**ASSIGNMENT: CLOUD AWS QUICKLAB**

**Building VPC, S3, EC2, and RDS Products with AWS Service Catalog**

In this report I will describe an step by step of the quicklab.

**Task 1. Create your Portfolio:** First I logged in aws with the administrator account with the credentials the lab gave.

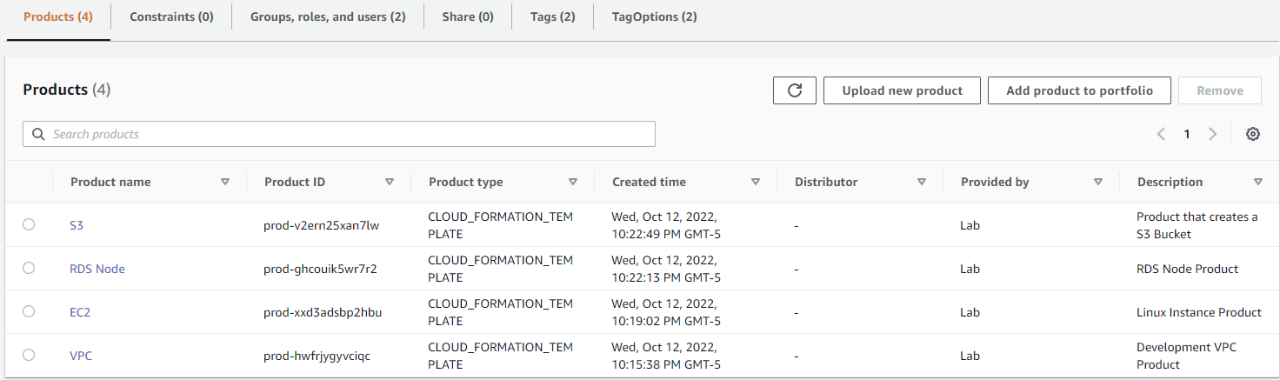
Portafolio were created and users “user-devoler” and “user-admin” were added to it. Portfolio tags “Portfolio” and “Provider” were created as well as the tags associated to products being launched “Operations” and “Development”.

Una captura de pantalla de una red social

Descripción generada automáticamente

Note the 2 users, the 2 Tags and the 2 TagOptions.

**Task 2. Create your products:** Into theportfolioa VPC, an EC2 instance with Linux, a RDS database instance with MySQL and a S3 bucket were created as products, all of them using the given templates files.



**Task 3. Create Launch Constraints for your Products:** Here, an IAM role is created for every of the 4 products, this allows them to launch the VPC, S3, EC2 and RDS by giving this IAM roles a Launch constraint.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

This way, “SC-S3-ROLE”, “SC-RDS-ROLE”, “SC-VPC-ROLE” and “SC-EC2-ROLE” roles were created in constraints tab.

Texto

Descripción generada automáticamente**Task 4-7. Launch VPC, S3, EC2 and RDS products:** First the development VPC is created with a public and private subnet with associated internet gateway. When launching the VPC product following the instructions, next error was stated:

In order to solved this I changed the provisioned product name for “myVPC\_scool”, which was not used in the platform.

Now, to launch the S3 bucket the previously product “S3” were used, with the “user-developer” ARN this user is allowed to access to the bucket.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamenteTo launch the Linux EC2 instance , the next parameters were stated. The SecurityGroup and the Subnet were found at the outputs on the VPC stack details. Same problem with the stack name as the one creating VPC happened, so I named it as my EC2\_school.

Finally, when launching the RDS Product, the DBSecurityGroup were also found at the outputs on the VPC stack details.

All the launchings were made from the user-developer account, not the admin account.

**Task 8: Add Template Constraints:** Constraints to restrict the selectable SecurityGroup and Subnet for the Linux EC2 Product and RDS Node Product by the developer user were created.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

This is verified in a next step, were another Linux EC2 instance is launched.

Interfaz de usuario gráfica, Texto, Aplicación

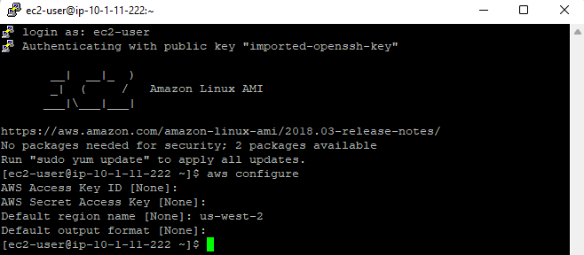
Descripción generada automáticamenteNote how in parameters selections, the SecurityGroup and the Subnet are restricted as it were stated before. It also happened whe launching another RDS Node Product.

This are the final provisioned products:

Una captura de pantalla de una computadora

Descripción generada automáticamente

**Maintaining High Availability with Auto Scaling**

****With Putty, I connect to the predefine EC2 instance via SSH. Then AWS CLI was configured with non-access and secret key on us-west-2 region.

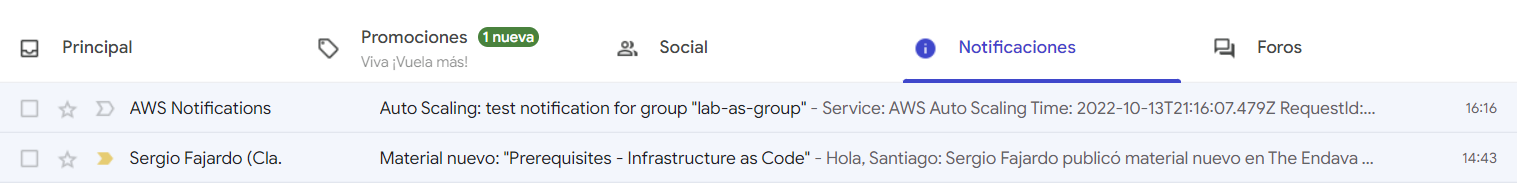
Then, I create a launch configuration to specifies de AMI that will be launch when auto scaling, the AMI is the same as the one in the predefined EC2 created by the lab, using the keyname, the security group and the AMI ID the lab provide. With this same logic an Auto Scaling Group is created with de load balancer, and subnets provided by the lab. Both commands with “aws autoscaling” API.

When auto creating a new instance after terminating or stopping the running one (because it detects that the fleet size is below the minimum size) it is automatically added to the load balancer.

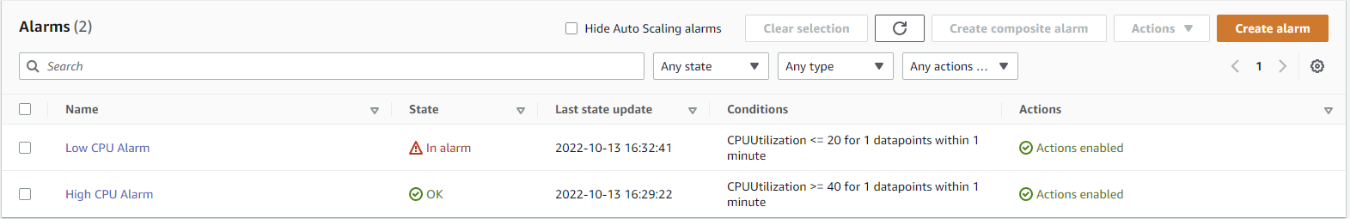
Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

After that auto scaling notifications are created using SNS, this will sed messages to my mail when an instance is launched or terminated.



Then a Scale-up and Scale-down policies are created to increase the number of servers whenever the average CPU of the web server fleet is more or equal than 40% or decrease when is less or equal than 20%. The policies are created using the command “aws autoscaling put-scaling-policy”. CloudWatch is configured to alert and auto scale by one when CPU reaches the stablished limits using the above policies.



Look how low CPU alarm is in alert because actual load is at 0%.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente**Note:** auto scaling group is ranged from 1 to 4 instances**,** so it can’t create more than that and terminate the only one running.

Now, using the load balancer DNS Name, in a browser tab a load is generated from 1% to 76%.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Because of the high load new instances were created to satisfy the policy.