

Documentation of 3-tier architecture

Introduction:-

A modern and robust system designed to optimize performance, enhance scalability, and ensure seamless data management. In this endeavor, we integrate three fundamental components: the application server, the web server, and the database server, each playing a pivotal role in achieving a powerful and efficient application architecture.

At the forefront, the web server stands as the interface between users and the application, handling HTTP requests and responses.

One application server in which “studentapp” application is present.

And one database server which will store that student information.

Steps involved in 3-tier architecture are:-

1.Create VPC:-

create vpc with the name of 3-tier architecture in mumbai region

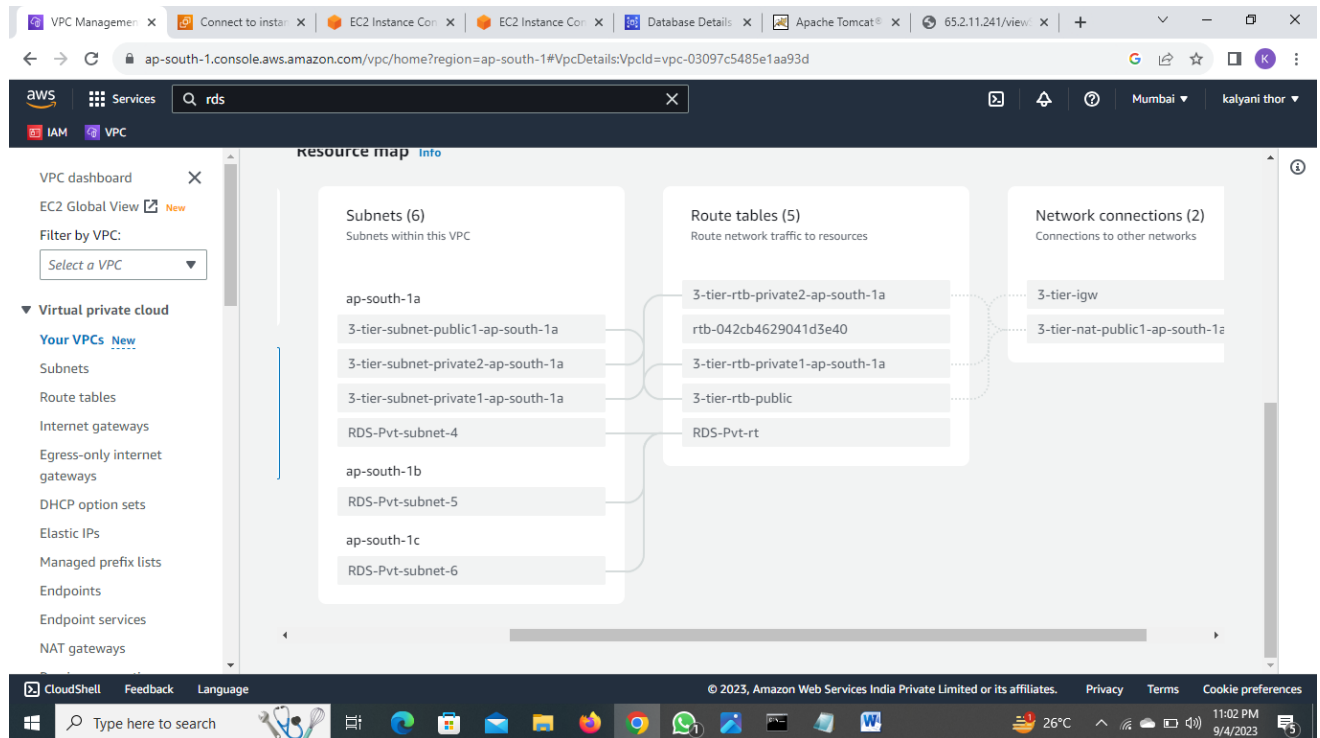
2.Create subnets:-

1 public subnet

2 private subnet

Create NAT-gateway

Rout-table of VPC



3. Create 2 instance:

1. app server:- (private server i.e in private subnet)
2. jump server:- (public server i.e in public subnet)

4. Create database:-Create database in RDS service that will be store data. I have used Mariadb database with username “admin” and password “admin123”.

5. now ssh jump server from mobaxtream

And ssh app server from jump server

6. Copy all the required files from jump server to app server

7. Now do the configurations in app server as follows:-



The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, 'Services' link, a search bar, and an '[Alt+S]' button. Below the navigation bar, there are tabs for 'IAM' and 'VPC'. The main content area displays a terminal window with the following commands:

```
1 sudo -i
2 ls
3 sudo yum install java-1.8.0-amazon-corretto.x86_64
4 curl -O https://d1cdn.apache.org/tomcat/tomcat-8/v8.5.93/bin/apache-tomcat-8.5.93.zip
5 unzip apache-tomcat-8.5.93.zip
6 ls
7 cd apache-tomcat-8.5.93/
8 ls
9 mv /home/ec2-user/student.war /home/ec2-user/apache-tomcat-8.5.93/webapps/
10 cd webapps
11 ls
12 cd ..
13 cd
14 ls
15 mv /home/ec2-user/mysql-connector.jar apache-tomcat-8.5.93/lib
16 cd apache-tomcat-8.5.93/
17 chmod 777 bin/catalina.sh
18 chmod 777 bin/catalina.sh
19 cd conf
20 ls
21 vim context.xml
22 cd ..
23 cd bin
24 ls
25 ./catalina.sh start
```

Below the terminal window, the instance ID 'i-05ed6c932f5dccc62 (app server)' is displayed, along with its Public IP '15.207.111.162' and Private IP '172.31.3.180'.

Add changes in apache-tomcat-8.5.93/conf file:-

vim context.xml(line no 21)

`<Resource name="jdbc/TestDB" auth="Container" type="javax.sql.DataSource"`

`maxTotal="100" maxIdle="30" maxWaitMillis="10000"`

`username="USERNAME" password="PASSWORD" driverClassName="com.mysql.jdbc.Driver"`

`url="jdbc:mysql://DNS of database/DATABASE"/>`

Here add all required fields such as username, password, DNS of database, and database name as shown bellow:-

Microsoft Teams x Connect to instance x EC2 Instance Connect x Database Details - RD x Apache Tomcat® - Ap x 43.204.116.167:8080/ x +

ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-Od4746399f82068b1&osUser=ec2-user&sshPor...

aws Services Search [Alt+S]

IAM VPC

The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-->

<!-- The contents of this file will be loaded for each web application -->

<Context>

<Resource name="jdbc/TestDB" auth="Container" type="javax.sql.DataSource" maxTotal="100" maxIdle="30" maxWaitMillis="10000" username="admin" password="admin123" driverClassName="com.mysql.jdbc.Driver" url="jdbc:mysql://Database-1.ckzbu5hcuqln.ap-south-1.rds.amazonaws.com/studentapp"/>

<!-- Default set of monitored resources. If one of these changes, the -->

-- INSERT --

24, 34 33%

i-Od4746399f82068b1 (app server1)

PublicIPs: 43.204.116.167 PrivateIPs: 172.31.0.43

CloudShell Feedback Language

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Type here to search

26°C 11:50 AM 8/29/2023

The screenshot shows the AWS CloudShell interface with a terminal window. The terminal displays a series of commands for setting up a web application and database on an EC2 instance. The commands include installing MariaDB, starting the service, creating a database, and setting up a web application. The terminal output shows the successful execution of these commands. Below the terminal window, the instance ID is listed as i-05ed6c932f5dccf62 (app server). The public IP address is 15.207.111.162 and the private IP address is 172.31.3.180. The bottom of the screenshot shows the Windows taskbar with various application icons and the system clock.

```
24 ls
25 ./catalina.sh start
26 curl localhost:8080
27 cd ..
28 cd webapps
29 sudo yum install mariadb105-server -y
30 sudo systemctl start mariadb
31 sudo systemctl enable mariadb
32 mysql -h database-1.ckzbu5hcug1n.ap-south-1.rds.amazonaws.com -u admin -padmin123
33 cd conf
34 cd ..
35 cd conf
36 ls
37 vim context.xml
38 cd ..
39 cd webapps
40 mysql -h database-1.ckzbu5hcug1n.ap-south-1.rds.amazonaws.com -u admin -padmin123
41 history
42 ssh -i "key_for_aws.pem" ec2-user@ec2-65-1-93-106.ap-south-1.compute.amazonaws.com
43 chmod 400 key_for_aws.pem
44 chmod 400 key_for_aws
45 ls
46 ssh -i "key_for_aws.pem" ec2-user@ec2-65-1-93-106.ap-south-1.compute.amazonaws.com
47 chmod 400 key_for_aws
48 ssh -i "key_for_aws.pem" ec2-user@ec2-65-1-93-106.ap-south-1.compute.amazonaws.com
```

i-05ed6c932f5dccf62 (app server)

PublicIPs: 15.207.111.162 PrivateIPs: 172.31.3.180

7. in mariadb hit commands:-

Create database studentapp;

Show database;

```
Use studentappCREATE TABLE if not exists students(student_id INT NOT
NULL AUTO_INCREMENT,
    student_name VARCHAR(100) NOT NULL,
student_addr VARCHAR(100) NOT NULL,
    student_age VARCHAR(3) NOT NULL,
    student_qual VARCHAR(20) NOT NULL,
    student_percent VARCHAR(10) NOT NULL,
    student_year_passed VARCHAR(10) NOT NULL,
    PRIMARY KEY (student_id)
);
```

Describe students;

```
aws Services Search [Alt+S]
IAM VPC
-> student_qual VARCHAR(20) NOT NULL,
-> student_percent VARCHAR(10) NOT NULL,
-> student_year_passed VARCHAR(10) NOT NULL,
-> PRIMARY KEY (student_id)
-> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [studentapp]> dscribe students;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for t
near 'dscribe students' at line 1
MariaDB [studentapp]> describe students;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| student_id | int(11) | NO | PRI | NULL | auto_increment |
| student_name | varchar(100) | NO | | NULL | |
| student_addr | varchar(100) | NO | | NULL | |
| student_age | varchar(3) | NO | | NULL | |
| student_qual | varchar(20) | NO | | NULL | |
| student_percent | varchar(10) | NO | | NULL | |
| student_year_passed | varchar(10) | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.001 sec)

MariaDB [studentapp]> exit

i-0fbb38aab2f6c5dfc (jump server)
PublicIPs: 65.2.11.241 PrivateIPs: 10.0.8.81
```

8. Now in jump server install nginx it will be work as proxy server.

and make changes in /etc/nginx/nginx.conf file.

Add ip address of your application server and database name

{Proxy_pass://https:ip_address:8080/student/;}

```
server {
    listen 80;
    listen [::]:80;
    server_name _;
    root /usr/share/nginx/html;

    # Load configuration files for the default server block.
    include /etc/nginx/default.d/*.conf;

    error_page 404 /404.html;
    location / {
        proxy_pass http://43.204.233.88:8080/student/;
    }

    location = /404.html {
    }

    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
    }
}

# Settings for a TLS enabled server.
#
"nginx.conf" 87L, 2378B
48,32-39
```

10.now hit the ip of jump server we will get the page of studentapp:-

Student Registration Form

Student Name

Student Address

Student Age

Student Qualification

Student Percentage

Year Passed



11. do some entries in application form

[Register Student](#)

Students List

Student ID	StudentName	Student Addr	Student Age	Student Qualification	Student Percentage	Student Year Passed	Edit	Delete
1	kalyani	nagpur	28	B E	80	2020	edit	delete
2	suraj	mumbai	30	B E	79	2022	edit	delete



12. now check on a app server entries are present or not.

The screenshot shows an AWS CloudShell terminal window with the following content:

```
student_id | int(11) | NO | PRI | NULL | auto_increment
student_name | varchar(100) | NO | NULL
student_addr | varchar(100) | NO | NULL
student_age | varchar(3) | NO | NULL
student_qual | varchar(20) | NO | NULL
student_percent | varchar(10) | NO | NULL
student_year_passed | varchar(10) | NO | NULL
7 rows in set (0.001 sec)
```

MariaDB [studentapp]> select * from student;
ERROR 1146 (42S02): Table 'studentapp.student' doesn't exist
MariaDB [studentapp]> select * from studentapp;
ERROR 1146 (42S02): Table 'studentapp.studentapp' doesn't exist
MariaDB [studentapp]> select * from students;

student_id	student_name	student_addr	student_age	student_qual	student_percent	student_year_passed
1	kalyani	nagpur	28	B.E	80	2020
2	suraj	mumbai	30	B.E	79	2022

2 rows in set (0.001 sec)

MariaDB [studentapp]>

i-0fbb38aab2f6c5dfc (jump server)
PublicIPs: 65.2.11.241 PrivateIPs: 10.0.8.81

CloudShell Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Yow will get the entries as we mentioned in the application form.

By this way we end up with the three tier architecture were web app i.e. jump server, and application app i.e. app server and third database are integrated with each other.