PROJECT

Project title: Three tier architecture {webserver,appication,database}

Date : 17/09/205

Name of the person : Gadi Thirupathi

Introduction: A Three-Tier Architecture is a cloud application design model 1.web Tier

2.Application Tier

3.Database Tier

that separates the syste into three independent layers.

- Web Tier
- Application Tier
- Database Tier

steps:

==> select a region

==> create a VPC with i.p:10.1.0.0/16 in th IPV4 version with the name tag <my-vpc>.

==> create the subnets in the created VPC such as,my-subnet01,   my-subnet02,my-subnet03.

==> create internet gateway and attach the VPC which we have created.

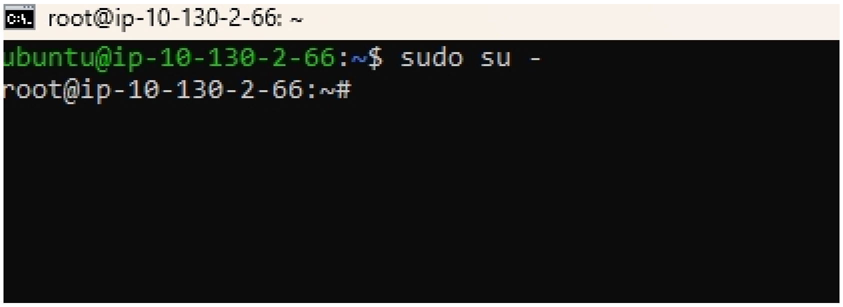
==> copy the VPC id and go to the route table paste the VPC id and edit the routes allow all traffic and click on the targets go through the internet gateway and the save the route.

==> copy the VPC id and go to the security groups paste the VPC id and go to the inbound rules and edit rules delete the existing inbound rule and allow the traffic via ssh port numer 22.

==> Here,now go to the Ec2 instances and click on launch instance and here create the three instances with the name tags such as , nginx01,tomcat01,mysql01 for all these machines select the ubuntu server and select the key and select the VPC which we created in the above select the exiting subnets and security gruops and launch three instances.

==> Now connect to the nginx server in the cammand promt

1.Web Tier



root@ip-10-130-2-66: ~

```
ubuntu@ip-10-130-2-66:~$ sudo su -
root@ip-10-130-2-66:~#
```

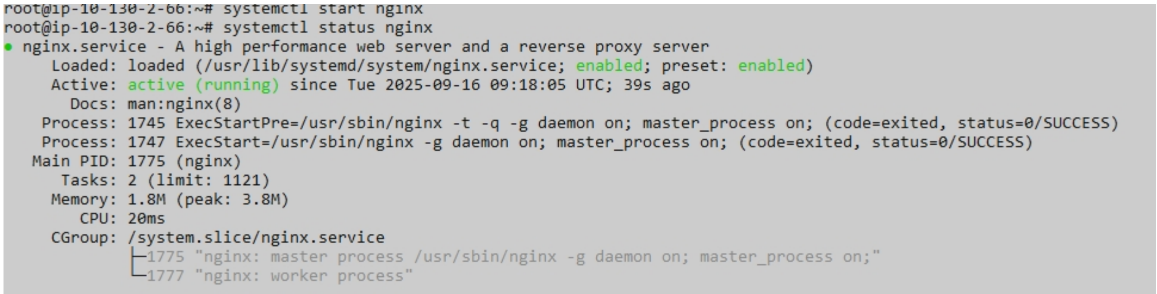
==> To install nginx in the machine enter those cammands such as,

#apt update

#apt install nginx -y

#systemctl start nginx

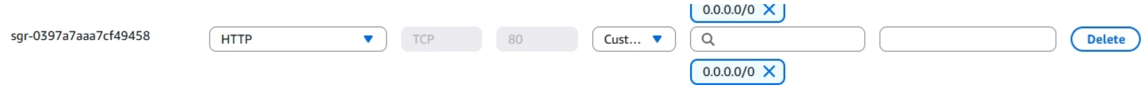
#systemctl status nginx



```
root@ip-10-130-2-66:~# systemctl start nginx
root@ip-10-130-2-66:~# systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-09-16 09:18:05 UTC; 39s ago
       Docs: man:nginx(8)
    Process: 1745 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 1747 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 1775 (nginx)
      Tasks: 2 (limit: 1121)
     Memory: 1.8M (peak: 3.8M)
        CPU: 20ms
     CGroup: /system.slice/nginx.service
             ├─1775 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─1777 "nginx: worker process"
```

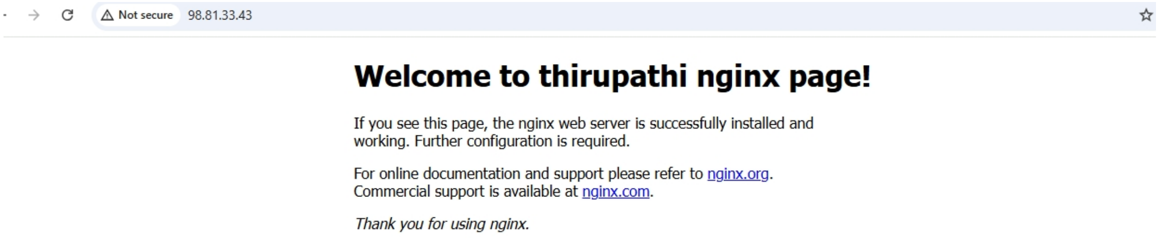
==> here in the above started the nginx and checked its status and it is in the running status.

==> copy the public ip and browse we can see that welcome to nginx.Before doing this we have to the edit the inbound rules in the security groups allow the traffic HTTP port 80.



sgr-0397a7aaa7cf49458	HTTP ▼	TCP	80	Cust... ▼	0.0.0.0/0 ✕ 🔍 0.0.0.0/0 ✕		Delete

==> Here now we can see thr browse page



⚠ Not secure 98.81.33.43

Welcome to thirupathi nginx page!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

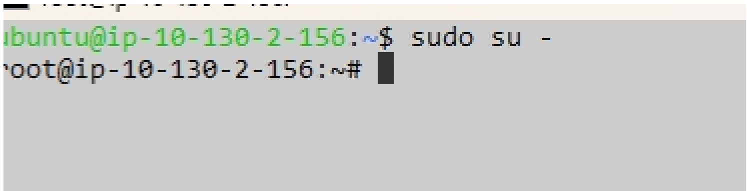
==>In the above completed the web Tier

===

2.Application Tier

steps:

==> connect to the application machine which we created in the above.



```
buntu@ip-10-130-2-156:~$ sudo su -
oot@ip-10-130-2-156:~#
```

==> In the above machine install the tomcat application from the chrome by using the following commands.

#apt update

#apt install wget -y

#wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.109/bin/apache-tomcat-9.0.109.tar.gz

#tar -xzvf (extract file from tar achive)

#apt install openjdk-17-jre-headless

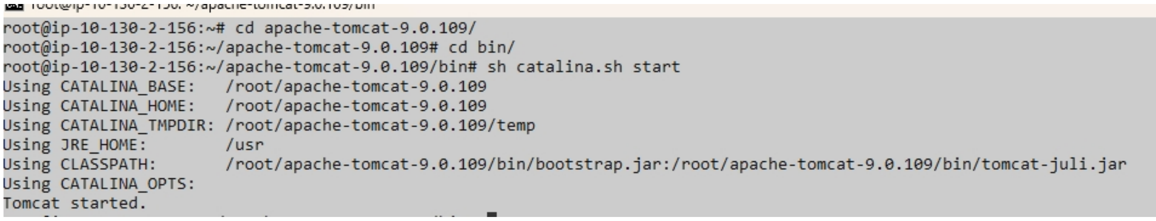
#cd apachi-tomcat

#cd bin

#sh catalina.sh start

#sh catalina.sh status

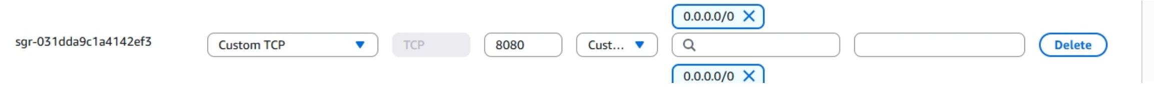
==> By all entering the above cammands the tomcat apllication is started.



```
root@ip-10-130-2-156:~# cd apache-tomcat-9.0.109/
root@ip-10-130-2-156:~/apache-tomcat-9.0.109# cd bin/
root@ip-10-130-2-156:~/apache-tomcat-9.0.109/bin# sh catalina.sh start
Using CATALINA_BASE:   /root/apache-tomcat-9.0.109
Using CATALINA_HOME:   /root/apache-tomcat-9.0.109
Using CATALINA_TMPDIR: /root/apache-tomcat-9.0.109/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /root/apache-tomcat-9.0.109/bin/bootstrap.jar:/root/apache-tomcat-9.0.109/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
```

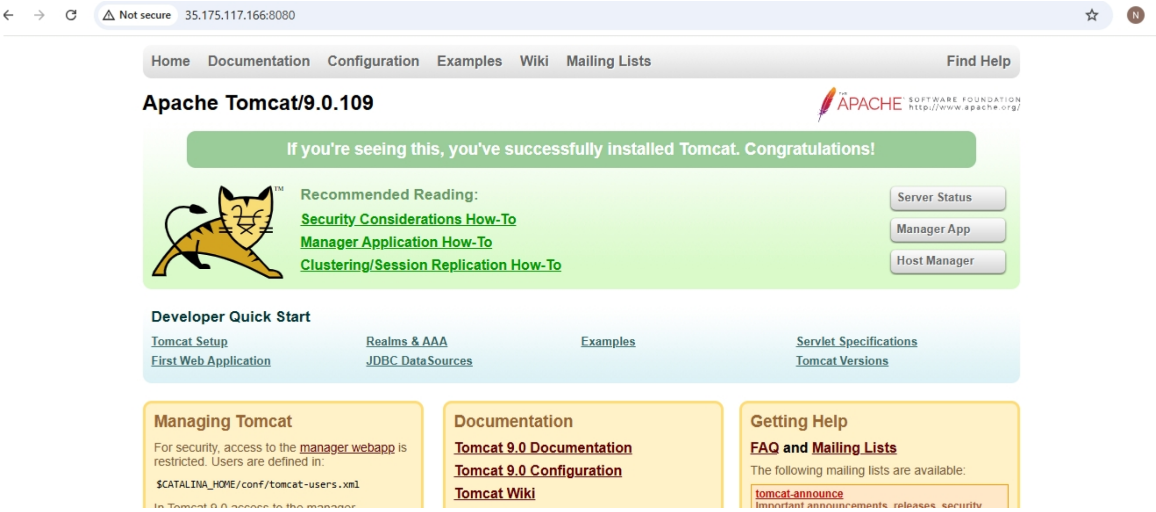
==> In the above tomcat aplication is started.

==>To browse the tomcat apllication ,go to the security groups click oninbound rules and edit the inbound rules select the custom tcp and give the port range 8080 and save the rule we can browse the tomat application with the public.ip.



sgr-031dda9c1a4142ef3 | Custom TCP | TCP | 8080 | Cust... | 0.0.0.0/0 | 0.0.0.0/0

==> And here we can able see in the tomcat application is browsing.



Not secure 35.175.117.166:8080

Apache Tomcat/9.0.109

If you're seeing this, you've successfully installed Tomcat. Congratulations!

Recommended Reading:
Security Considerations How-To
Manager Application How-To
Clustering/Session Replication How-To

Developer Quick Start
Tomcat Setup
First Web Application
Realms & AAA
JDBC DataSources
Examples
Servlet Specifications
Tomcat Versions

Managing Tomcat
For security, access to the manager webapp is restricted. Users are defined in:
$CATALINA_HOME/conf/tomcat-users.xml

Documentation
Tomcat 9.0 Documentation
Tomcat 9.0 Configuration
Tomcat Wiki

Getting Help
FAQ and Mailing Lists
The following mailing lists are available:
tomcat-announce

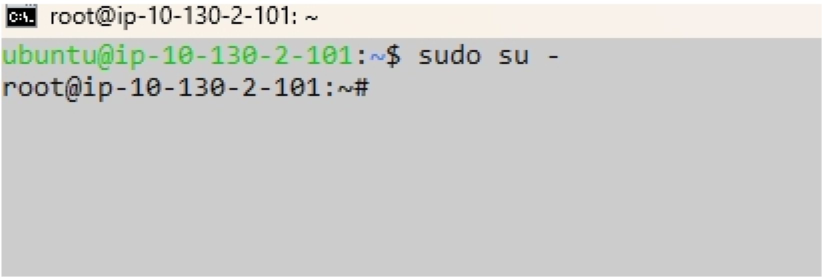
==> In the above completed the application tier

==

3.Database Tier

steps:

==> connect to the database machine which we created in the above.



```
root@ip-10-130-2-101: ~
ubuntu@ip-10-130-2-101:~$ sudo su -
root@ip-10-130-2-101:~#
```

==> In the above machine install database by using the following cammands.

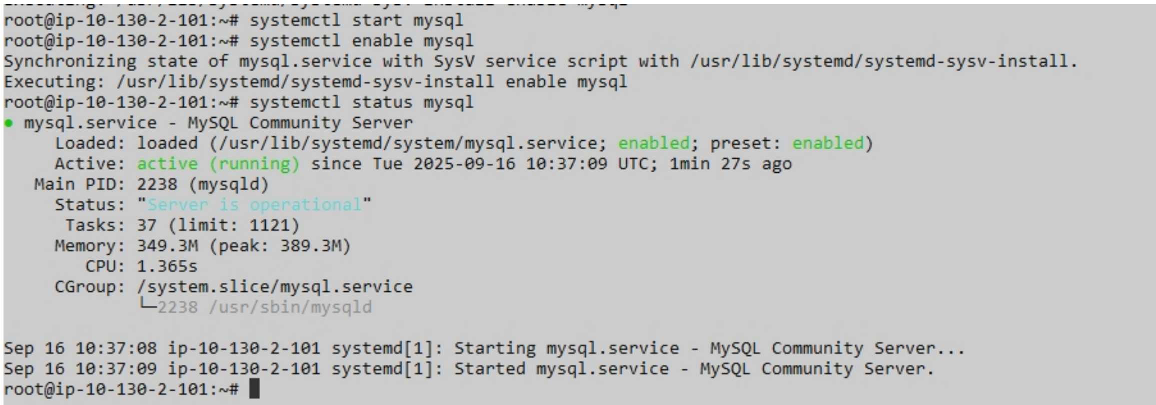
#apt update

#apt install mysql-server -y

#systemctl enable mysql

#systemctl start mysql

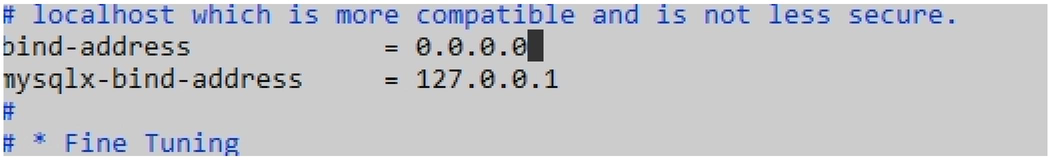
#system status mysql



```
root@ip-10-130-2-101:~# systemctl start mysql
root@ip-10-130-2-101:~# systemctl enable mysql
Synchronizing state of mysql.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable mysql
root@ip-10-130-2-101:~# systemctl status mysql
● mysql.service - MySQL Community Server
     Loaded: loaded (/usr/lib/systemd/system/mysql.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-09-16 10:37:09 UTC; 1min 27s ago
   Main PID: 2238 (mysqld)
     Status: "Server is operational"
      Tasks: 37 (limit: 1121)
     Memory: 349.3M (peak: 389.3M)
        CPU: 1.365s
     CGroup: /system.slice/mysql.service
             └─2238 /usr/sbin/mysqld

Sep 16 10:37:08 ip-10-130-2-101 systemd[1]: Starting mysql.service - MySQL Community Server...
Sep 16 10:37:09 ip-10-130-2-101 systemd[1]: Started mysql.service - MySQL Community Server.
root@ip-10-130-2-101:~# █
```

#vi /etc/mysql/mysq.conf.d/mysqld.conf

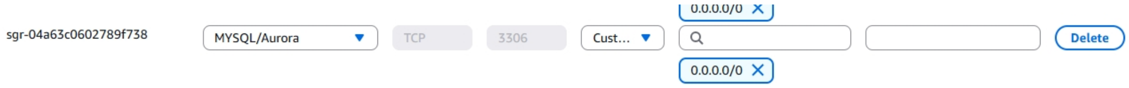


```
# localhost which is more compatible and is not less secure.
bind-address            = 0.0.0.0
mysqlx-bind-address     = 127.0.0.1
#
# * Fine Tuning
```

change ==>bind-address = 0.0.0.0 and save the file quit

#systemctl restart mysql

==> To connect the application tier go to security groups and click on inbound edit the inbound rules allow traffic mysql/aurora with the port range 3306.



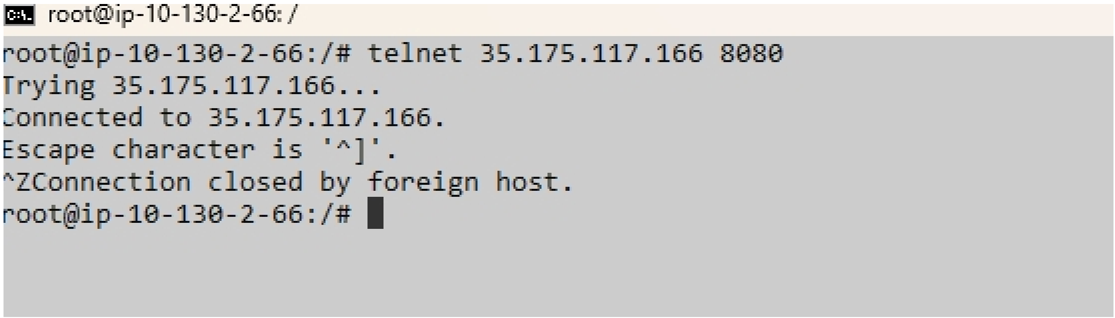
sgr-04a63c0602789f738 MYSQL/Aurora TCP 3306 Cust... 0.0.0.0/0 0.0.0.0/0

==>Now it connects to application tier.

==

==>Connect the nginx server to tomcat application.

use the command:#telnet appip 8080



root@ip-10-130-2-66: /

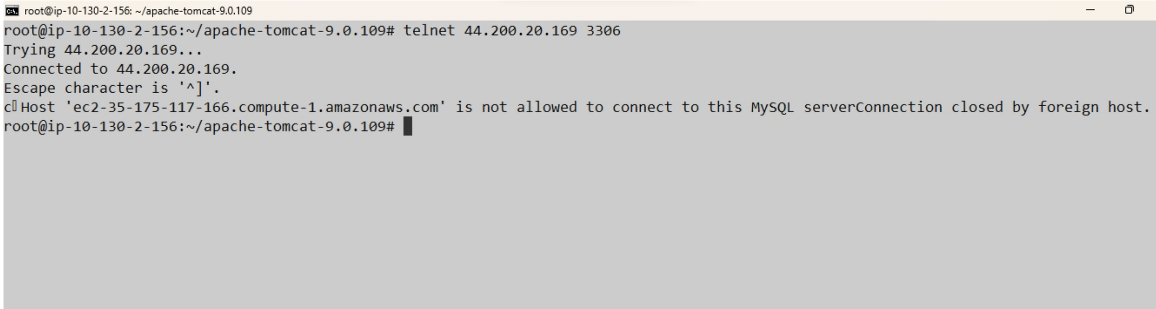
```
root@ip-10-130-2-66:/# telnet 35.175.117.166 8080
Trying 35.175.117.166...
Connected to 35.175.117.166.
Escape character is '^]'.
^ZConnection closed by foreign host.
root@ip-10-130-2-66:/#
```

nginx server to tomcat application is succesfully connected.

==

==>Connect the tomcat apllication to database.

use the cammand:#telnet dbip 3306



```
root@ip-10-130-2-156:~/apache-tomcat-9.0.109# telnet 44.200.20.169 3306
Trying 44.200.20.169...
Connected to 44.200.20.169.
Escape character is '^]'.
c Host 'ec2-35-175-117-166.compute-1.amazonaws.com' is not allowed to connect to this MySQL serverConnection closed by foreign host.
root@ip-10-130-2-156:~/apache-tomcat-9.0.109# 
```

tomcat application is succesfully connected to database.

Conclusion: The implementation of a three-tier architeturein AWS using web,Application and database layers demonstrates the importance of modular and secure cloud infrastructure.By isolating each tier within its own security group.