

INTELLIPAAT AWS CAPSTONE PROJECT 2

- Iyappan A

PROBLEM STATEMENT:

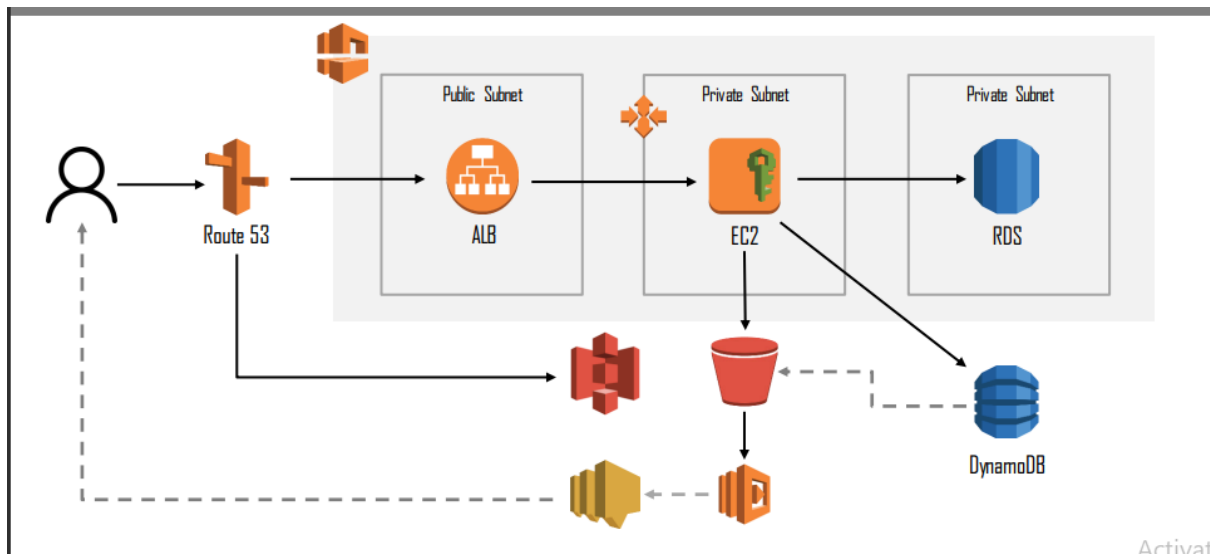
1. Use a python web application to pull employee form data
2. Data should include basic ID information with image
3. Store it in secure database using AWS services
4. Retrieve the data and check

Employee profile of XYZ company – New employees input their information and upload photos. Existing employees can get their information.

RESOURCE REQUIREMENTS:

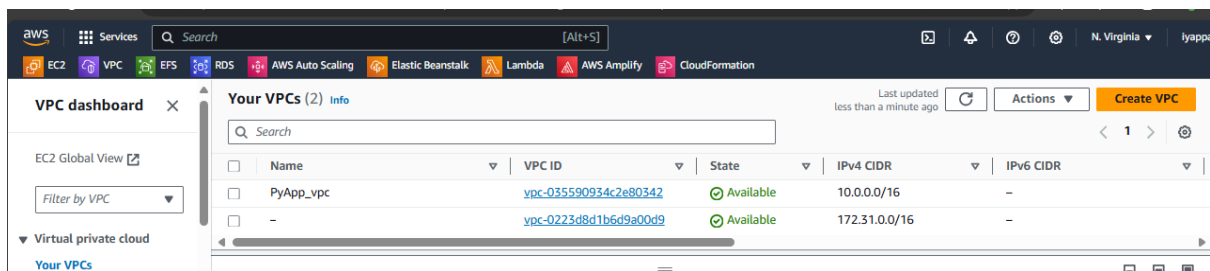
1. VPC network and other supplementary components
 - a. Subnets
 - b. Internet gateway
 - c. Natgateway
 - d. Routetables
2. S3 bucket – to store image files from form
3. DynamoDB Table – to maintain table of employee ID and image URL
4. RDS – to maintain SQL database
5. Ec2 machine – to host python web application
6. Loadbalancer – to route the traffic

TECHNICAL ARCHITECTURE:

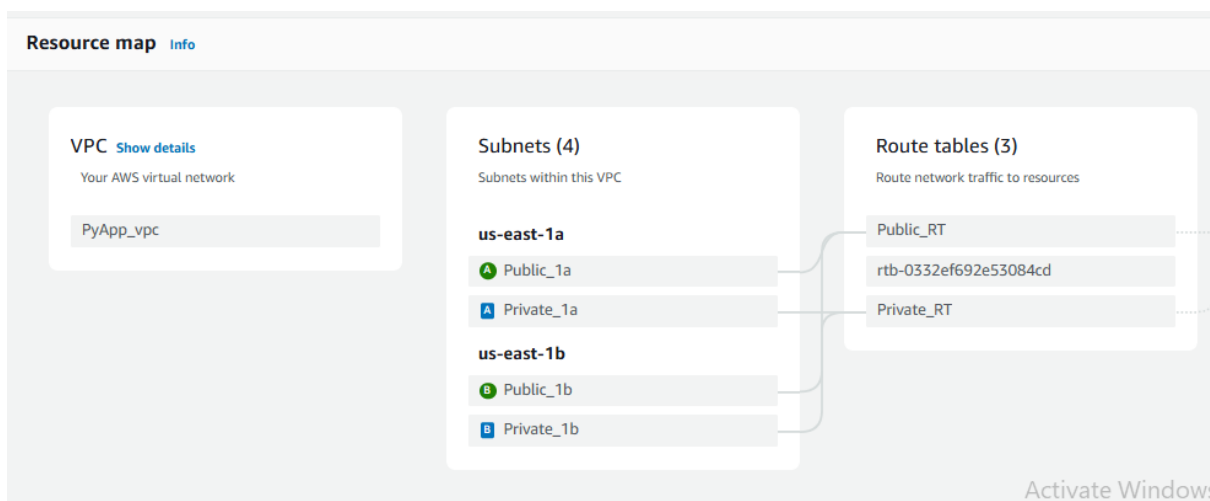


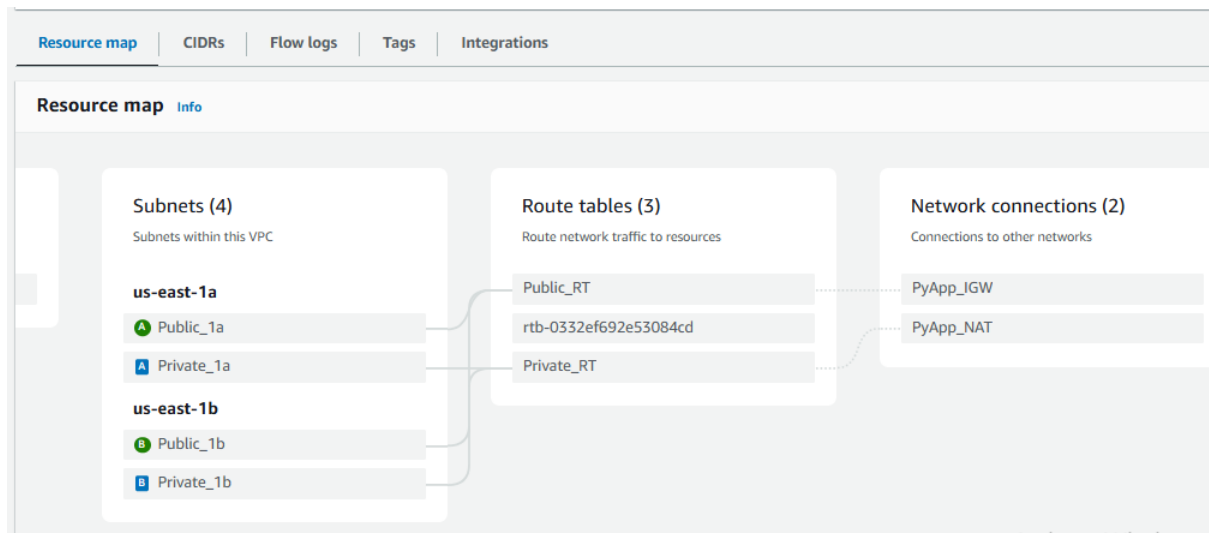
SOLUTION:

1. Create a VPC with 4 subnets (2 public and 2 private)

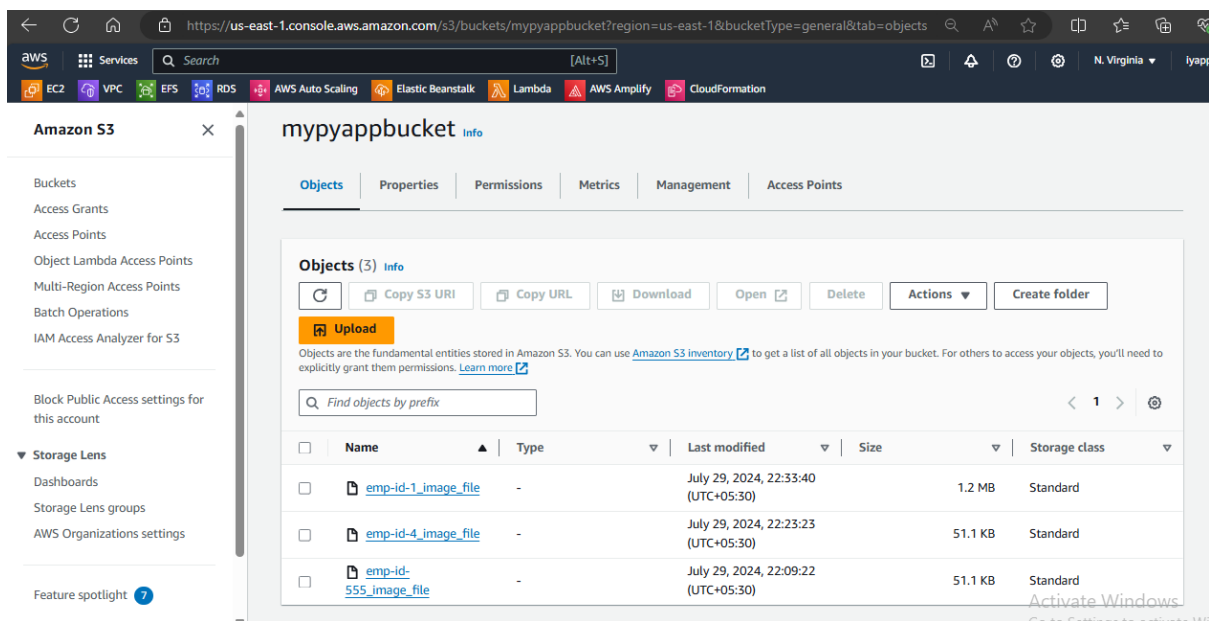


2. Create route table separate for both public and private.

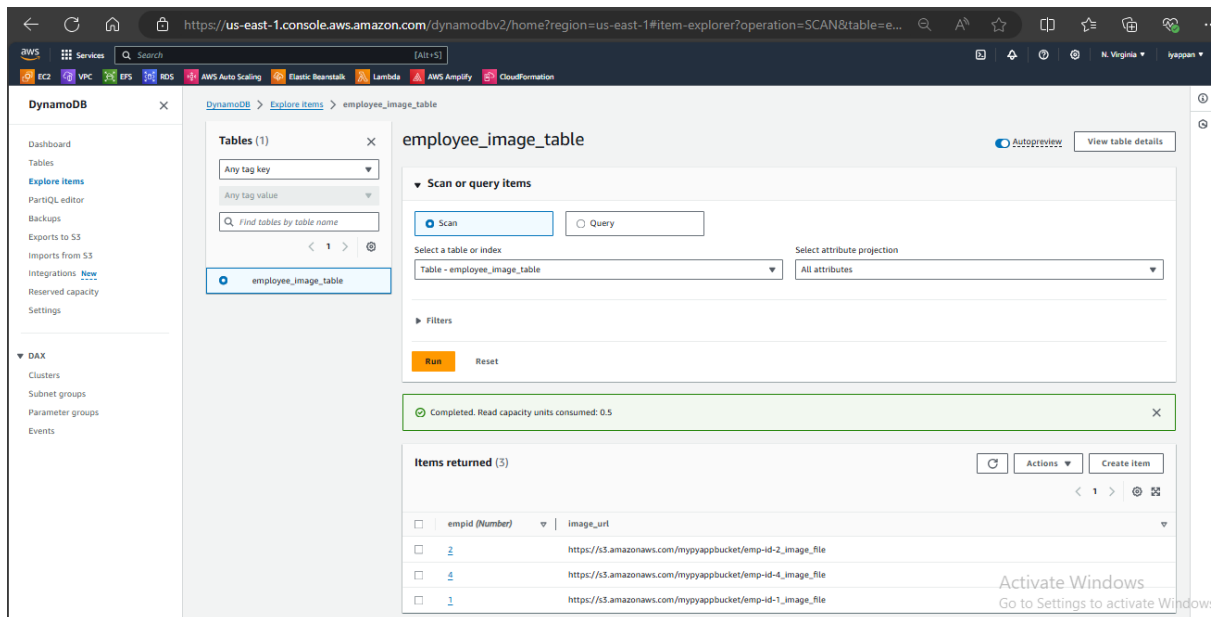




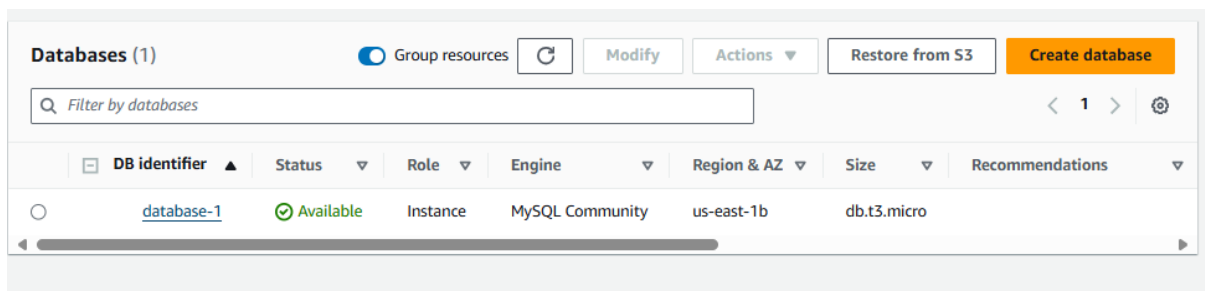
3. Attach internet gateway to public subnets
4. Create NAT in public subnet and attach NAT gateway to private subnets
5. Create a s3 bucket



6. Create DyanmoDB table with Employee DB and schema to store Image url and corresponding emp ID.



7. Create a RDS database with Employee DB and schema to store data from the form.



8. Create an IAM role with S3 access and DynamoDB access policy

The screenshot shows the AWS IAM console page for the role 'iam_role_ec2'. The role is described as 'Allows EC2 instances to call AWS services on your behalf.' The 'Summary' section shows the creation date as July 29, 2024, 22:06 (UTC+05:30), the ARN as 'arn:aws:iam::260083387156:role/iam_role_ec2', and the instance profile ARN as 'arn:aws:iam::260083387156:instance-profile/iam_role_ec2'. The last activity was 35 minutes ago. The 'Permissions' section shows two attached policies: 'AmazonDynamoDBFullAccess' and 'AmazonS3FullAccess', both of which are AWS managed.

Policy name	Type	Attached entities
AmazonDynamoDBFullAccess	AWS managed	3
AmazonS3FullAccess	AWS managed	3

9. Create EC2 instance in private subnet (Py_App)

10. Create EC2 instance in public subnet to access private instance to update and do further configuration (Config_Ec2)

The screenshot shows the AWS EC2 console 'Instances' page. It lists two instances: 'public_test' and 'Py_App_private'. Both instances are in the 'Running' state and are using 't2.micro' instance types. The 'public_test' instance has an ID of 'i-06fd1e63c77bac40a' and the 'Py_App_private' instance has an ID of 'i-07d66bdafd7af1b87'. Both instances have 2/2 status checks passing.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
public_test	i-06fd1e63c77bac40a	Running	t2.micro	2/2 checks passing	View alarms
Py_App_private	i-07d66bdafd7af1b87	Running	t2.micro	2/2 checks passing	View alarms

11. Create a target group include private instance association

12. Attach target group to load balancer, configure loadbalancer in public subnets availability zones.

13. Attach IAM role to Py_App Ec2 instance (private instance)

14. Login into Config_Ec2 instance (public instance)

15. Copy key_Pair into instance and try to SSH from public instance to private instance.

16. Install update and other dependencies in the private instance (Py_App instance)

17. Launch the app ensure the app configurations are pointed at databases correctly

```
EC2 VPC EFS RDS AWS Auto Scaling Elastic Beanstalk Lambda AWS A
customhost = "database-1.cfb6u4qbf9na.us-east-1.rds.amazonaws.com"
customuser = "admin"
custompass = "admin123"
customdb = "employee"
custombucket = "mypyappbucket"
customregion = "us-east-1"
~
~
~
~
~
~
```

18. Ping from browser and fill the employee form

← ↻ 🔒 Not secure | py-app-lb-1425201520.us-east-1.elb.amazonaws.com

Employee Database

GET EMPLOYEE INFORMATION

Employee ID:

First Name:

Last Name:

Primary Skills:

Location:

Image: my pic11.jpg

UPDATE DATABASE

[About Us](#)

Activate Windows
Go to Settings to activate Windows.

19. Check RDS, DynamoDB, and s3 bucket to ensure data storing.

```
mysql> use employee;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from employee;
+-----+-----+-----+-----+-----+
| emp_id | first_name | last_name | primary_skills | location |
+-----+-----+-----+-----+-----+
| 1      | iyappan   | A        | Devops        | chennai  |
| 1      | iyappan   | A        | Devops        | chennai  |
| 2      | valli     | asaithambi | cooking       | cuddalore |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

Items returned (3)

empid (Number)	image_url
2	https://s3.amazonaws.com/mypyappbucket/emp-id-2_image_file
4	https://s3.amazonaws.com/mypyappbucket/emp-id-4_image_file
1	https://s3.amazonaws.com/mypyappbucket/emp-id-1_image_file

mypyappbucket

Objects (3)

Name	Type	Last modified	Size	Storage class
emp-id-1_image_file	-	July 29, 2024, 22:33:40 (UTC+05:30)	1.2 MB	Standard
emp-id-4_image_file	-	July 29, 2024, 22:23:23 (UTC+05:30)	51.1 KB	Standard
emp-id-555_image_file	-	July 29, 2024, 22:09:22 (UTC+05:30)	51.1 KB	Standard

Hence the projected is completed and tested successfully.