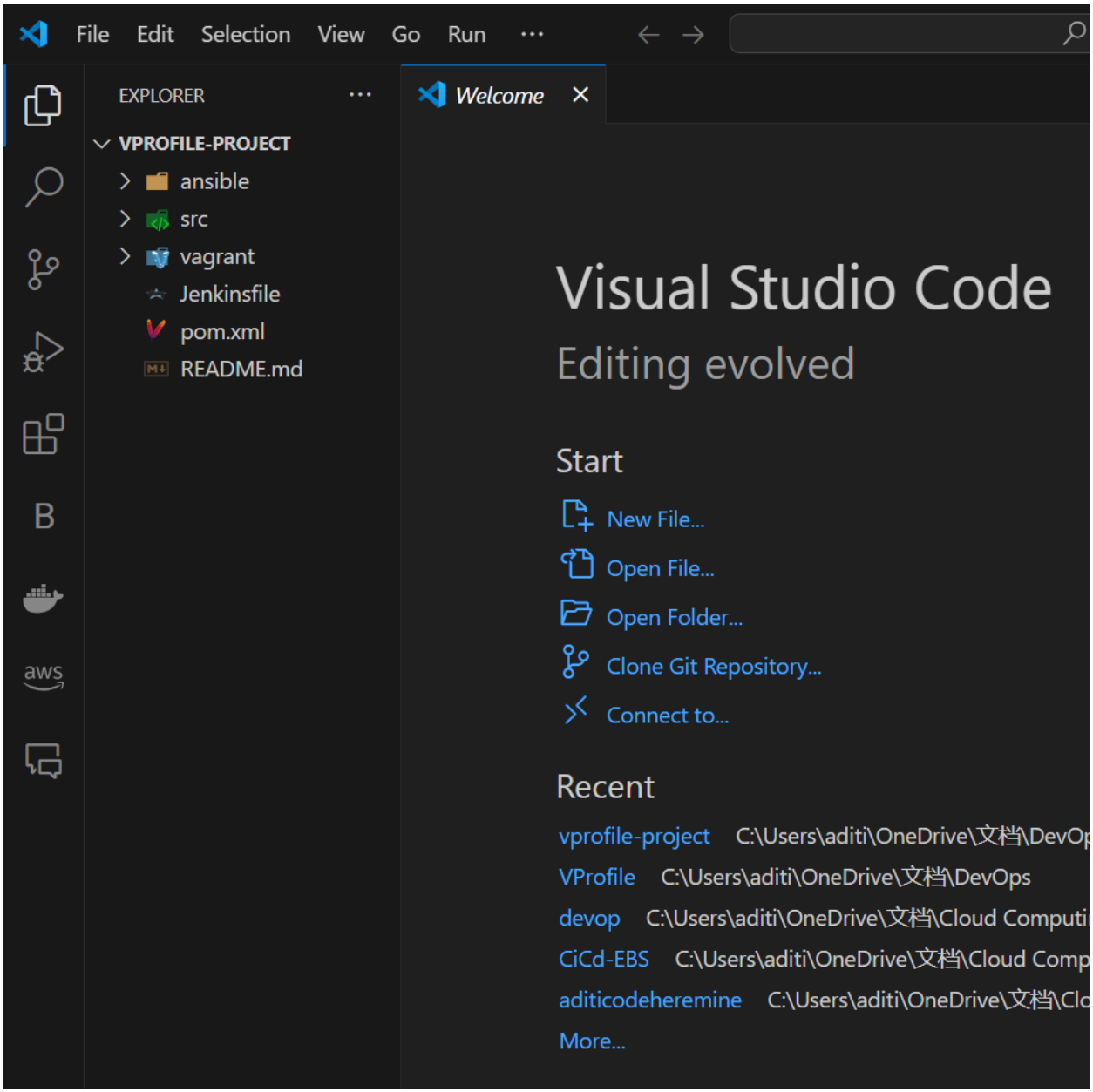


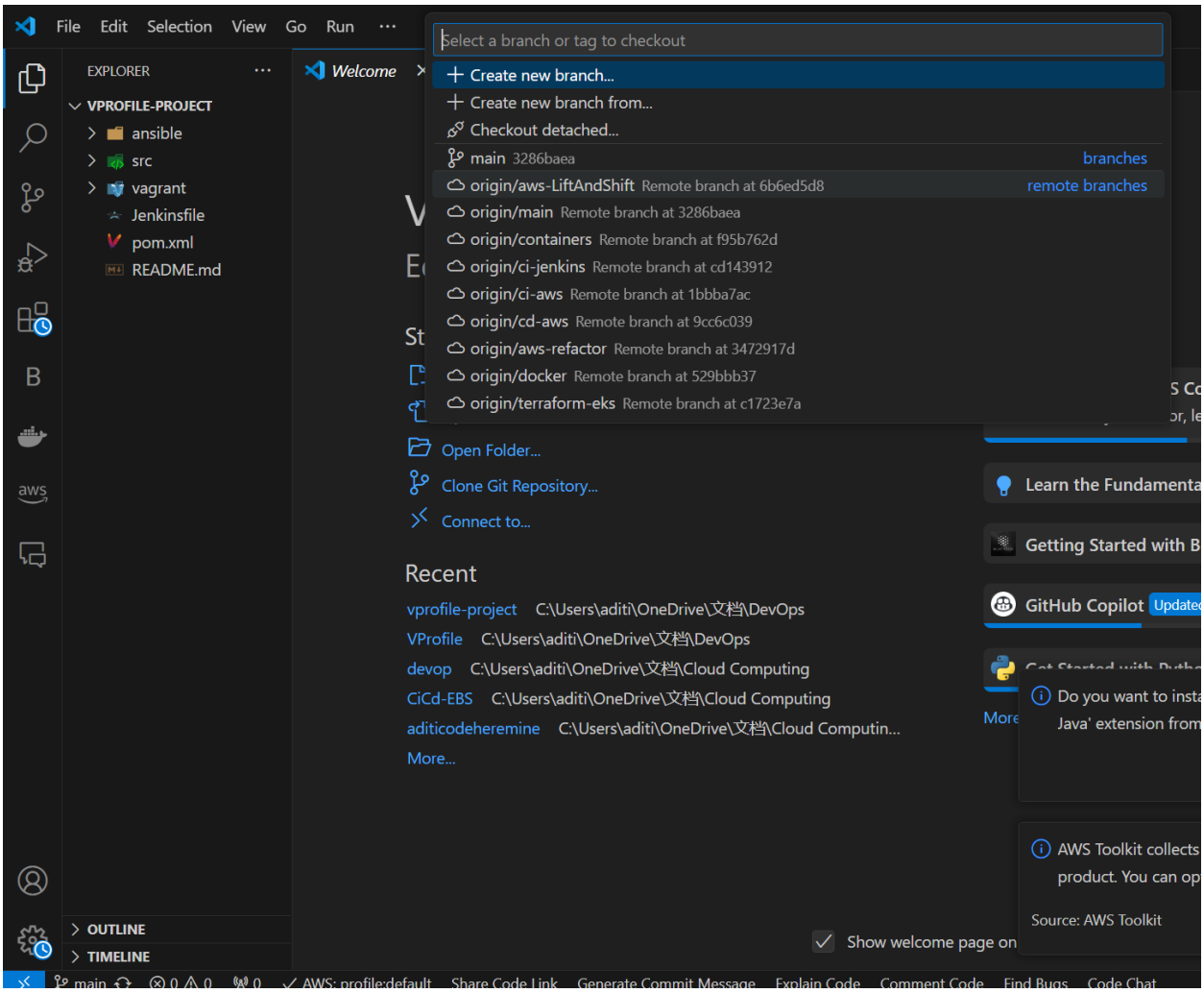


7) Create a Login Key Pair

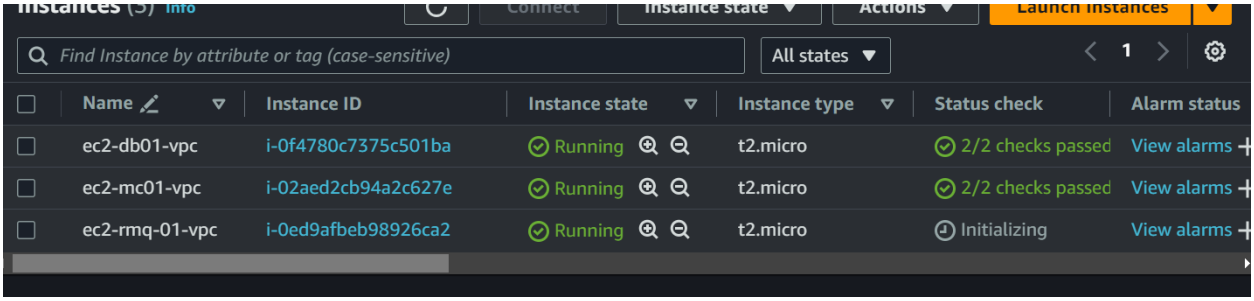


8) Clone the source code of the project provided by development team

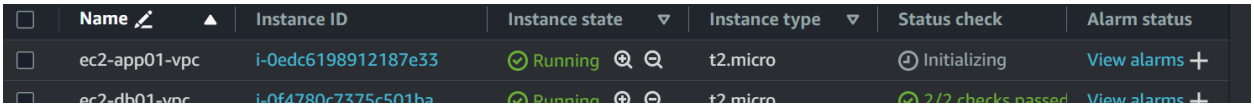

File
Edit
Selection
View
Go
Run
EXPLORER
VPROFILE-PROJECT
ansible
src
vagrant
Jenkinsfile
pom.xml
README.md
Welcome
Visual Studio Code
Editing evolved
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
Recent
vprofile-project C:\Users\aditi\OneDrive\文档\DevOp
VProfile C:\Users\aditi\OneDrive\文档\DevOps
devop C:\Users\aditi\OneDrive\文档\Cloud Computi
CiCd-EBS C:\Users\aditi\OneDrive\文档\Cloud Comp
aditicodeheremine C:\Users\aditi\OneDrive\文档\Clo
More...



9) Launch Instance for db01 which is MySQL , mc01 for MCache, rmq01 for RabbitMQ belongs to backend security group



10) Launch Instance for Application : Tomcat --- app01



Name	Instance ID	Instance state	Instance type	Status check	Alarm status
ec2-app01-vpc	i-0edc6198912187e33	Running	t2.micro	Initializing	View alarms
ec2-db01-vpc	i-0f4780c7375c501ba	Running	t2.micro	2/2 checks passed	View alarms

11) Check health and status

```
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-13-139:~$ sudo -i
root@ip-172-31-13-139:~# systemctl status mariadb
● mariadb.service - MariaDB 10.11.7 database server
     Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; preset: >
     Active: active (running) since Tue 2024-06-11 12:02:45 UTC; 4min 42s ago
       Docs: man:mariadbd(8)
             https://mariadb.com/kb/en/library/systemd/
    Process: 13897 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /va>
    Process: 13899 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_S>
    Process: 13901 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ] &&>
    Process: 13974 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_>
    Process: 13977 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0>
   Main PID: 13961 (mariadbd)
     Status: "Taking your SQL requests now..."
      Tasks: 10 (limit: 7463)
     Memory: 84.6M (peak: 87.6M)
        CPU: 485ms
     CGroup: /system.slice/mariadb.service
             └─13961 /usr/sbin/mariadbd

Jun 11 12:02:45 ip-172-31-13-139 mariadbd[13961]: 2024-06-11 12:02:45 0 [Note] >
```

12) Check if you can access Maria db

```
root@ip-172-31-13-139:~# mysql -u admin -padmin123 accounts
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 10.11.7-MariaDB-2ubuntu2 Ubuntu 24.04

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 [accounts]> show tables;
+--------------------+
| Tables_in_accounts |
+--------------------+
| role               |
| user               |
| user_role          |
+--------------------+
3 rows in set (0.000 sec)

MariaDB [accounts]>
```

13) Check connectivity of Memechace

```
aditi@ADITI MINGW64 ~/downloads
$ ssh -i vprofile-key-pair.pem ubuntu@3.133.132.163
The authenticity of host '3.133.132.163 (3.133.132.163)' can't be established.
ED25519 key fingerprint is SHA256:So+8eXdVidHOhrXF+dbvcxZfyXEMP59Ed0wqa69/MP4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.133.132.163' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1008-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Tue Jun 11 12:17:41 UTC 2024

  System load:  0.0               Processes:             106
  Usage of /:   26.3% of 6.71GB   Users logged in:       0
  Memory usage: 24%               IPv4 address for enX0: 172.31.2.248
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

82 updates can be applied immediately.
48 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-2-248:~$ sudo -i
root@ip-172-31-2-248:~# ss -tunlp | grep 11211
tcp   LISTEN 0      1024            0.0.0.0:11211      0.0.0.0:*    users:(("me
mcached",pid=1708,fd=26))
tcp   LISTEN 0      1024               [::1]:11211         [::]:*    users:(("me
mcached",pid=1708,fd=27))
root@ip-172-31-2-248:~#
```

14) Check the connectivity of RabbitMQ

```
o run a command as administrator (user "root"), use "sudo <command>".
ee "man sudo_root" for details.

buntu@ip-172-31-13-233:~$ sudo -i
oot@ip-172-31-13-233:~# systemctl status rabbitmq-server
 rabbitmq-server.service - RabbitMQ broker
    Loaded: loaded (/usr/lib/systemd/system/rabbitmq-server.service; enabled; >
    Active: active (running) since Tue 2024-06-11 12:05:51 UTC; 15min ago
  Main PID: 5520 (beam.smp)
     Tasks: 50 (limit: 1130)
    Memory: 93.7M (peak: 147.9M)
       CPU: 6.658s
    CGroup: /system.slice/rabbitmq-server.service
            ├─5520 /usr/lib/erlang/erts-13.2.2.5/bin/beam.smp -W w -MBas ageff>
            ├─5530 erl_child_setup 32768
            ├─5575 /usr/lib/erlang/erts-13.2.2.5/bin/inet_gethost 4
            ├─5576 /usr/lib/erlang/erts-13.2.2.5/bin/inet_gethost 4
            └─5579 /bin/sh -s rabbit_disk_monitor

un 11 12:05:49 ip-172-31-13-233 rabbitmq-server[5520]:   Release series suppor>
un 11 12:05:49 ip-172-31-13-233 rabbitmq-server[5520]:   Doc guides:  https://>
un 11 12:05:49 ip-172-31-13-233 rabbitmq-server[5520]:   Support:     https://>
un 11 12:05:49 ip-172-31-13-233 rabbitmq-server[5520]:   Tutorials:   https://>
un 11 12:05:49 ip-172-31-13-233 rabbitmq-server[5520]:   Monitoring:  https://>
un 11 12:05:49 ip-172-31-13-233 rabbitmq-server[5520]:   Logs: /var/log/rabbit>
un 11 12:05:49 ip-172-31-13-233 rabbitmq-server[5520]:         <stdout>
un 11 12:05:49 ip-172-31-13-233 rabbitmq-server[5520]:   Config file(s): /etc/>
un 11 12:05:51 ip-172-31-13-233 systemd[1]: Started rabbitmq-server.service - >
un 11 12:05:51 ip-172-31-13-233 rabbitmq-server[5520]:   Starting broker... co>
ines 1-24/24 (END)
```

15) Same for Tomcat

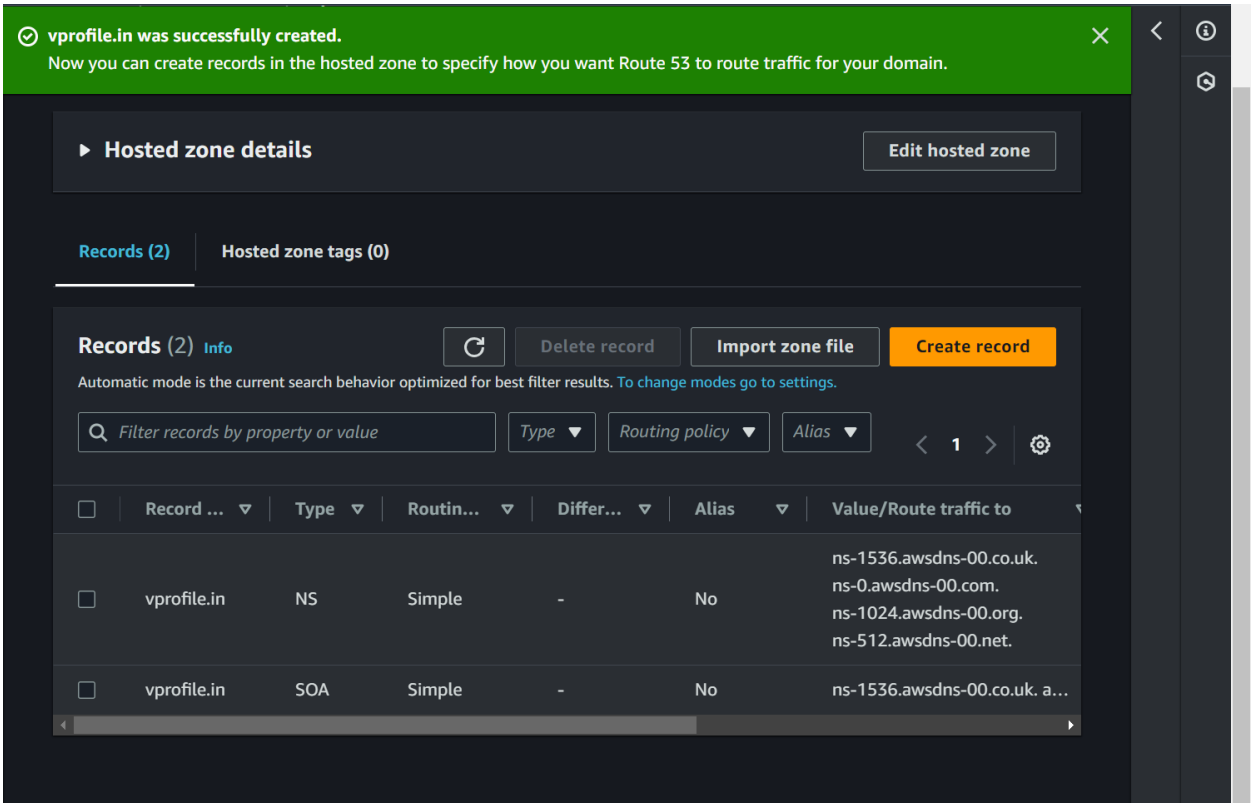
```
root@ip-172-31-3-224:~# systemctl status tomcat10
● tomcat10.service - Apache Tomcat 10 Web Application Server
     Loaded: loaded (/usr/lib/systemd/system/tomcat10.service; enabled; preset:>
     Active: active (running) since Tue 2024-06-11 12:47:55 UTC; 3min 7s ago
       Docs: https://tomcat.apache.org/tomcat-10.0-doc/index.html
    Process: 14421 ExecStartPre=/usr/libexec/tomcat10/tomcat-update-policy.sh (>
   Main PID: 14426 (java)
      Tasks: 28 (limit: 1130)
     Memory: 109.0M (peak: 114.3M)
        CPU: 9.191s
     CGroup: /system.slice/tomcat10.service
             └─14426 /usr/lib/jvm/java-11-openjdk-amd64/bin/java -Djava.util.lo>

Jun 11 12:48:03 ip-172-31-3-224 tomcat10[14426]: Deployment of deployment descr>
Jun 11 12:48:03 ip-172-31-3-224 tomcat10[14426]: Deploying deployment descripto>
Jun 11 12:48:03 ip-172-31-3-224 tomcat10[14426]: The path attribute with value >
Jun 11 12:48:05 ip-172-31-3-224 tomcat10[14426]: At least one JAR was scanned f>
Jun 11 12:48:05 ip-172-31-3-224 tomcat10[14426]: Deployment of deployment descr>
Jun 11 12:48:05 ip-172-31-3-224 tomcat10[14426]: Deploying web application dire>
Jun 11 12:48:06 ip-172-31-3-224 tomcat10[14426]: At least one JAR was scanned f>
Jun 11 12:48:06 ip-172-31-3-224 tomcat10[14426]: Deployment of web application >
Jun 11 12:48:06 ip-172-31-3-224 tomcat10[14426]: Starting ProtocolHandler ["htt>
Jun 11 12:48:06 ip-172-31-3-224 tomcat10[14426]: Server startup in [6403] milli>
lines 1-22/22 (END)
```

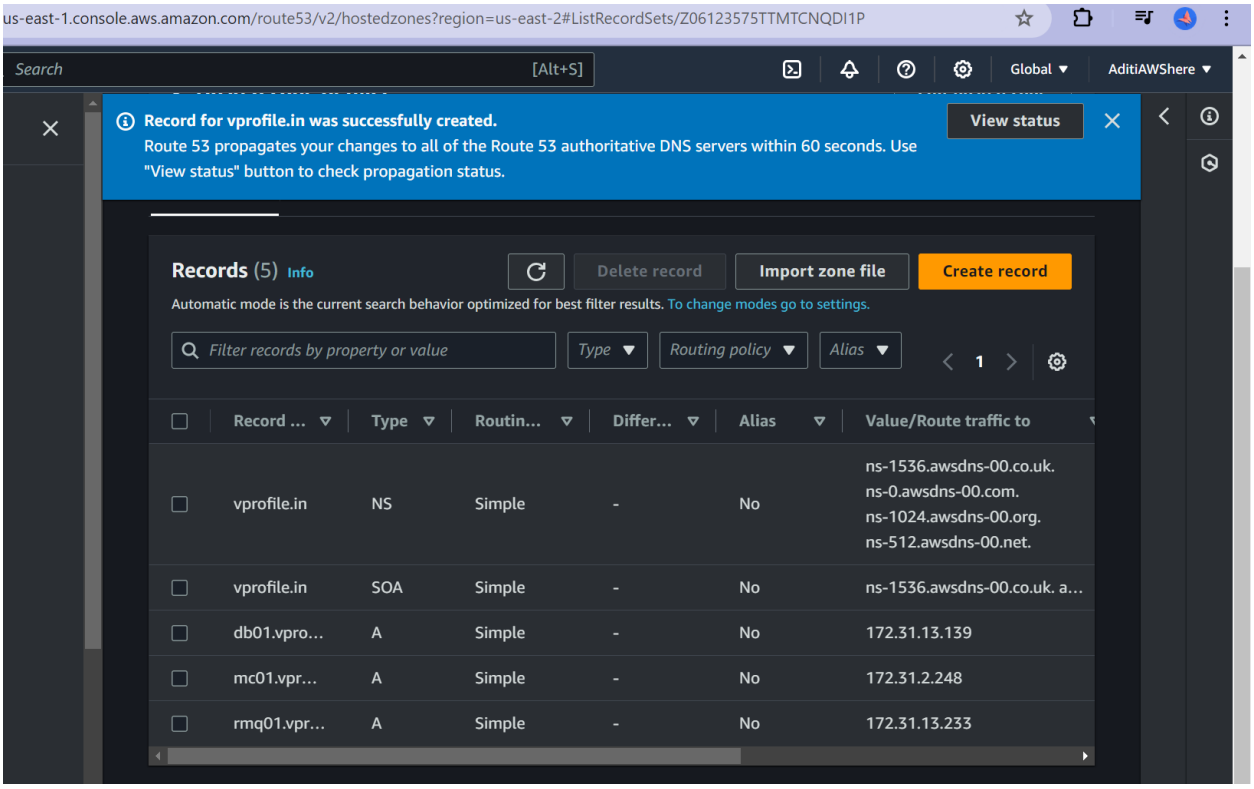
```
root@ip-172-31-3-224:~#
root@ip-172-31-3-224:~#
root@ip-172-31-3-224:~#
root@ip-172-31-3-224:~#
root@ip-172-31-3-224:~#
root@ip-172-31-3-224:~# ls /var/lib/tomcat10/
conf  lib  logs  policy  webapps  work
root@ip-172-31-3-224:~# ls /var/lib/tomcat10/webapps/
ROOT
root@ip-172-31-3-224:~#
```

16) Configure famous DNS called Route53

- We need to create the zones, which will be your domain name and than in that domain you will be having different hosts
- That host will be having Ip address or Cname

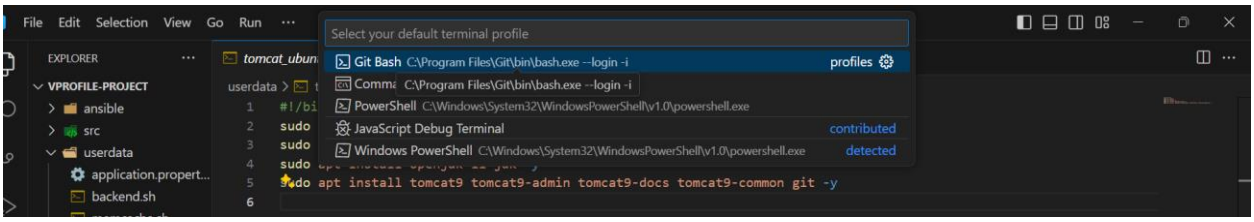


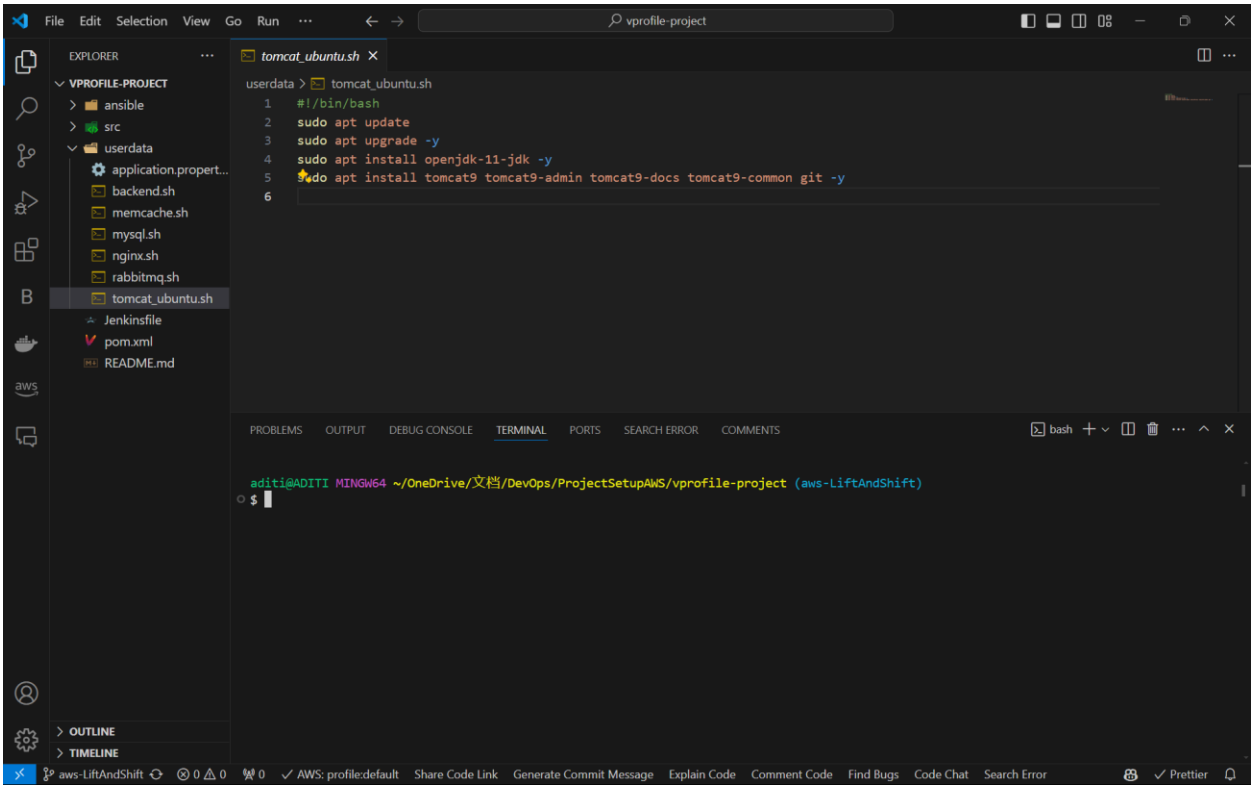
- Create different records
- We need these records so that the app server will connect to these backend services in application .properties files. Basically, it is a backend server record.



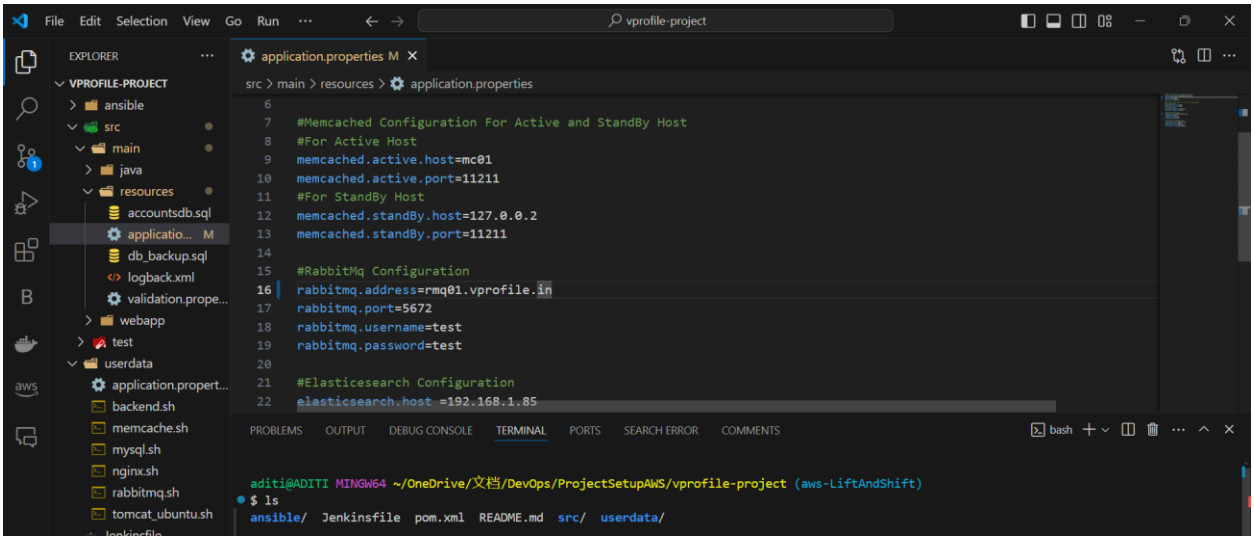
17) Now it is time to build our artifacts.

- Upload it to S3 Bucket, and from there, fetch to the EC2 instance to the tomcat EC2 instance.
- In VS code make git bash the default terminal





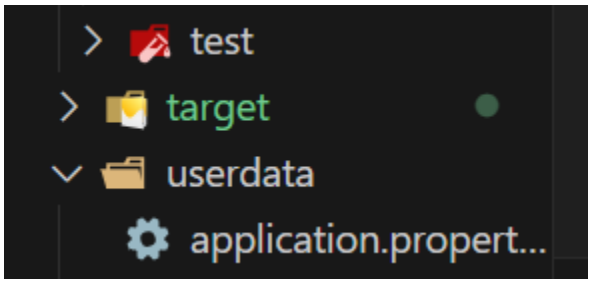
- Add .vprofile.in to db01 and rmq01.



- Check if Maven is configured properly

```
aditi@ADITI MINGW64 ~/OneDrive/文档/DevOps/ProjectSetupAWS/vprofile-project (aws-LiftAndShift)
$ mvn -version
Apache Maven 3.9.7 (8b094c9513efc1b9ce2d952b3b9c8eaedaf8cbf0)
Maven home: C:\ProgramData\chocolatey\lib\maven\apache-maven-3.9.7
Java version: 11.0.23, vendor: Amazon.com Inc., runtime: C:\Program Files\Amazon Corretto\jdk11.0.23_9
Default locale: en_US, platform encoding: Cp1252
OS name: "windows 11", version: "10.0", arch: "amd64", family: "windows"
```

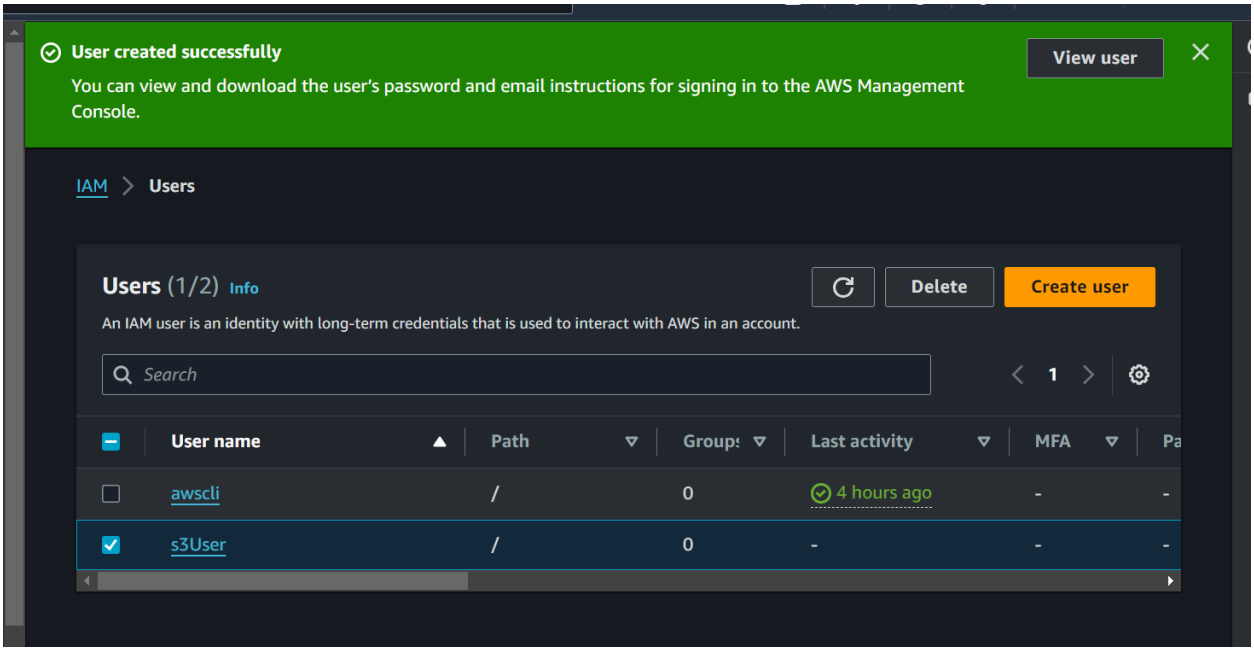
- Run command mvn install in the particular folder to build the artifact
- If build is successful you will be able to see target/ folder



- Now it is time to push the artifacts to S3 bucket hence we need an IAM role and s3 bucket.

```
[INFO]
[INFO] --- install:3.1.1:install (default-install) @ vprofile ---
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.5.0/plexus-utils-3.5.0.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.5.0/plexus-utils-3.5.0.jar (267 kB at 12
2 kB/s)
[INFO] Installing C:\Users\aditi\OneDrive\??\DevOps\ProjectSetupAWS\vprofile-project\pom.xml to C:\Users\aditi\.m2\repository\com\visualp
athit\vprofile\v2\vprofile-v2.pom
[INFO] Installing C:\Users\aditi\OneDrive\??\DevOps\ProjectSetupAWS\vprofile-project\target\vprofile-v2.war to C:\Users\aditi\.m2\reposit
ory\com\visualpathit\vprofile\v2\vprofile-v2.war
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  07:00 min
[INFO] Finished at: 2024-06-11T20:37:53+05:30
[INFO] ------------------------------------------------------------------------
```

- Create a user with s3 bucketfullAccess.

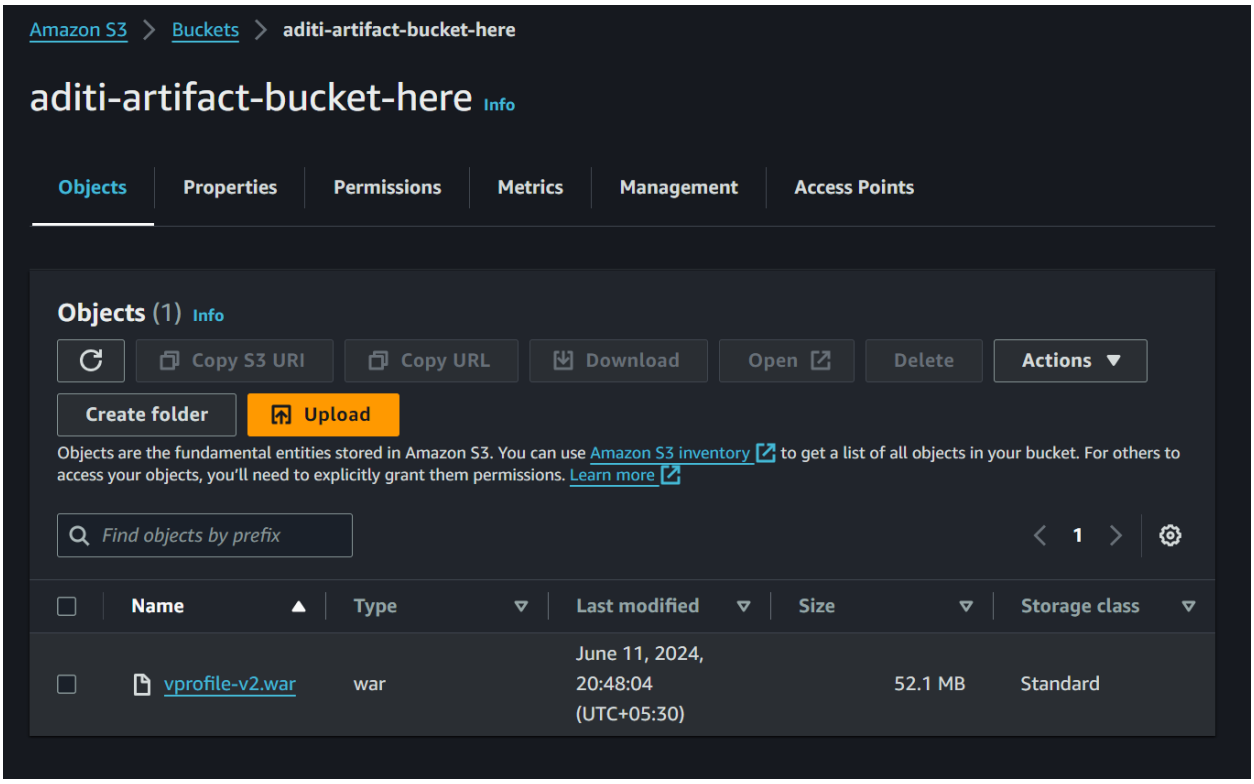


```
aditi@ADITI MINGW64 ~/OneDrive/文档/DevOps/ProjectSetupAWS/vprofile-project (aws-LiftAndShift)
$ aws s3 mb s3://aditi-artifact-bucket
make_bucket: aditi-artifact-bucket

aditi@ADITI MINGW64 ~/OneDrive/文档/DevOps/ProjectSetupAWS/vprofile-project (aws-LiftAndShift)
$
```

- Copy artifact to s3 bucket

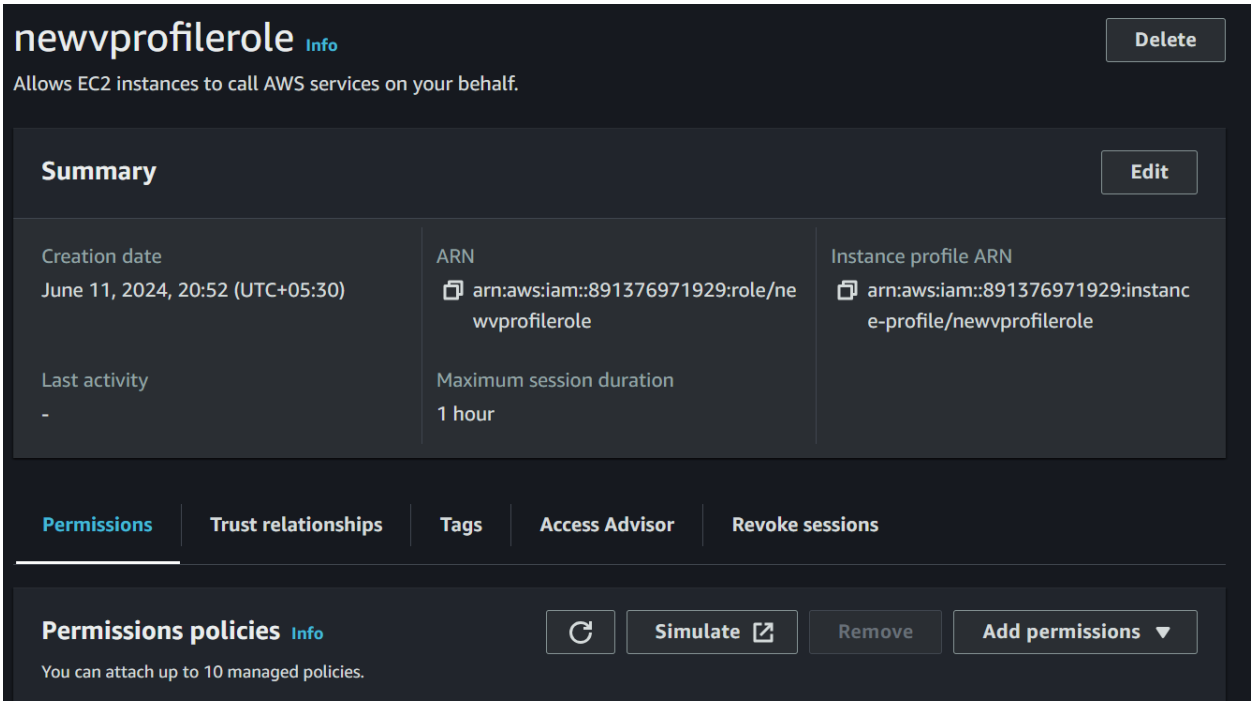
```
aditi@ADITI MINGW64 ~/OneDrive/文档/DevOps/ProjectSetupAWS/vprofile-project (aws-LiftAndShift)
$ aws s3 cp target/vprofile-v2.war s3://aditi-artifact-bucket-here/
Completed 512.0 KiB/52.1 MiB (91.8 KiB/s) with 1 file(s) remaining
```

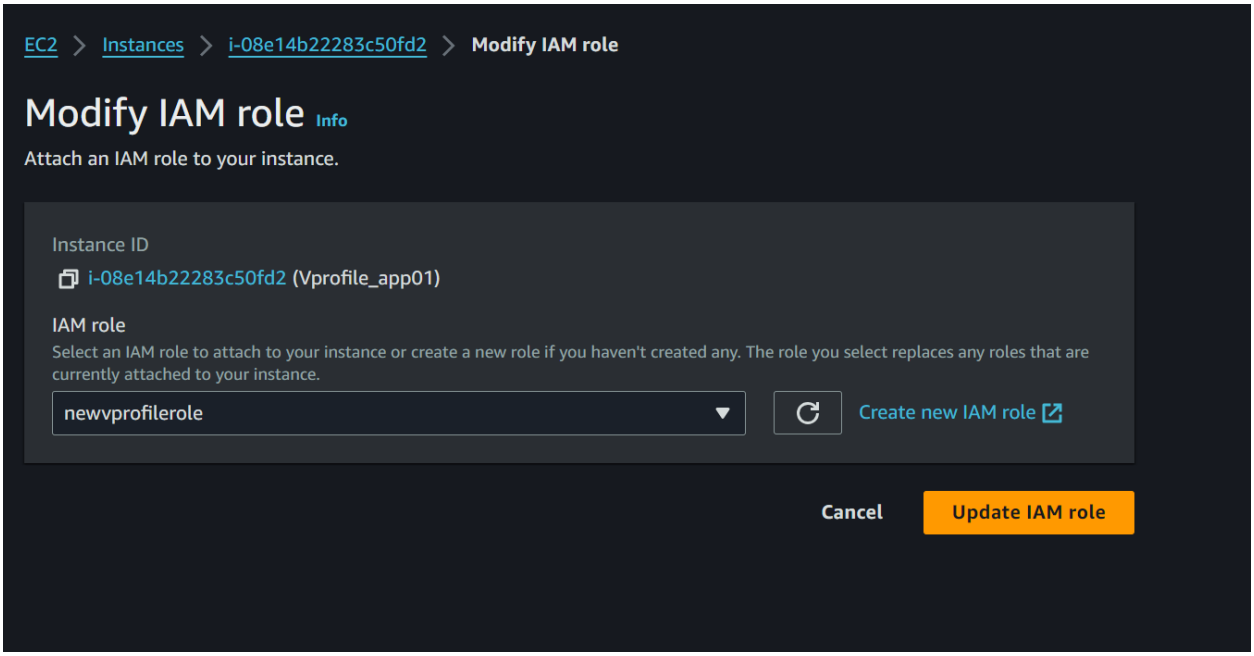


18) Final step :-

Download this artifact to our tomcat ec2 instance.

Go to IAM and go to roles and create the role and attach the role to instance :





- Run ssh for the instance

```
root@ip-172-31-3-224:~# apt update
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [12
6 kB]
Hit:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Pa
ckages [153 kB]
Get:6 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd6
4 Packages [61.4 kB]
Get:7 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [137
kB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [36.0
 kB]
Get:9 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [
42.1 kB]
Get:10 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en
[16.2 kB]
Fetched 699 kB in 1s (848 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
root@ip-172-31-3-224:~# apt install aws cli -y
```

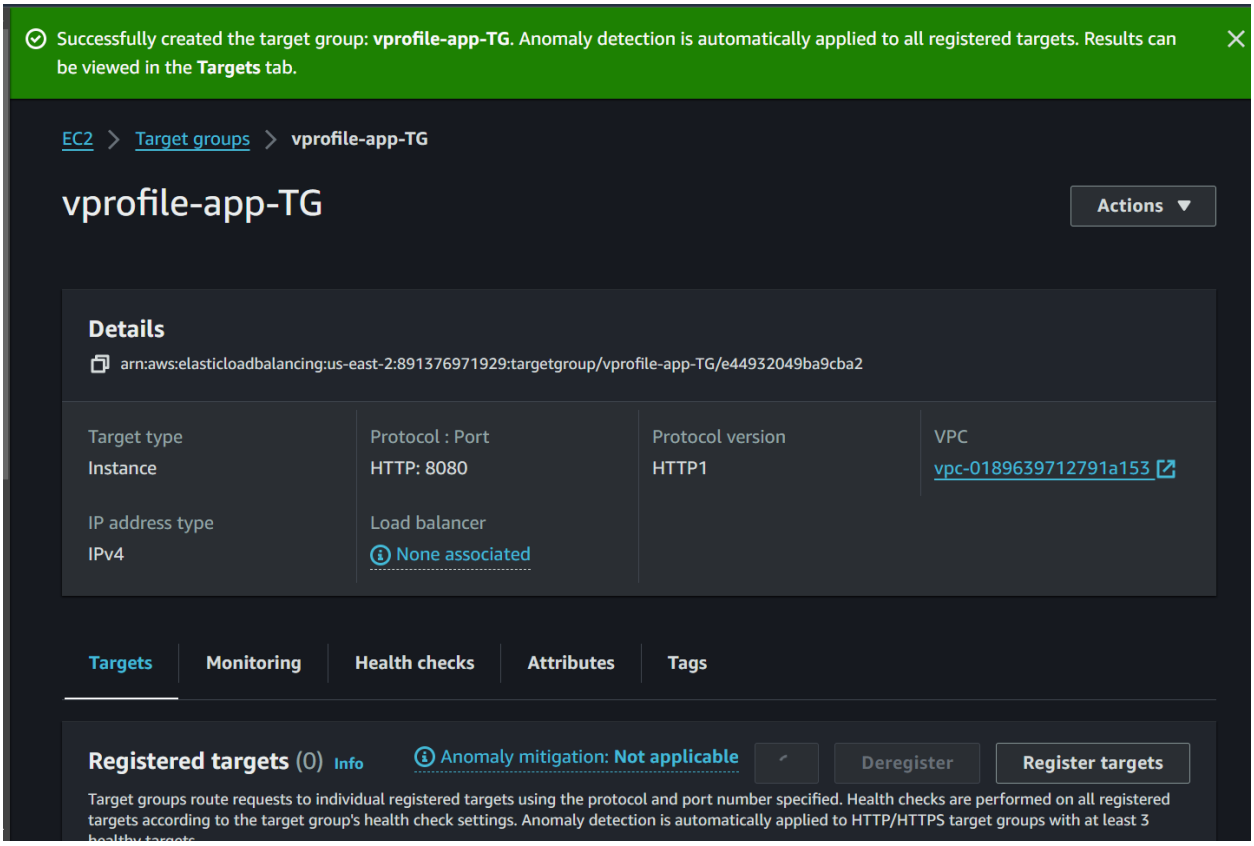
```
root@ip-172-31-3-224:~# aws s3 ls
2024-06-11 15:16:19 aditi-artifact-bucket-here
2024-05-27 07:39:29 elasticbeanstalk-us-west-2-891376971929
root@ip-172-31-3-224:~#
```

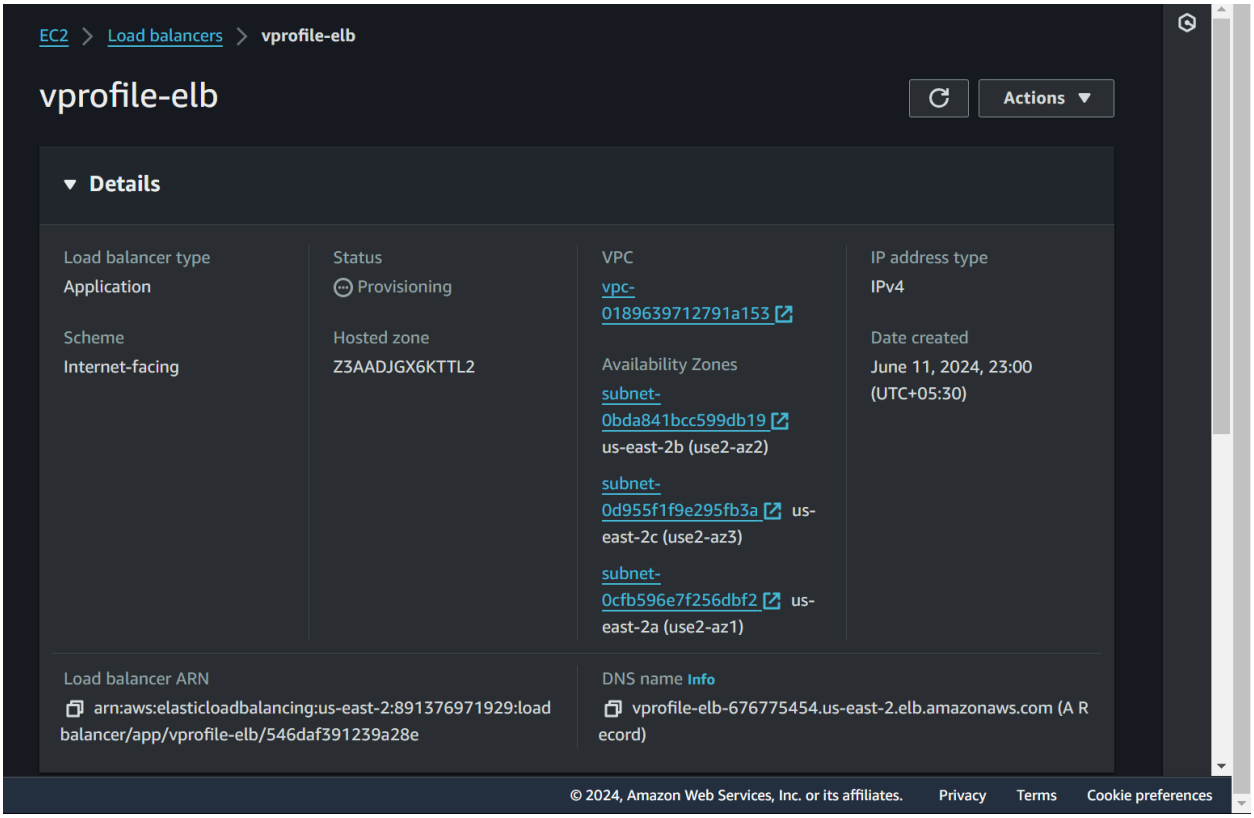
```
oot@ip-172-31-3-224:~# aws s3 cp s3://aditi-artifact-bucket-here/vprofile-v2.wa
 /tmp/
```

```
root@ip-172-31-3-224:~# rm -rf /var/lib/tomcat10/webapps/ROOT
root@ip-172-31-3-224:~# cp /tmp/vprofile-v2.war /var/lib/tomcat10/webapps/ROOT.w
ar
```

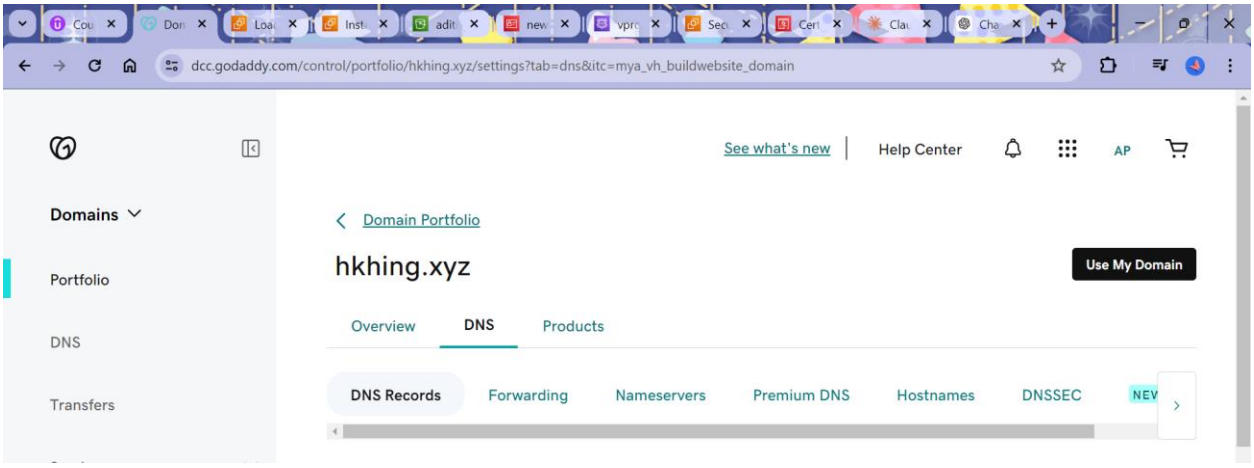
```
e changed on disk. Run 'systemctl daemon-reload' to reload units.
root@ip-172-31-3-224:~# sudo systemctl daemon-reload
root@ip-172-31-3-224:~# systemctl start tomcat10
root@ip-172-31-3-224:~# ls /var/lib/tomcat10/webappps/
ls: cannot access '/var/lib/tomcat10/webappps/': No such file or directory
root@ip-172-31-3-224:~# ls /var/lib/tomcat10/webapps/
ROOT  ROOT.war
root@ip-172-31-3-224:~#
```

19) Enable Load Balancers





20) Go to your service provider by clicking on DNS name



	NS	@	ns66.domaincontrol.com.	1 Hour	Can't delete	Can't edit
	CNAME	vprofileapp	vprofile-elb-676775454.us-east-2.elb.amazonaws.com.	1 Hour		
	CNAME	www	hkhing.xyz.	1 Hour		

Check your domain name in browser…