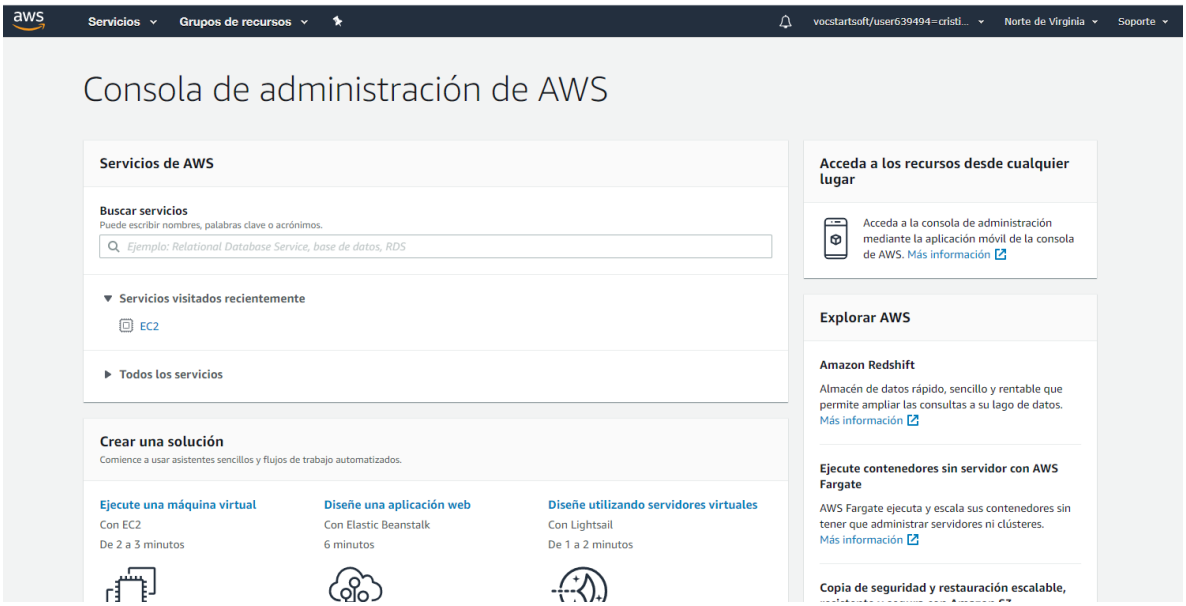
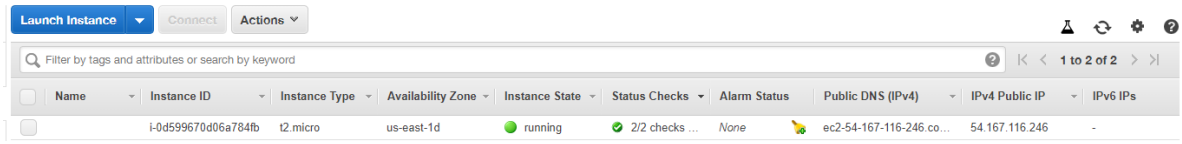
Cristian David López Arévalo

1. Acceda a la consola de administración de AWS

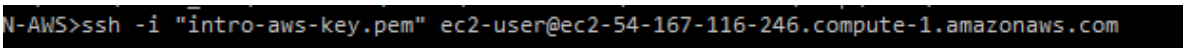


2. Cree una máquina virtual linux siguiendo los pasos en:https://aws.amazon.com/es/getting-started/tutorials/launch-a-virtual-machine/

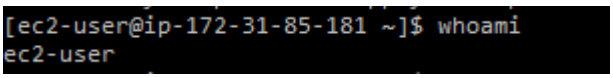


3. Conéctese a la máquina virtual usando ssh. Verifique que está en la máquina virtual introduciendo comandos simples como: whoami, ls, pwd.

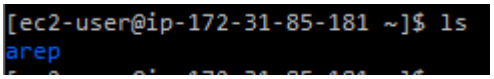
   Ssh



   Whoami



   Ls



Pwd

```
[ec2-user@ip-172-31-85-181 ~]$ pwd
/home/ec2-user
```

4. Verifique que java está instalado. Note que el compilador de java (javac) no está instalado en la máquina virtual.

```
[ec2-user@ip-172-31-85-181 ~]$ java -version
openjdk version "1.8.0_242"
OpenJDK Runtime Environment (build 1.8.0_242-b08)
OpenJDK 64-Bit Server VM (build 25.242-b08, mixed mode)
```

5. Salga del ssh usando "exit".

```
[ec2-user@ip-172-31-85-181 ~]$ exit
logout
Connection to ec2-54-167-116-246.compute-1.amazonaws.com closed.
```

6. En su **máquina local,** usando netbeans cree un cliente que se pueda conectar a una url e imprimir la respuesta de esa url en pantalla. Observe que el código de ejemplo recibe la url como el primer argumento en la línea de comandos.

```java
public class ClientServer implements Runnable {

    private URL url;
    private static int threads;
    static AtomicInteger succesA = new AtomicInteger(0);
    static AtomicInteger failA = new AtomicInteger(0);

    public ClientServer(String url1) {
        try {
            this.url = new URL(url1);
        } catch (MalformedURLException e) {
            System.out.println("MalformedURLException: " + e);
            e.printStackTrace();
        }
    }

    Run | Debug
    public static void main(final String[] args) {
        ClientServer cs = new ClientServer(args[0]);
        threads = Integer.parseInt(args[1]);
        Thread[] peticiones = new Thread[threads];
        for (int i = 0; i < threads; i++) {
            peticiones[i] = new Thread(cs);
            peticiones[i].start();
        }
        for (int i = 0; i < threads; i++) {
            try {
                peticiones[i].join();
            } catch (InterruptedException e) {
                e.printStackTrace();
                System.out.println("InterruptedException: " + e);
            }
        }
        System.out.println("succes: " + succesA + " " + " fail : " + failA);

    }

    @Override
    public void run() {
        try {
            BufferedReader reader = new BufferedReader(new InputStreamReader(url.openStream()));
            String inputLine = null;
            while ((inputLine = reader.readLine()) != null) {
                // System.out.println(inputLine);
            }
            succesA.addAndGet(1);
        } catch (final IOException x) {
            System.err.println(x);
            failA.addAndGet(1);
        }
    }
}
```

7. Pruebe su cliente en la máquina local con un comando similar al siguiente:

```
>mvn exec:java -Dexec.mainClass="edu.escuelaing.arep.cliente.ClientServer" -Dexec.args="http://www
.google.com 1"
[INFO] Scanning for projects...
[INFO]
[INFO] --------------------< edu.escuelaing.arep:cliente >---------------------
[INFO] Building cliente 1.0-SNAPSHOT
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- exec-maven-plugin:1.6.0:java (default-cli) @ cliente ---
succes: 1  fail : 0
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 2.991 s
[INFO] Finished at: 2020-03-09T21:35:34-05:00
[INFO] ------------------------------------------------------------------------
```

8. Suba el proyecto compilado a su máquina virtual usando sftp.

```
N-AWS>sftp -i "intro-aws-key.pem" ec2-user@ec2-54-167-116-246.compute-1.amazonaws.com
Connected to ec2-user@ec2-54-167-116-246.compute-1.amazonaws.com.
sftp> put -r target/classes
```

9. Ejecute el cliente que instaló en su máquina virtual de AWS para conectarse a la aplicación que instaló en Heroku durante el parcial o el taller.

```
[ec2-user@ip-172-31-85-181 target]$ java edu/escuelaing/arep/cliente/ClientServer https://shi
elded-coast-58752.herokuapp.com/index.html 400
```

```
[ec2-user@ip-172-31-85-181 target]$ java edu/escuelaing/arep/cliente/ClientServer https://shielded-coast-58752.herokuapp.com/index.html 400
succes: 400  fail : 0
```

10. Borre las instancias y unidades de almacenamiento en su cuenta AWS para no generar costos.