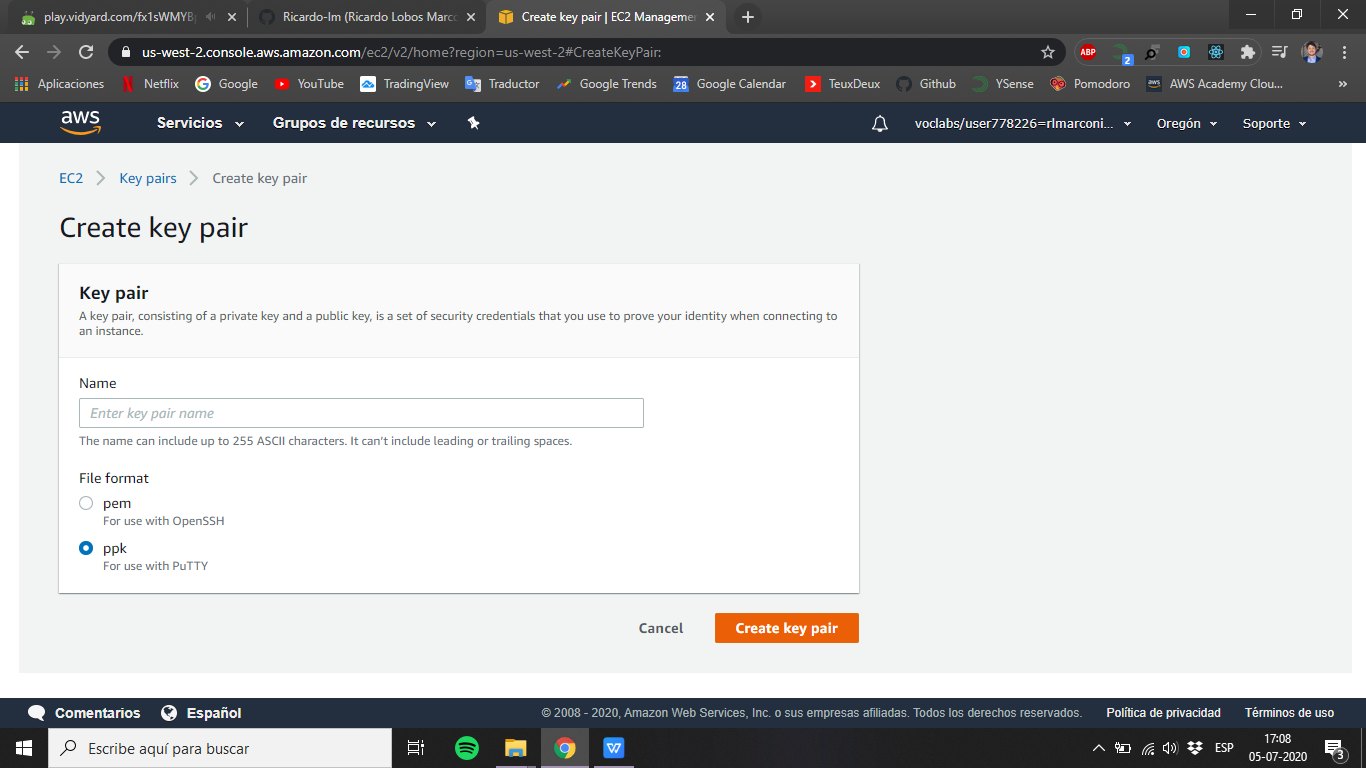
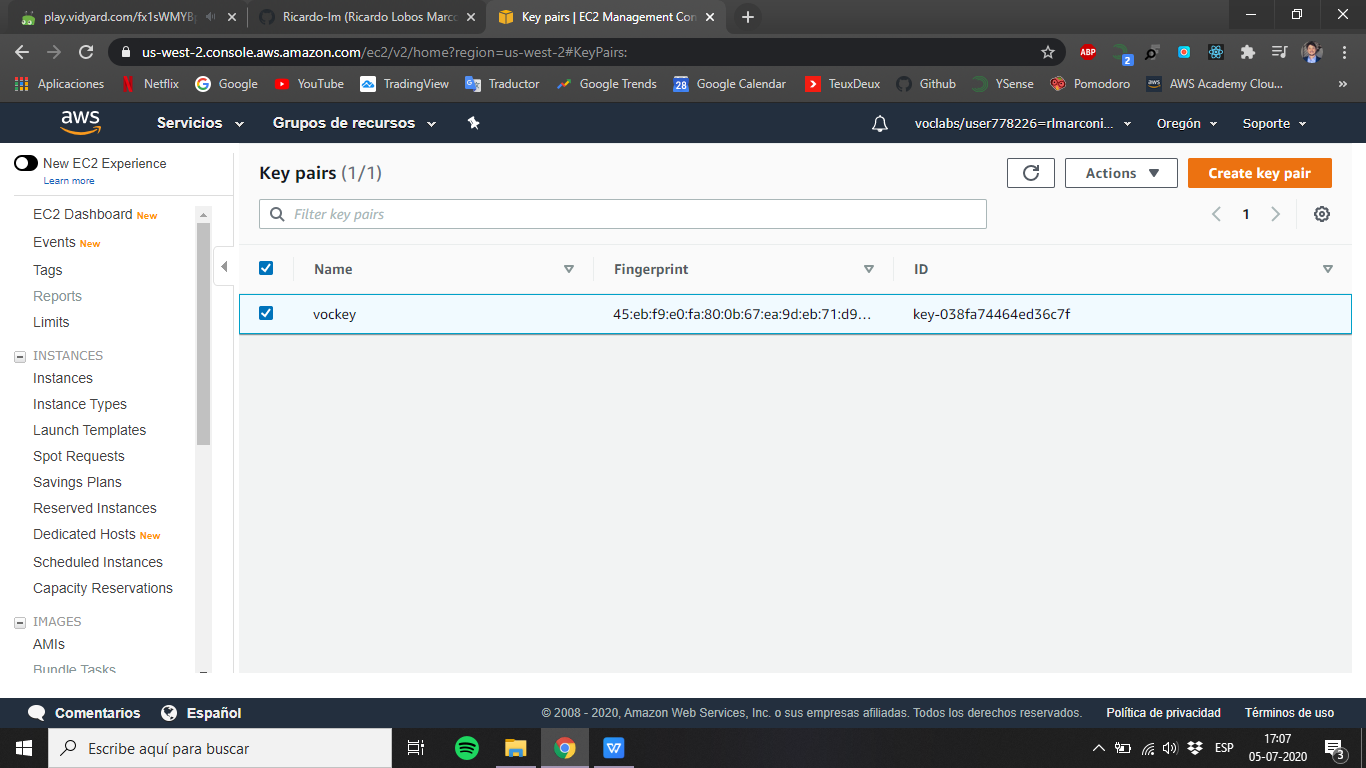
**DISEÑANDO UNA ARQUITECTURA DE 3 CAPAS EN AWS**

**CREAR KEY PAIR**

1. Ingresar a servicio EC2 opción Key Pair y dar click a “Create Key Pair”

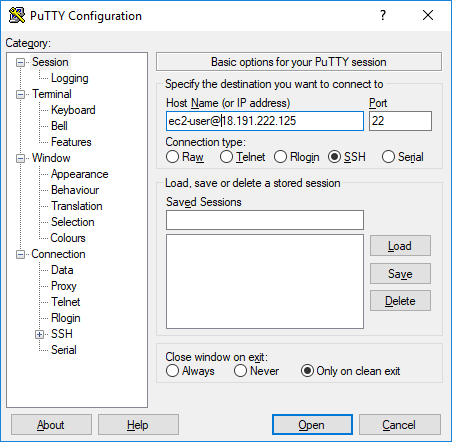


1. Escoger un nombre para la key pair, escoger ppk y descargar la key pair.

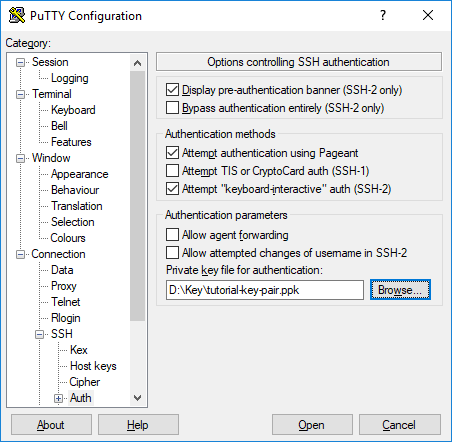


**CONECTARSE A SERVIDOR WEB**

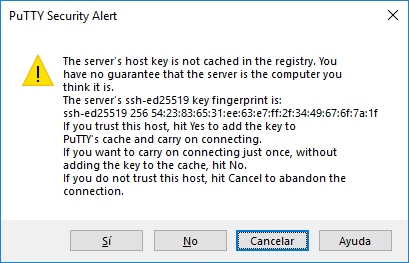
1. Ir al servicio EC2 / Instances y seleccionar el servidor creado para conocer su IP pública.
2. En el Putty colocar en hostname: “ec2-user@IP PUBLICA”



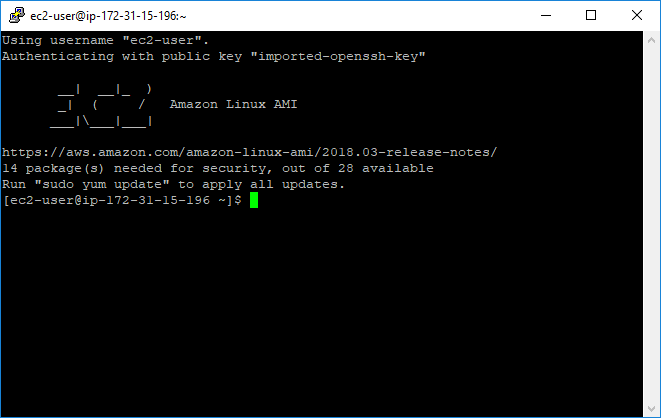
1. Ingresar en el menú de la izquierda en Connection / SSH / Auth y cargar la llave privada creada (\*.ppk)



1. Dar click en “Open”. Aparecerá una ventana de aviso a la que se debe dar “Si”



1. Con los pasos previos estarás conectado al servidor Web



**INSTALACIÓN DE UN SERVIDOR WEB APACHE CON PHP**

1. Actualizar servidor

|  |
| --- |
| ec2-user ~]$ **sudo yum update -y** |

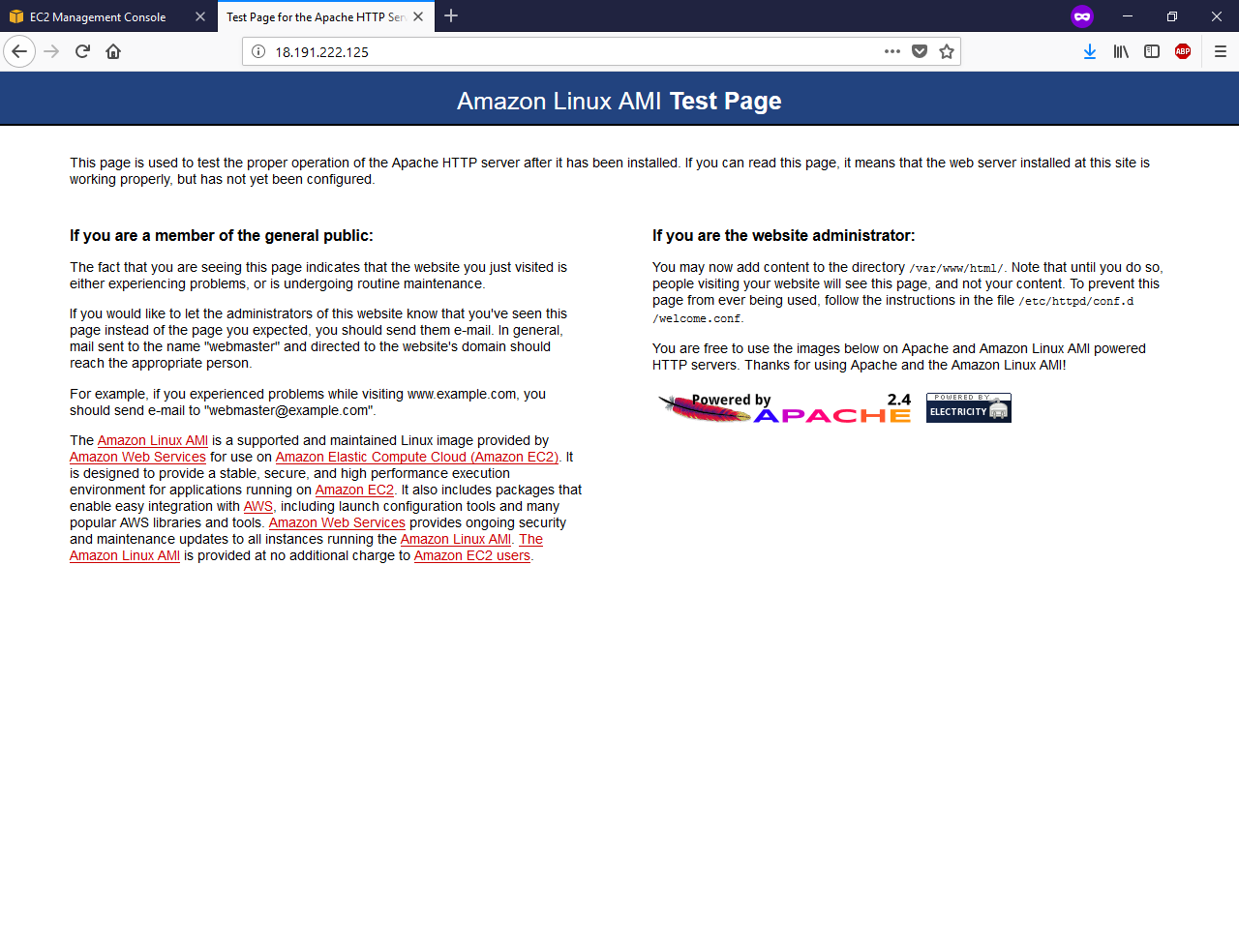
1. Instalar paquetes necesarios

|  |
| --- |
| [ec2-user ~]$ **sudo yum install -y httpd24 php56 php56-mysqlnd** |

1. Iniciar el servicio web

|  |
| --- |
| [ec2-user ~]$ **sudo service httpd start** |

1. Verificar el acceso a web a través de la IP pública. Se mostrará la página web de Apache por defecto.



1. Configure el servidor web para que se inicie en cada arranque del sistema utilizando el comando chkconfig:

|  |
| --- |
| [ec2-user ~]$ **sudo chkconfig httpd on** |

**CONEXIÓN DEL SERVIDOR WEB APACHE CON LA INSTANCIA DE BASE DE DATOS DE RDS**

1. Conectarse como root

|  |
| --- |
| [ec2-user~]$ sudo su  [root~]# |

1. Mientras está conectado a la instancia EC2, cambie el directorio a /var/www y cree un subdirectorio nuevo denominado inc y cree un archivo en el directorio inc, denominado dbinfo.inc y, a continuación, edite el archivo mediante vi (o a cualquier otro editor de su elección).

|  |
| --- |
| [root ~]$ cd /var/www  [root ~]$ mkdir inc  [root ~]$ cd inc  [root ~]$ vi dbinfo.inc |

1. Añada el siguiente contenido al archivo dbinfo.inc, donde endpoint es el punto de enlace de la instancia de base de datos MySQL de RDS, sin el puerto.

|  |
| --- |
| <?php  define('DB\_SERVER', 'bdmysqldev.ciuvttjlfpnt.us-east-1.rds.amazonaws.com');  define('DB\_USERNAME', 'tutorial\_user');  define('DB\_PASSWORD', 'Elementi2020');  define('DB\_DATABASE', 'mysql');  ?> |

1. Cambie el directorio a /var/www/html y cree un archivo en el directorio html, denominado SamplePage.php y, a continuación, edite el archivo mediante vi (o a cualquier otro editor de su elección).

|  |
| --- |
| [root ~]$ cd /var/www/html  [root ~]$ vi SamplePage.php |

1. Añada el siguiente contenido al archivo SamplePage.php

|  |
| --- |
| <?php include "../inc/dbinfo.inc"; ?>  <html>  <body>  <h1>Sample page</h1>  <?php  /\* Connect to MySQL and select the database. \*/  $connection = mysqli\_connect(DB\_SERVER, DB\_USERNAME, DB\_PASSWORD);  if (mysqli\_connect\_errno()) echo "Failed to connect to MySQL: " . mysqli\_connect\_error();  $database = mysqli\_select\_db($connection, DB\_DATABASE);  /\* Ensure that the EMPLOYEES table exists. \*/  VerifyEmployeesTable($connection, DB\_DATABASE);  /\* If input fields are populated, add a row to the EMPLOYEES table. \*/  $employee\_name = htmlentities($\_POST['NAME']);  $employee\_address = htmlentities($\_POST['ADDRESS']);  if (strlen($employee\_name) || strlen($employee\_address)) {  AddEmployee($connection, $employee\_name, $employee\_address);  }  ?>  <!-- Input form -->  <form action="<?PHP echo $\_SERVER['SCRIPT\_NAME'] ?>" method="POST">  <table border="0">  <tr>  <td>NAME</td>  <td>ADDRESS</td>  </tr>  <tr>  <td>  <input type="text" name="NAME" maxlength="45" size="30" />  </td>  <td>  <input type="text" name="ADDRESS" maxlength="90" size="60" />  </td>  <td>  <input type="submit" value="Add Data" />  </td>  </tr>  </table>  </form>  <!-- Display table data. -->  <table border="1" cellpadding="2" cellspacing="2">  <tr>  <td>ID</td>  <td>NAME</td>  <td>ADDRESS</td>  </tr>  <?php  $result = mysqli\_query($connection, "SELECT \* FROM EMPLOYEES");  while($query\_data = mysqli\_fetch\_row($result)) {  echo "<tr>";  echo "<td>",$query\_data[0], "</td>",  "<td>",$query\_data[1], "</td>",  "<td>",$query\_data[2], "</td>";  echo "</tr>";  }  ?>  </table>  <!-- Clean up. -->  <?php  mysqli\_free\_result($result);  mysqli\_close($connection);  ?>  </body>  </html>  <?php  /\* Add an employee to the table. \*/  function AddEmployee($connection, $name, $address) {  $n = mysqli\_real\_escape\_string($connection, $name);  $a = mysqli\_real\_escape\_string($connection, $address);  $query = "INSERT INTO EMPLOYEES (NAME, ADDRESS) VALUES ('$n', '$a');";  if(!mysqli\_query($connection, $query)) echo("<p>Error adding employee data.</p>");  }  /\* Check whether the table exists and, if not, create it. \*/  function VerifyEmployeesTable($connection, $dbName) {  if(!TableExists("EMPLOYEES", $connection, $dbName))  {  $query = "CREATE TABLE EMPLOYEES (  ID int(11) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,  NAME VARCHAR(45),  ADDRESS VARCHAR(90)  )";  if(!mysqli\_query($connection, $query)) echo("<p>Error creating table.</p>");  }  }  /\* Check for the existence of a table. \*/  function TableExists($tableName, $connection, $dbName) {  $t = mysqli\_real\_escape\_string($connection, $tableName);  $d = mysqli\_real\_escape\_string($connection, $dbName);  $checktable = mysqli\_query($connection,  "SELECT TABLE\_NAME FROM information\_schema.TABLES WHERE TABLE\_NAME = '$t' AND TABLE\_SCHEMA = '$d'");  if(mysqli\_num\_rows($checktable) > 0) return true;  return false;  }  ?> |

1. Instalar los siguientes paquetes

|  |
| --- |
| wget http://repo.mysql.com/mysql-community-release-el7-5.noarch.rpm  sudo rpm -ivh mysql-community-release-el7-5.noarch.rpm  yum -y update  yum -y install mysql-community-client-5.6.43-2.el7.x86\_64 |

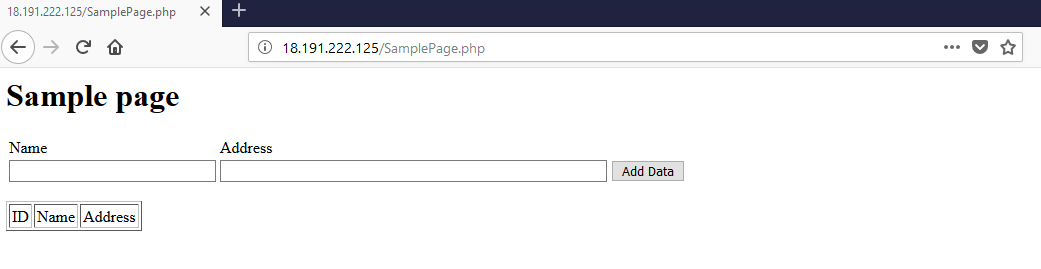
1. Conectarse a la base de datos

|  |
| --- |
| mysql -h nombre\_servidor -u nombre\_usuario -p |

1. Ejecutar en el mysql

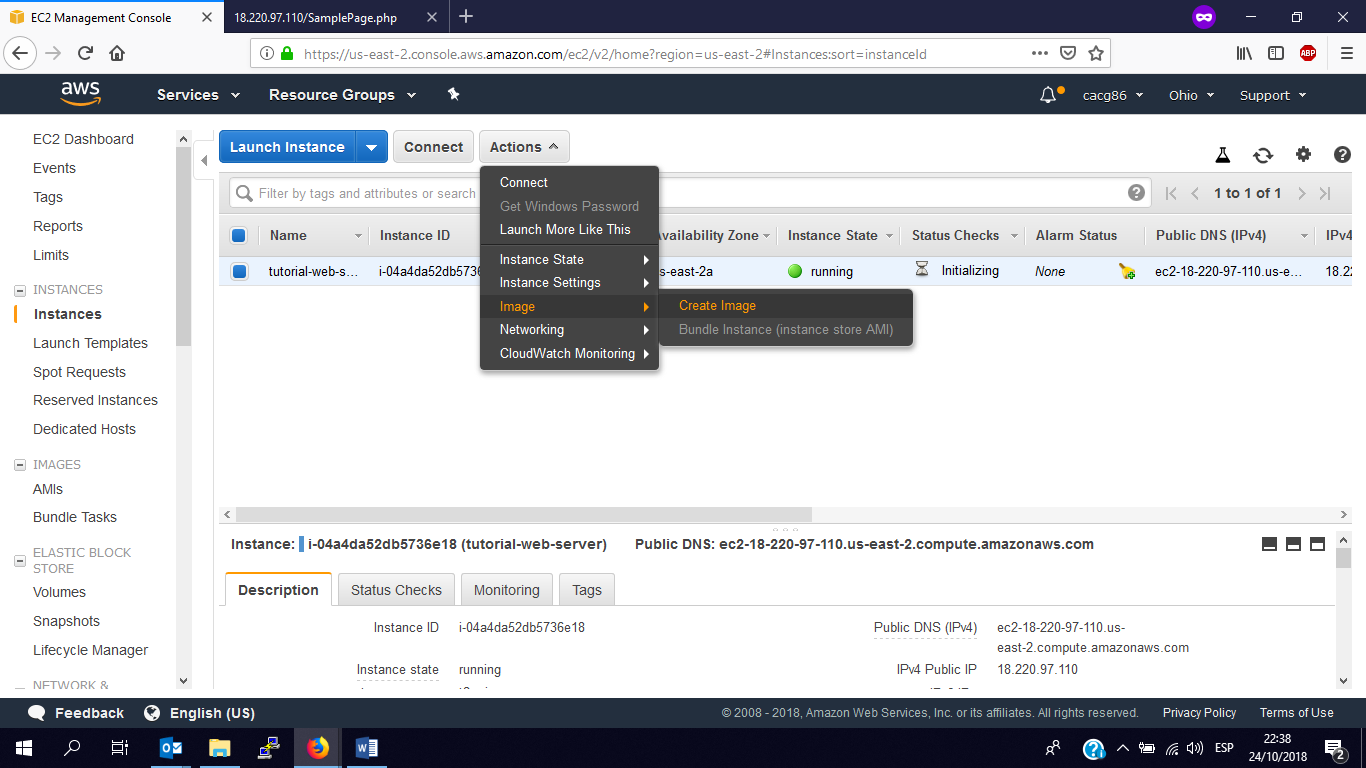
|  |
| --- |
| CREATE DATABASE sample;  use sample;  CREATE TABLE EMPLOYEES (  ID int(11) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,  NAME VARCHAR(45),  ADDRESS VARCHAR(90)  ); |

1. Ingresar a la web para verificar el buen funcionamiento de la Web: IP/SamplePage.php



**CREACIÓN DE AMI**

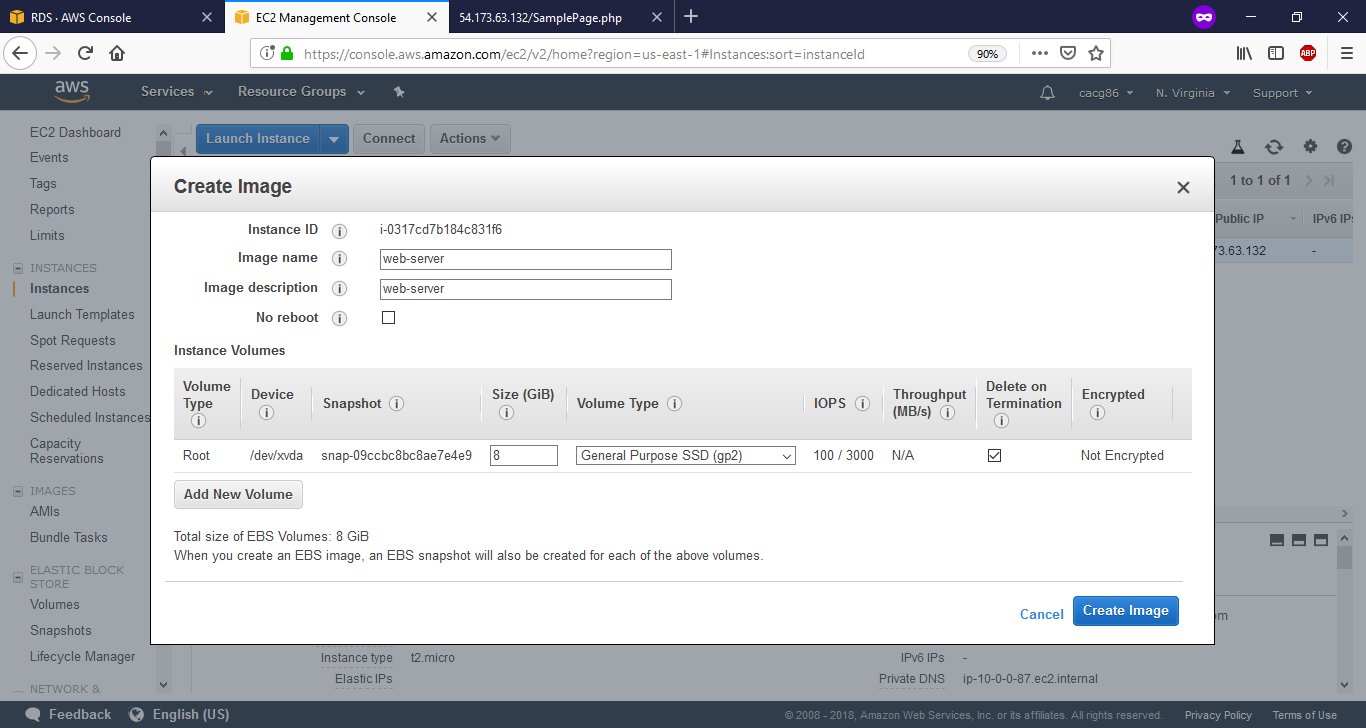
1. Ingresar EC2 / Instancias y seleccionar el servidor web creado. Dar click a la opción “créate image”, según la siguiente imagen



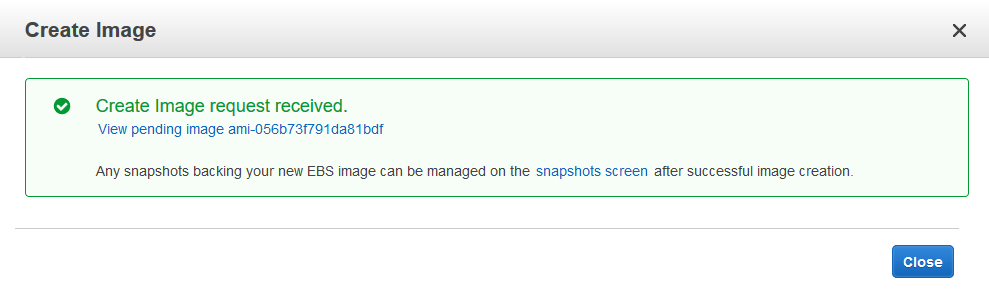
1. Completar los datos solicitados

Image name: web-server

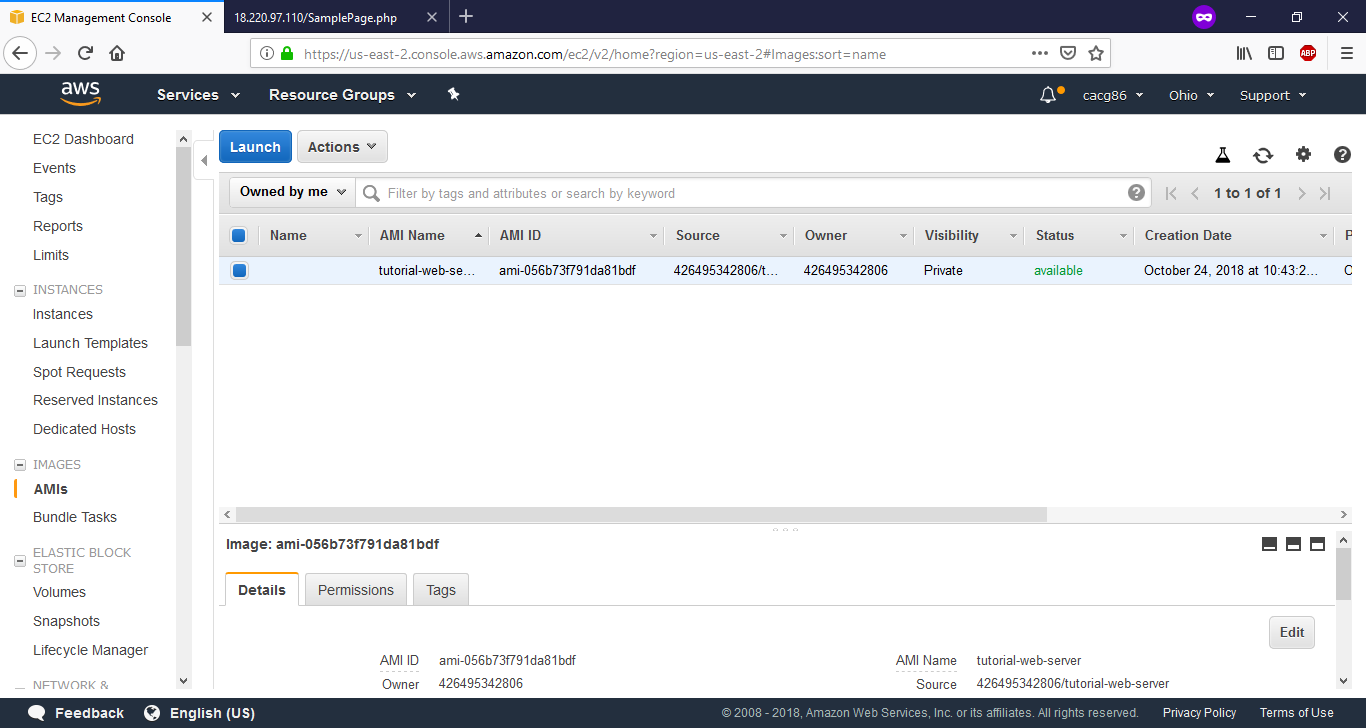
Image description: web-server



1. Aparecerá el aviso que la imagen está siendo creada.

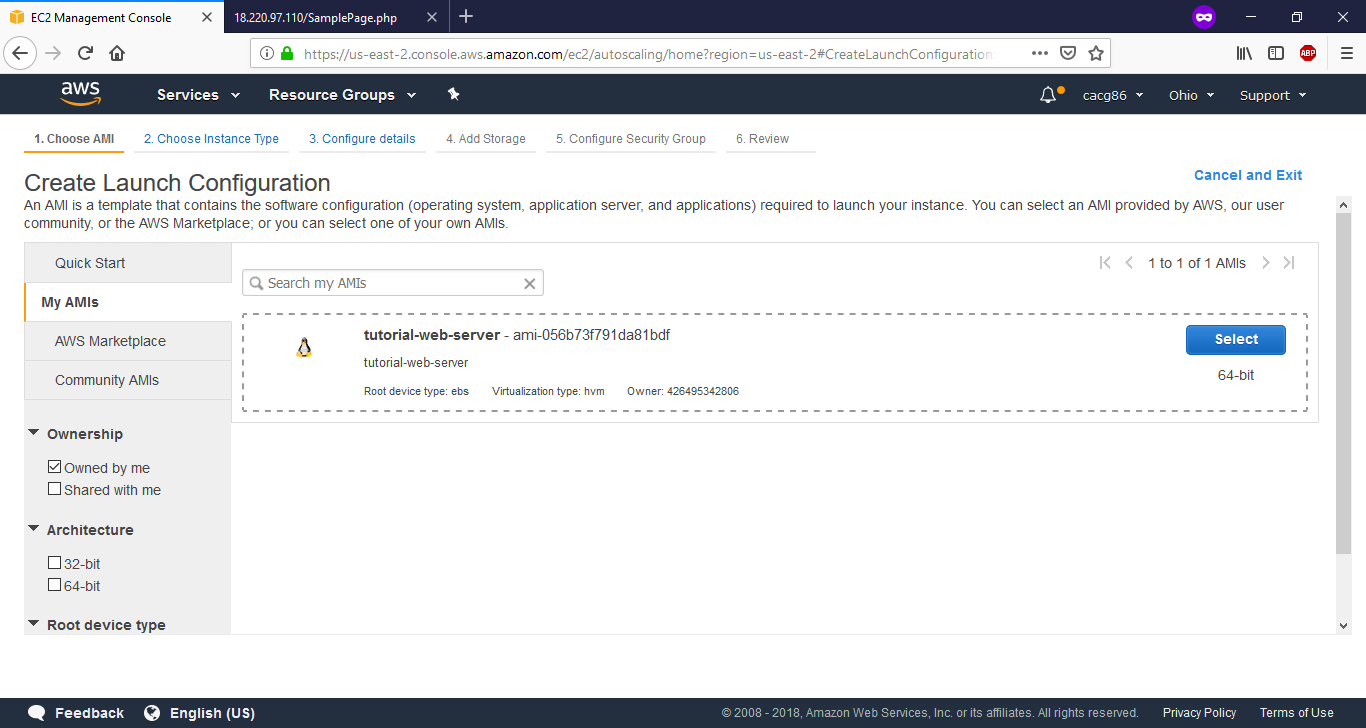


1. Ingresar a EC2 / AMIs y confirmar que el estado de la imagen es “available”

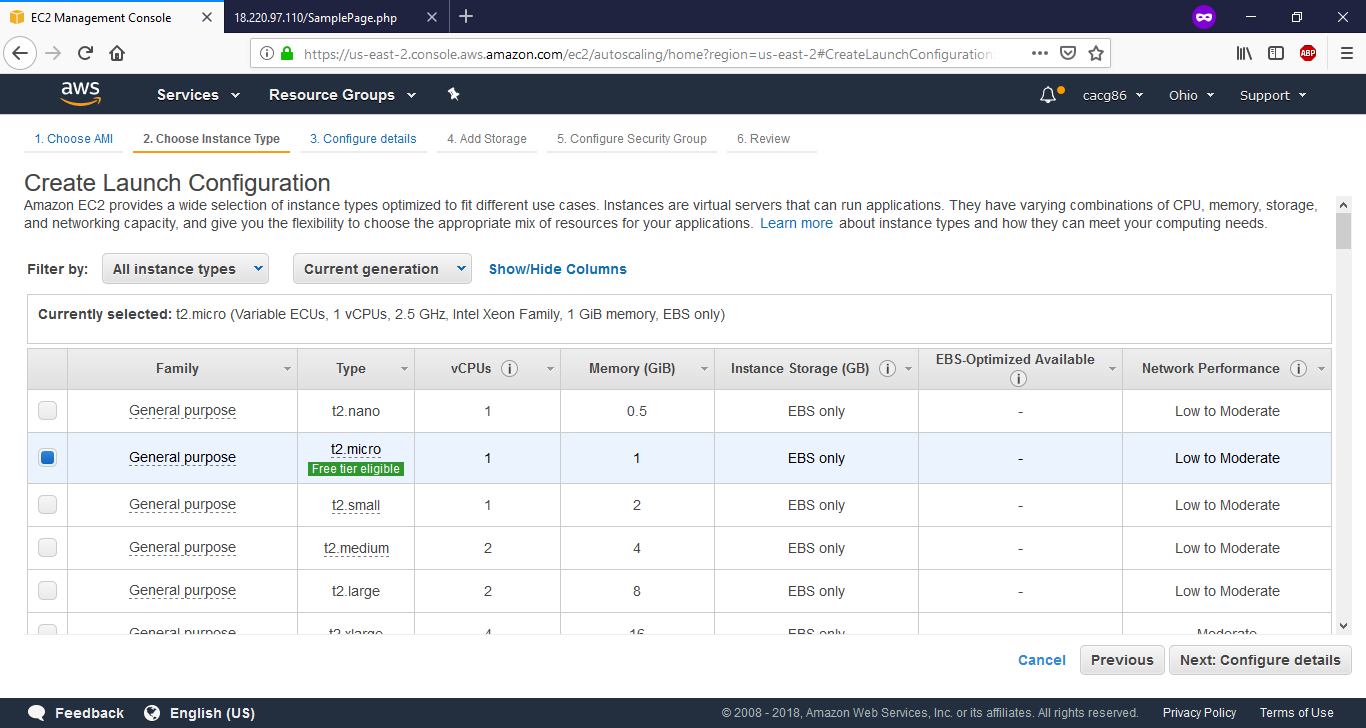


**CREACIÓN DE AUTOSCALING GROUP**

1. Ingresar a servicio EC2 / Auto scaling / Launch Configuration y dar click a “Create Launch configuration”:
2. Seleccionar “My Amis” y elegir AMI creada

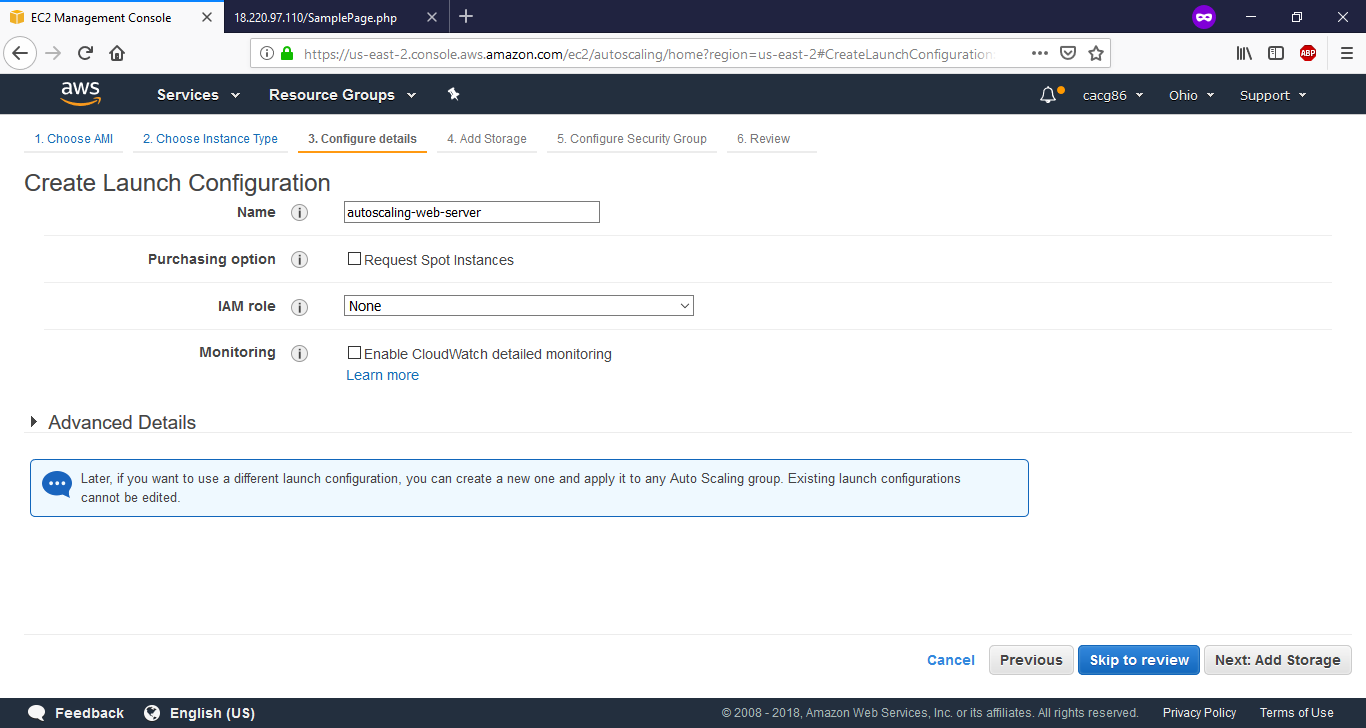


1. Elegir en tamaño t2.micro

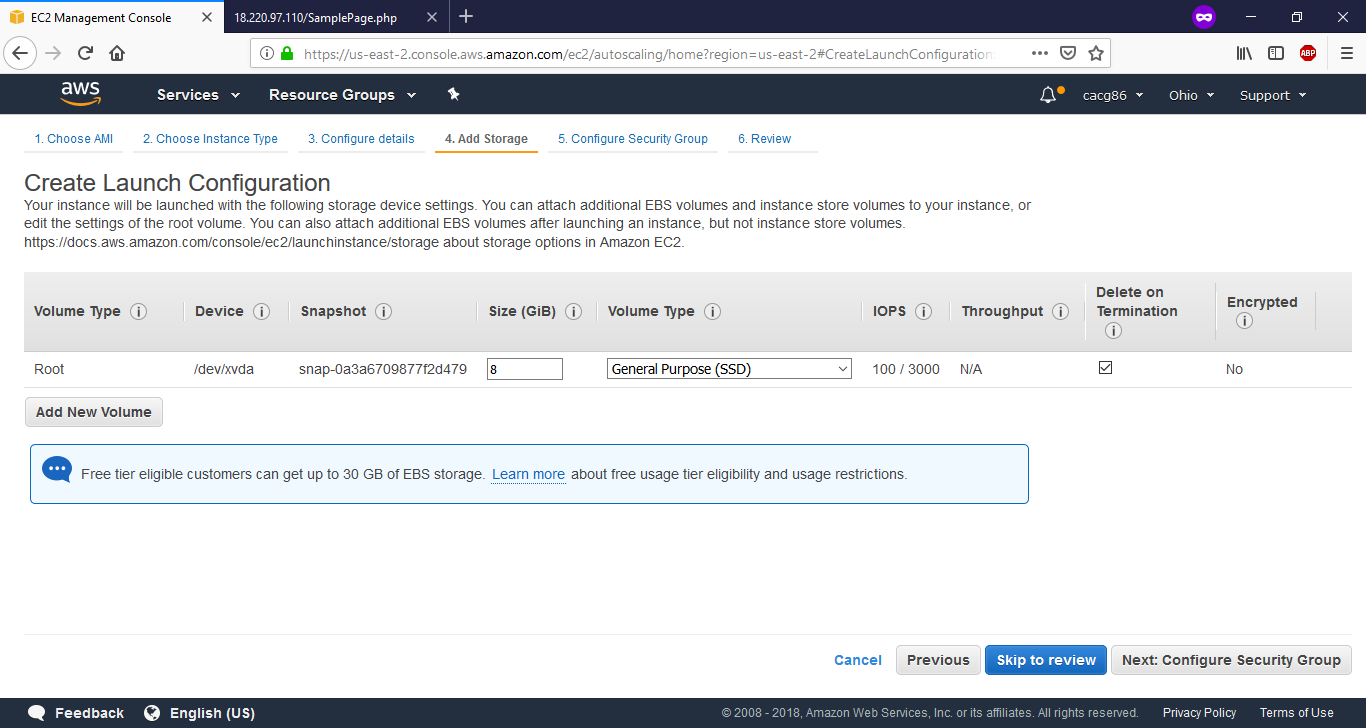


1. Indicar el nombre del autoscaling group

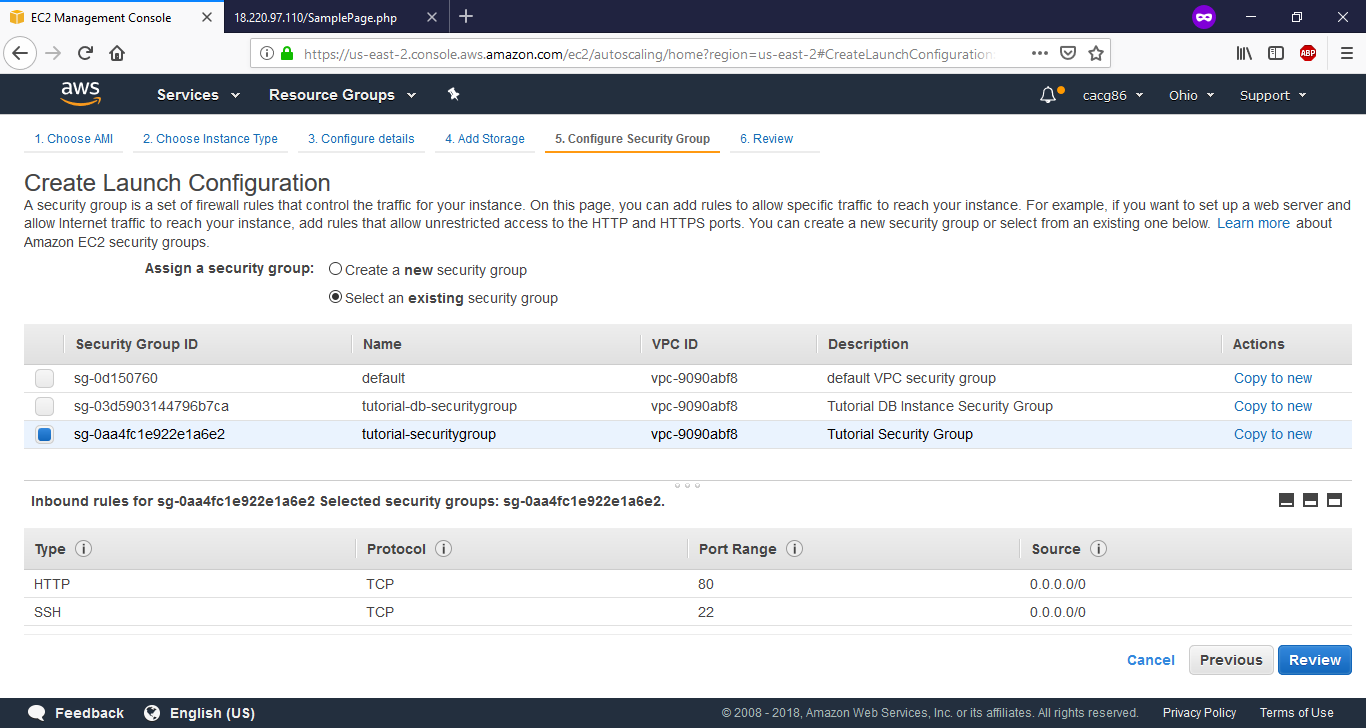
Name: autoscaling-web-server



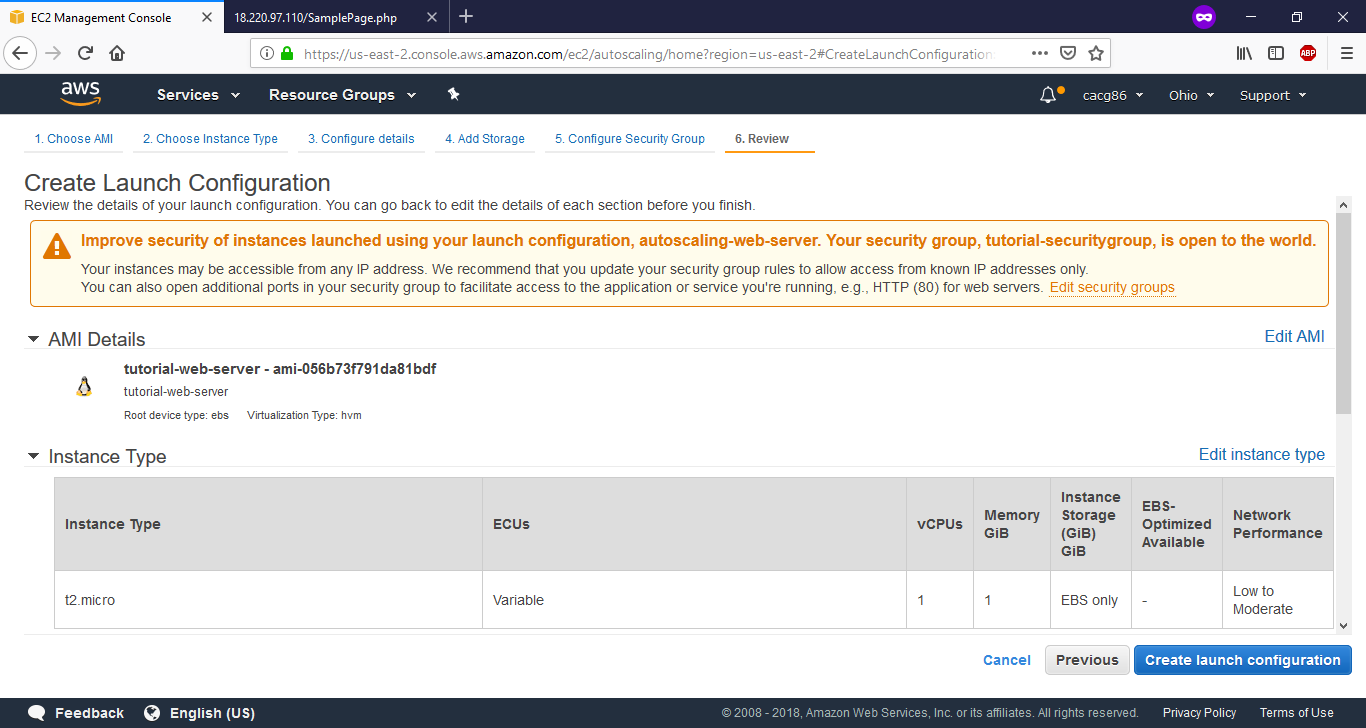
1. Dejar por defecto el Storage



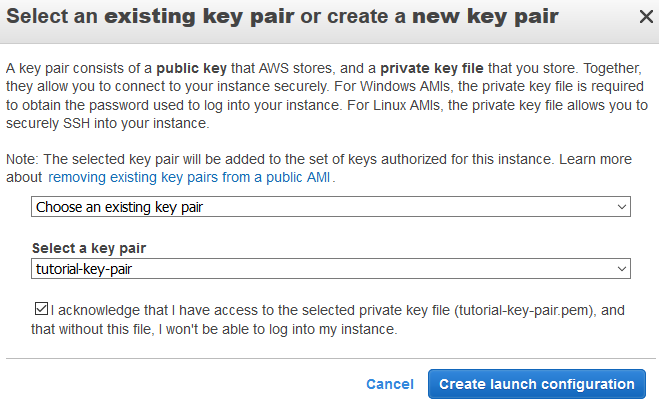
1. Escoger el security group de acceso a puerto 80



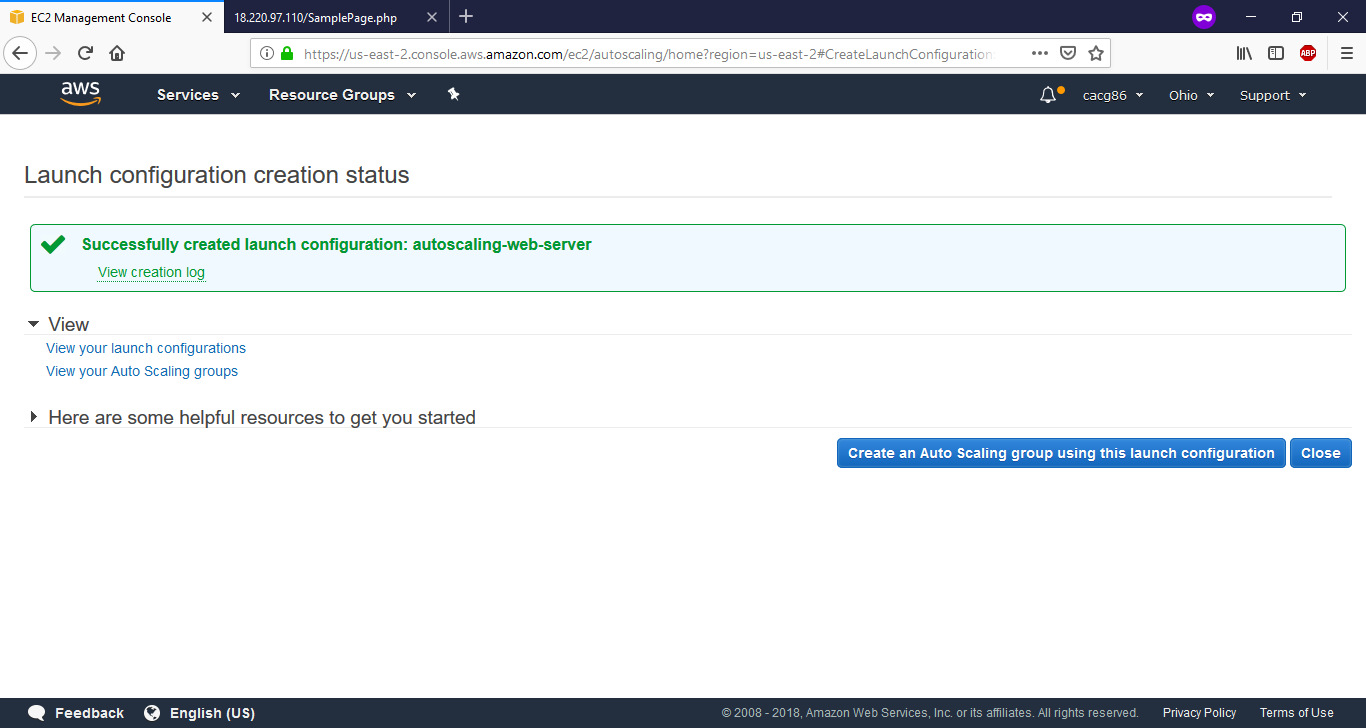
1. Dar click a la creación



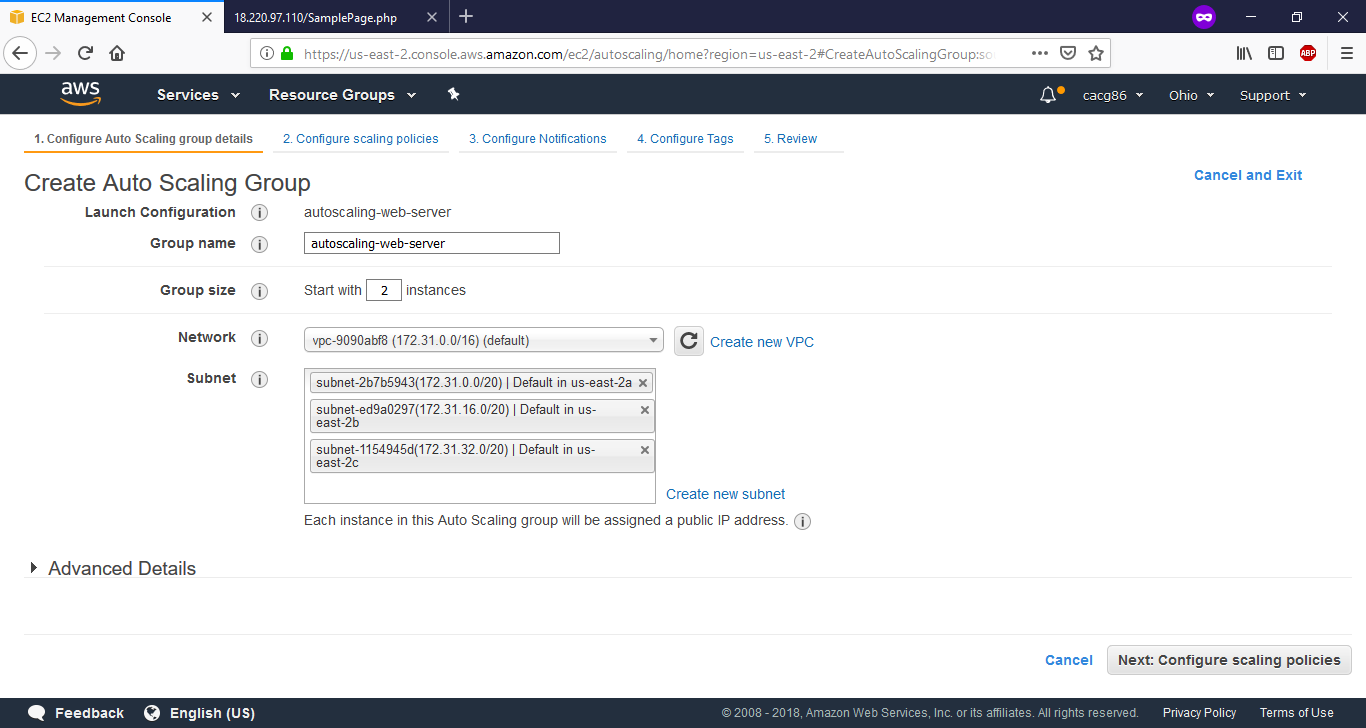
1. Escoger la llave anteriormente creada



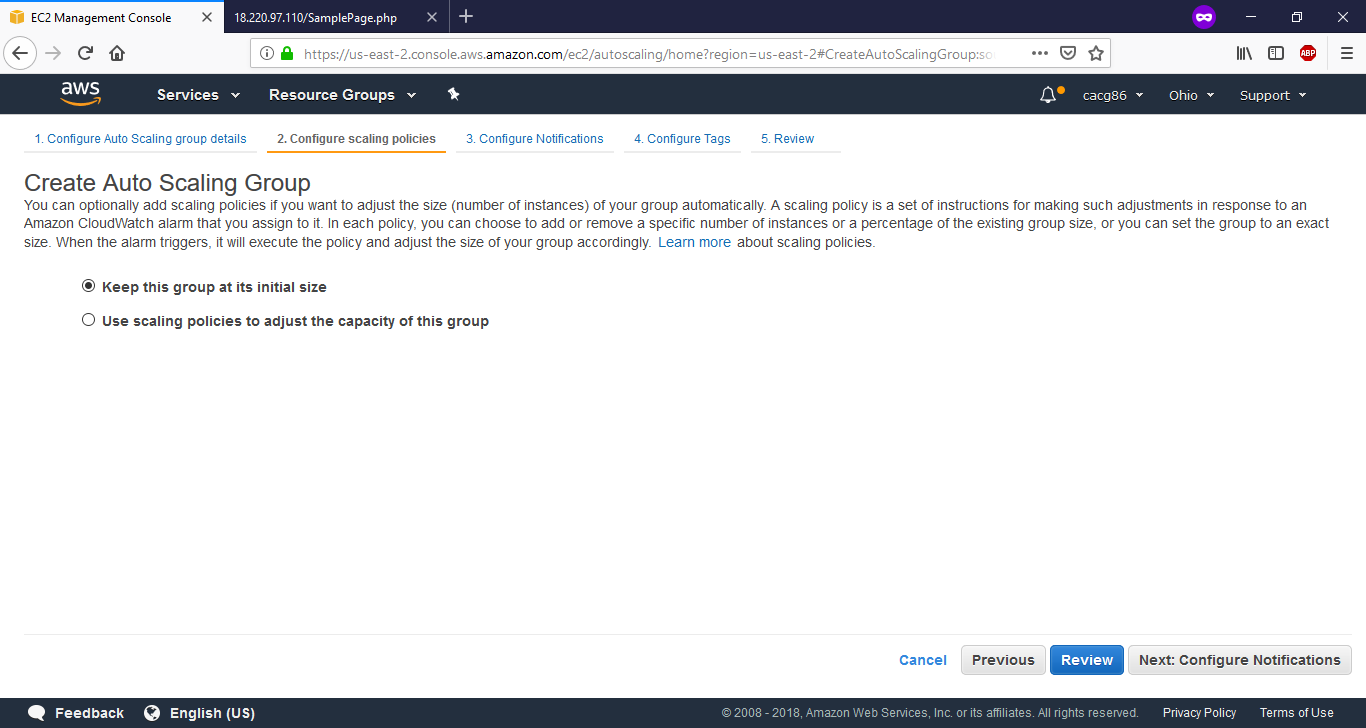
1. Dar click a “Create an Auto scaling group usin this launch configuration”



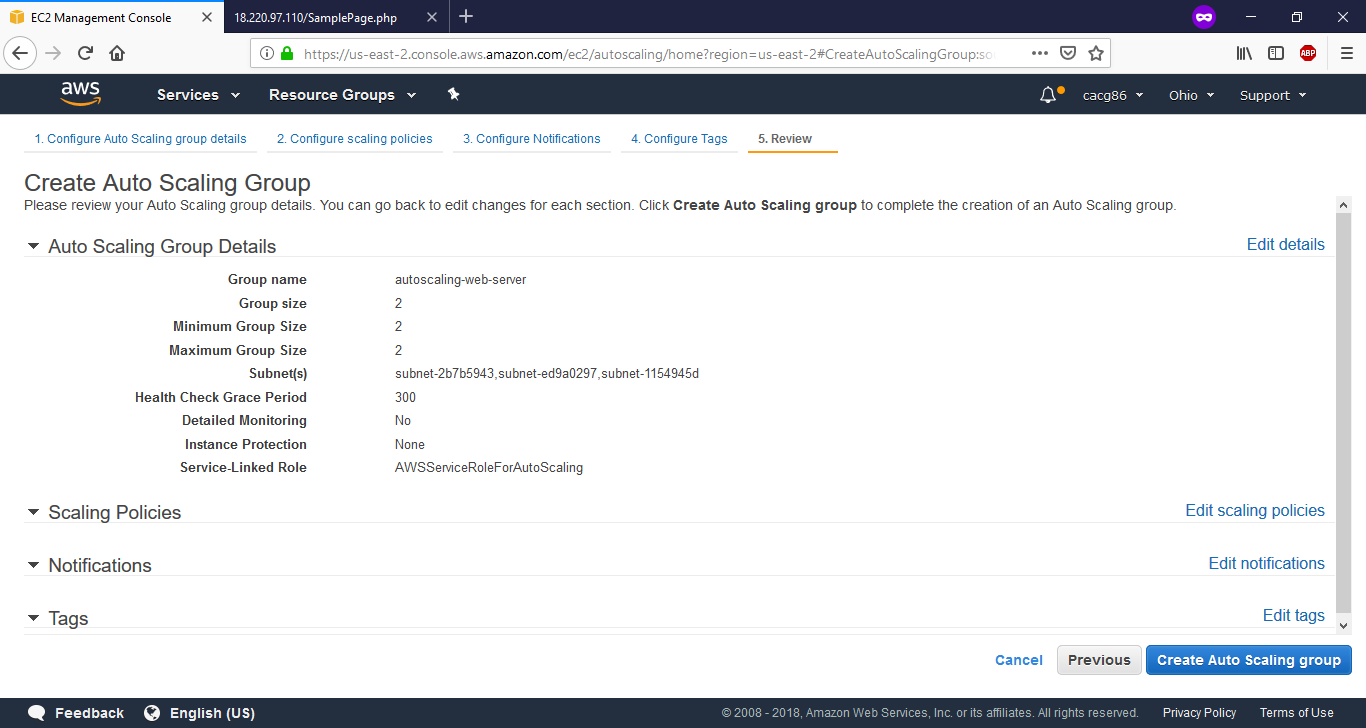
1. Crear el autoscaling group, brindar nombre y empezar con 2 instancias. Seleccionar en subnets las 3 creadas por default.



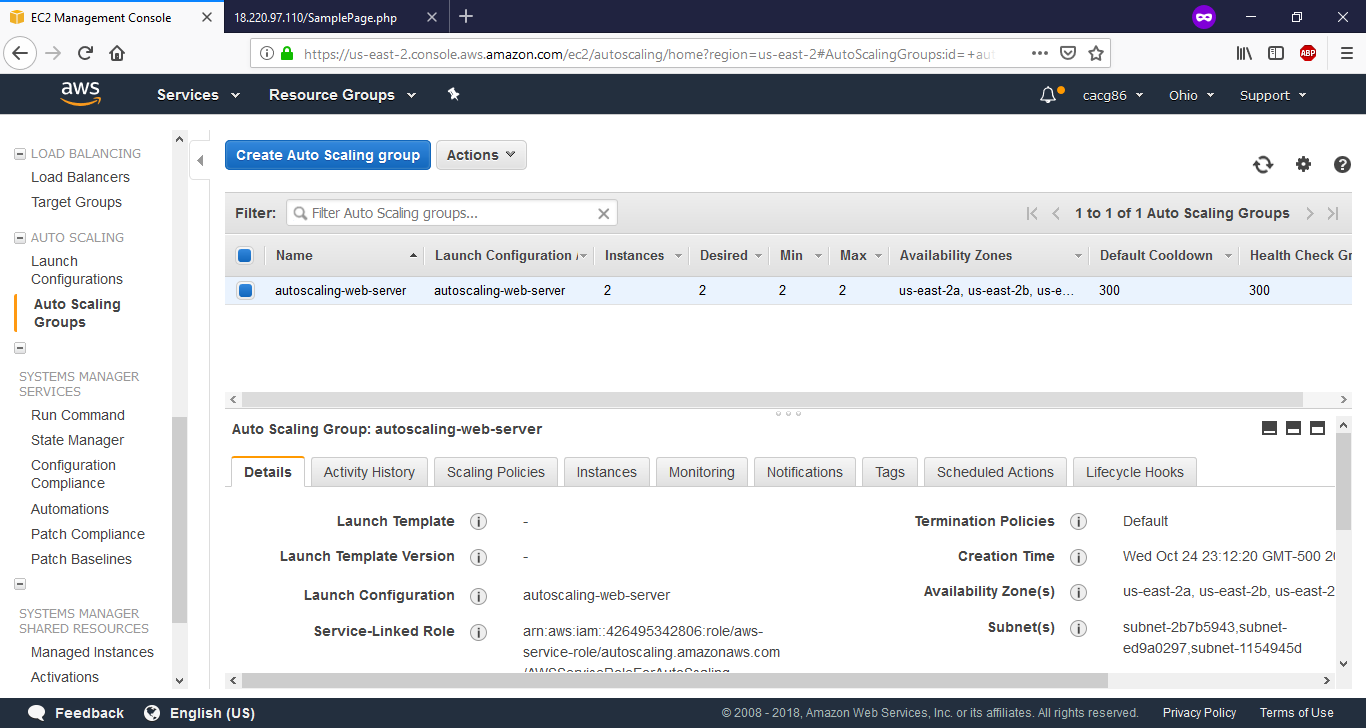
1. Dejar la opción por defecto y darle “Review”



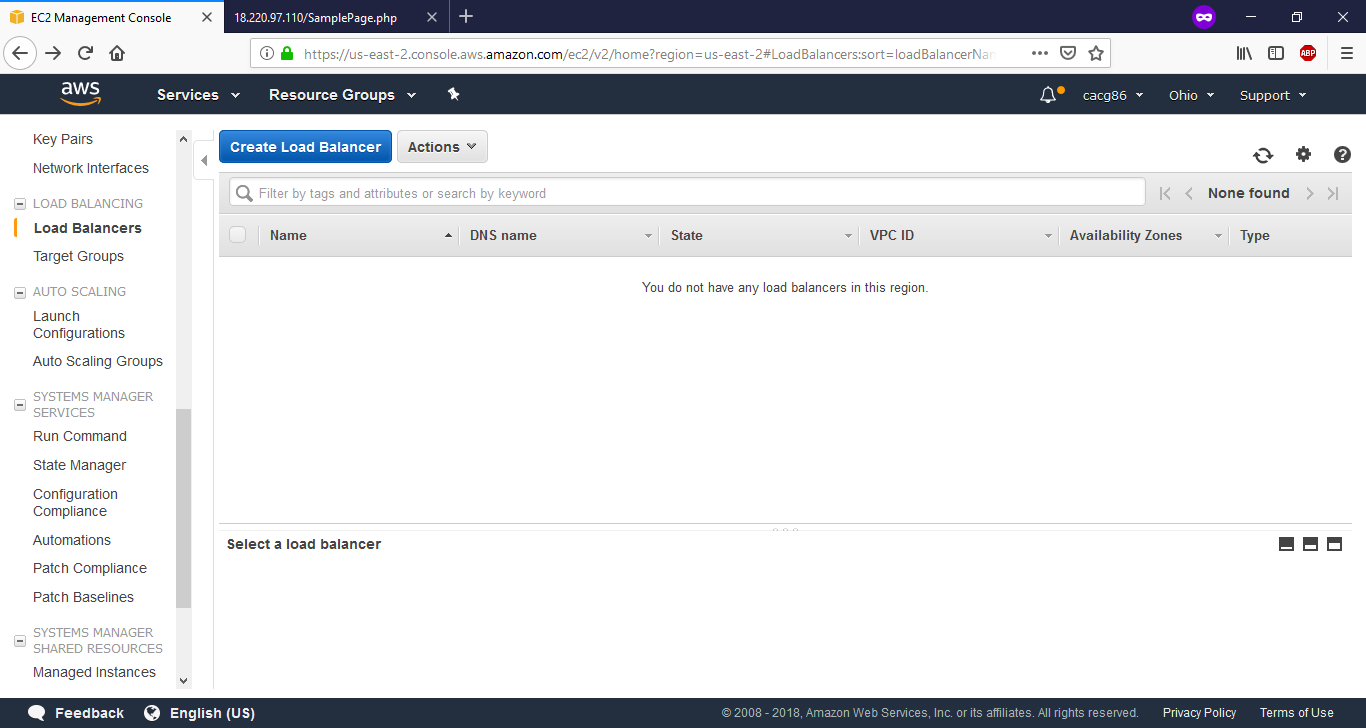
1. Crear el autoscaling group



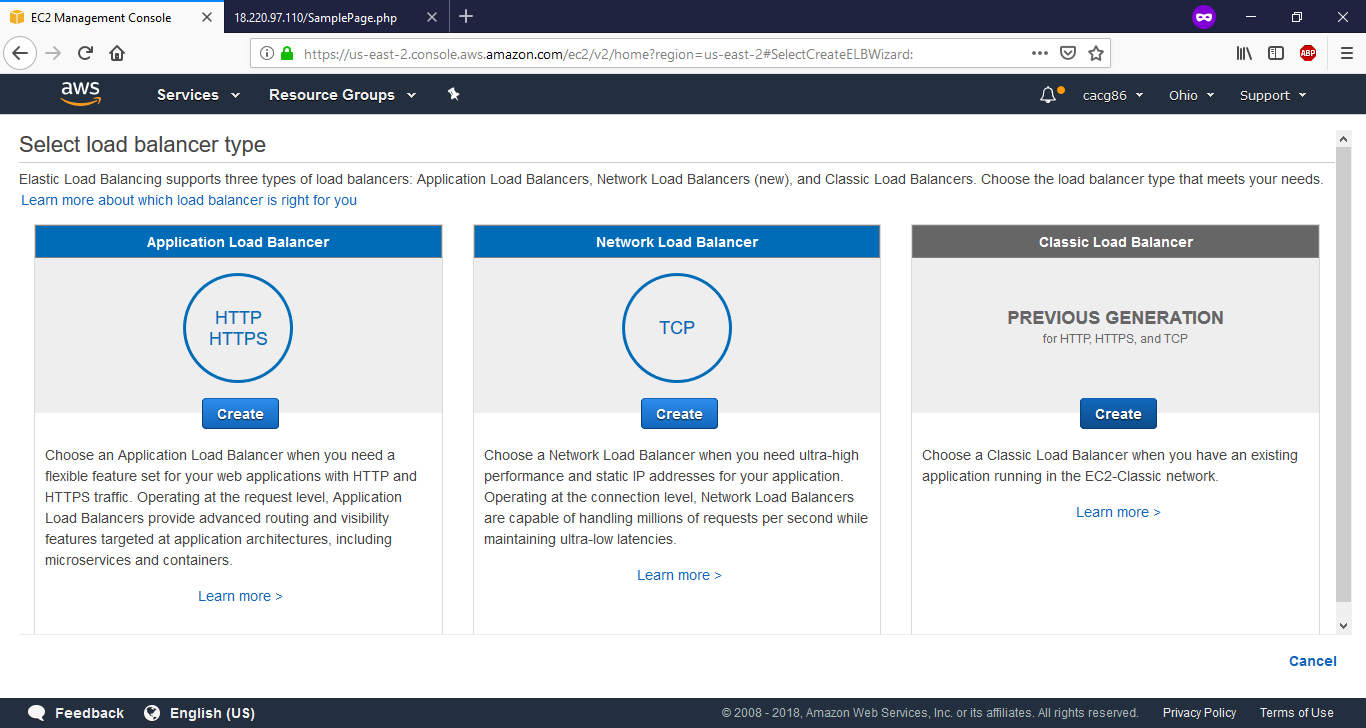
1. Esperar hasta que el autoscaling group cree las instancias indicada.



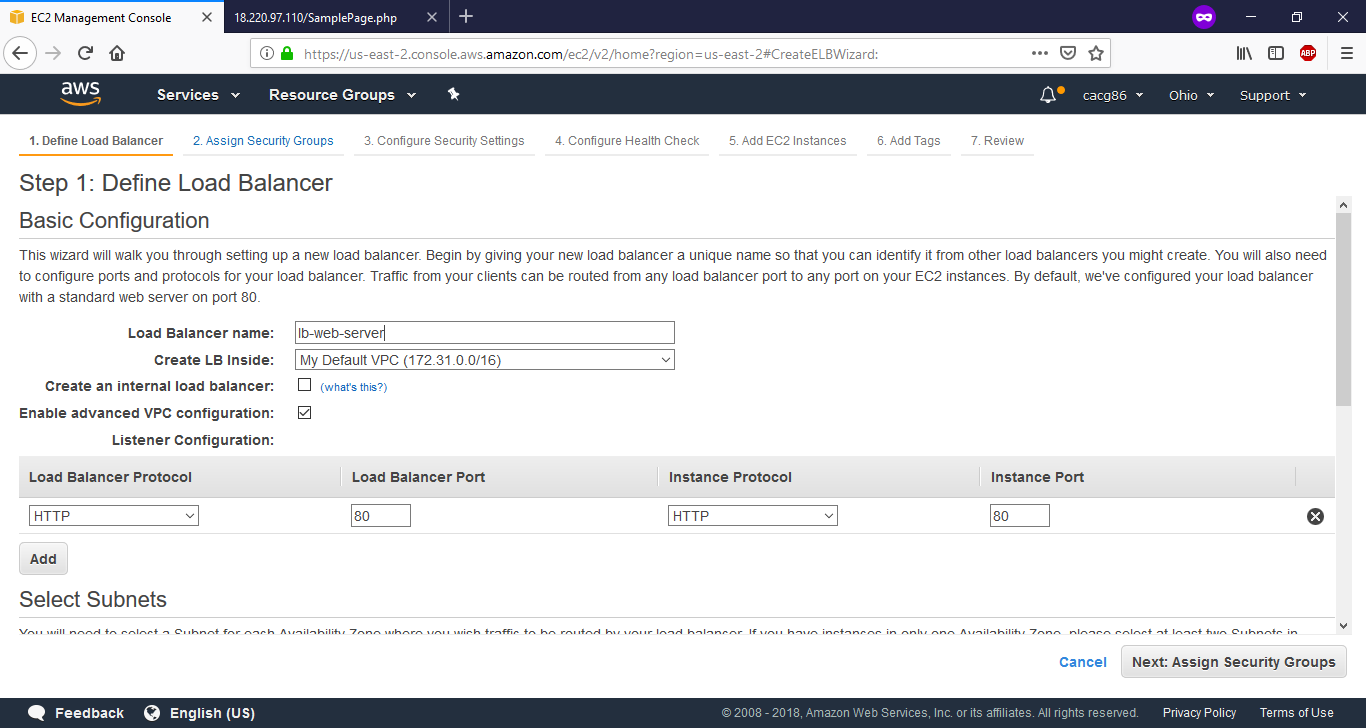
1. Ingresar a EC2 / Load Balancing / Load balancer y dar click a “Create load balancer”

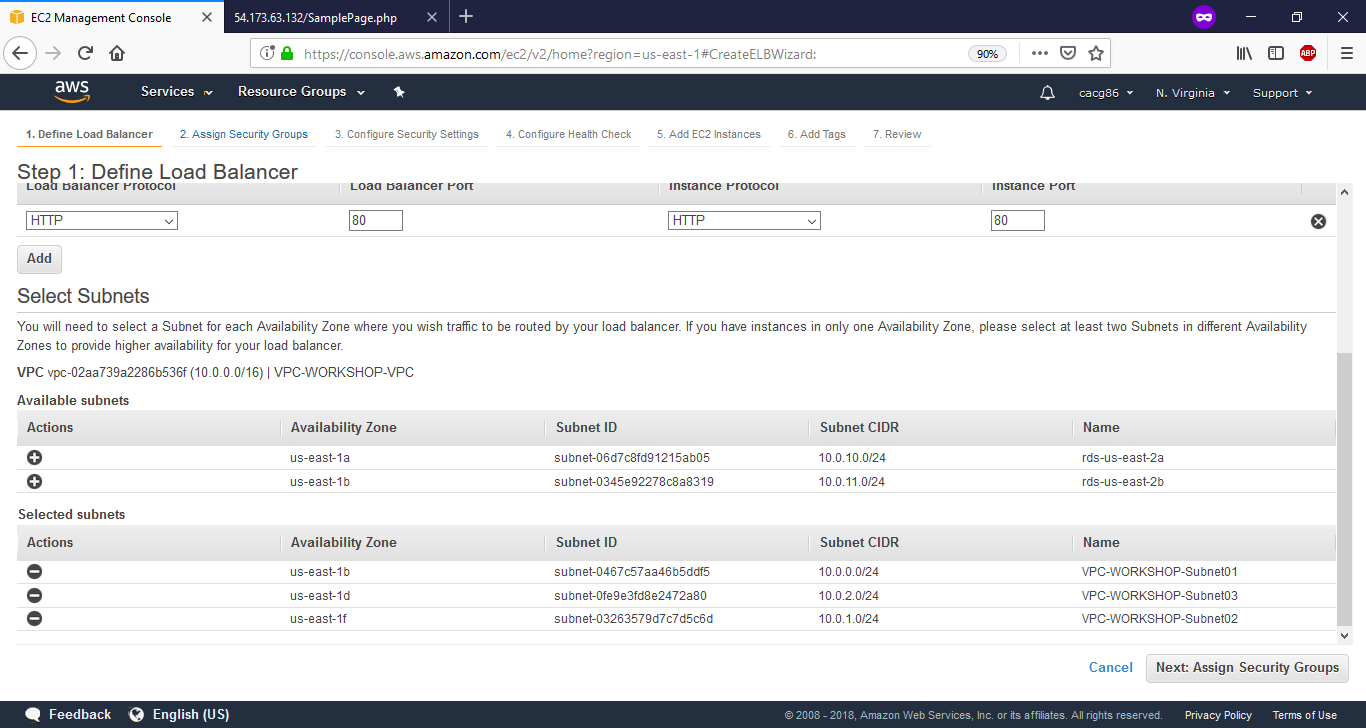


1. Escoger balanceador clásico

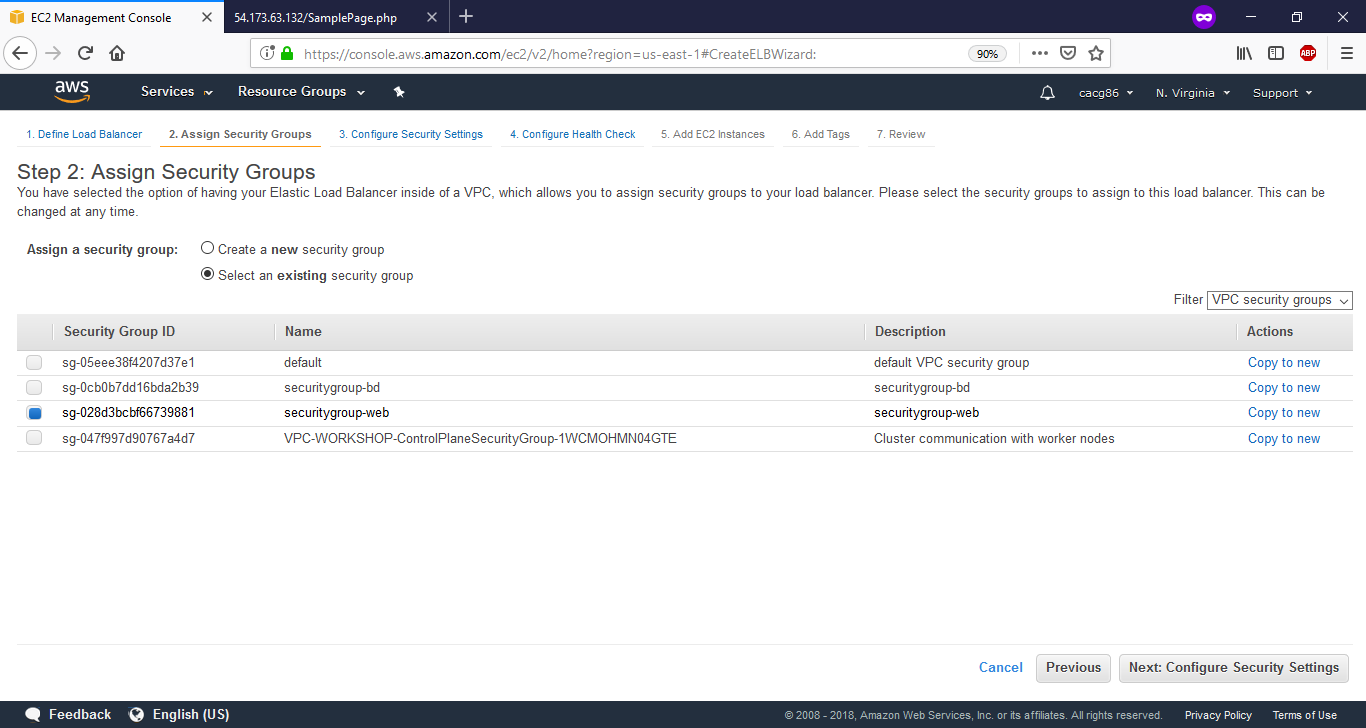


1. Brindar nombre del balanceador, habilitar la configuración de la VPC y escoger las subnet creadas en un inicio.

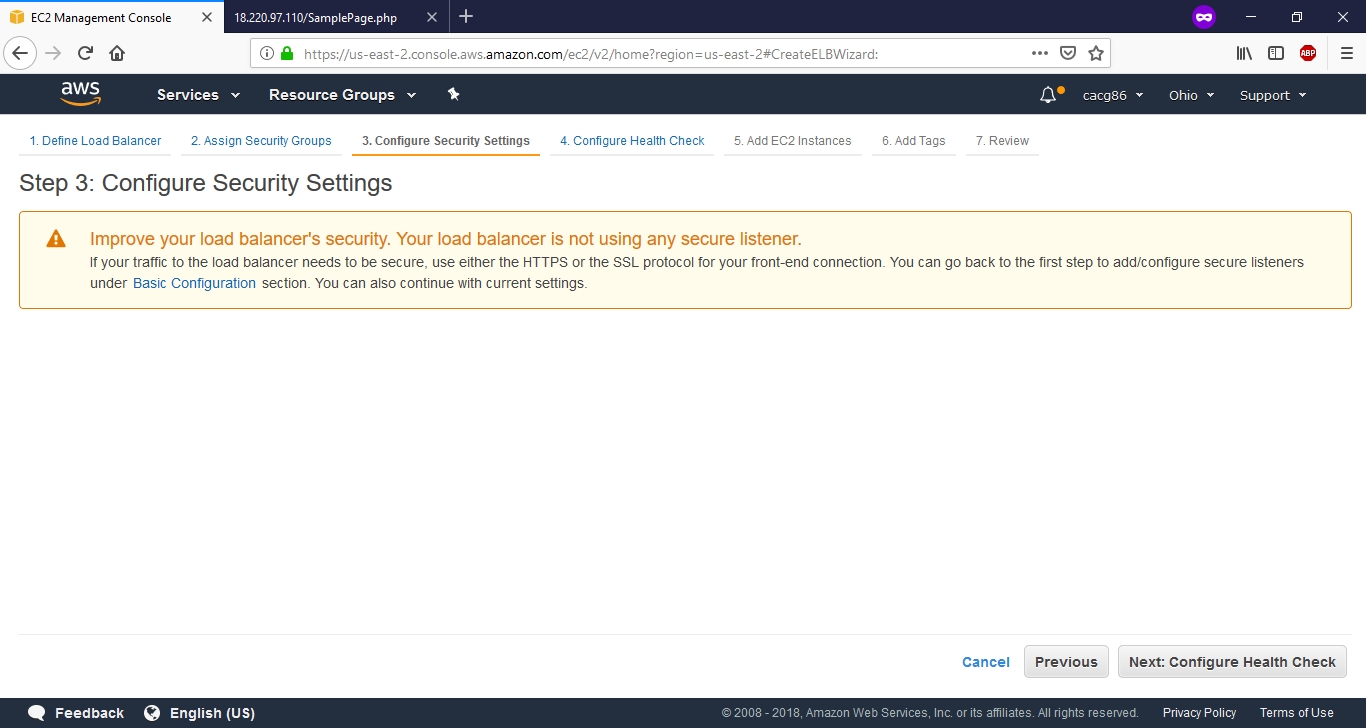




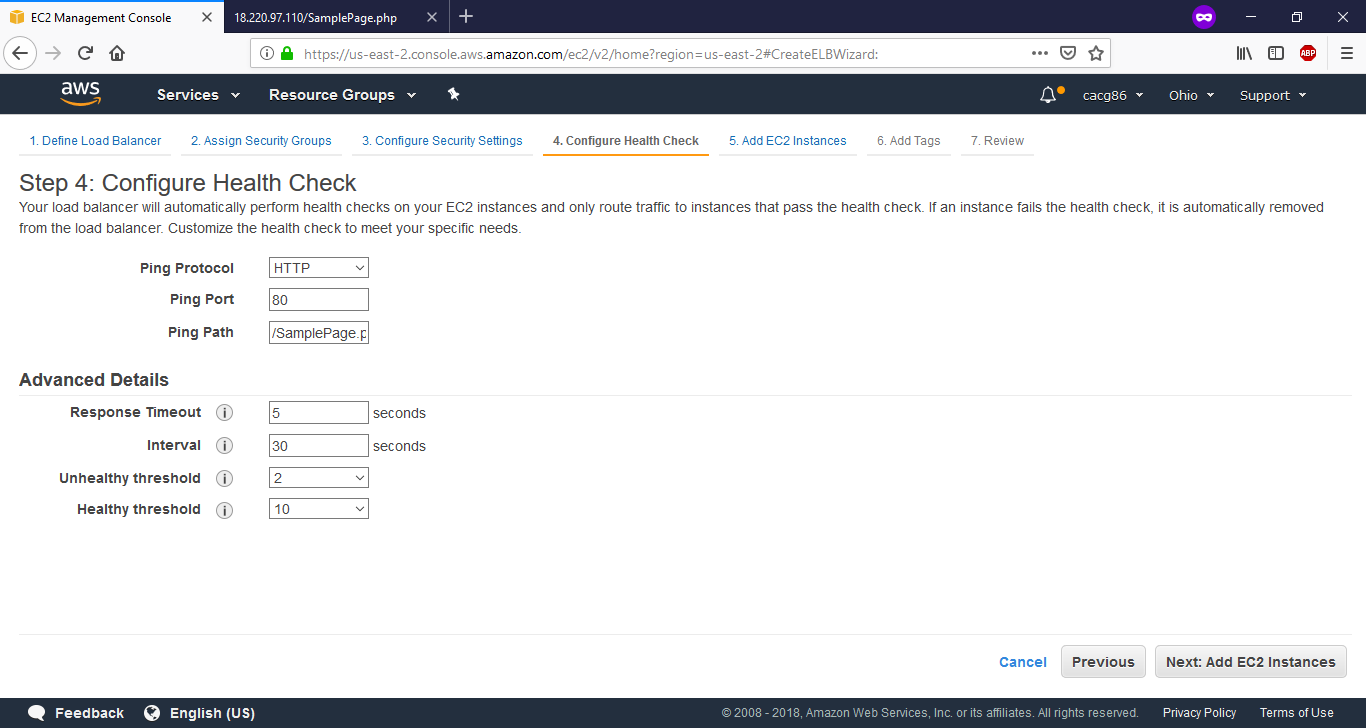
1. Asignar el security group



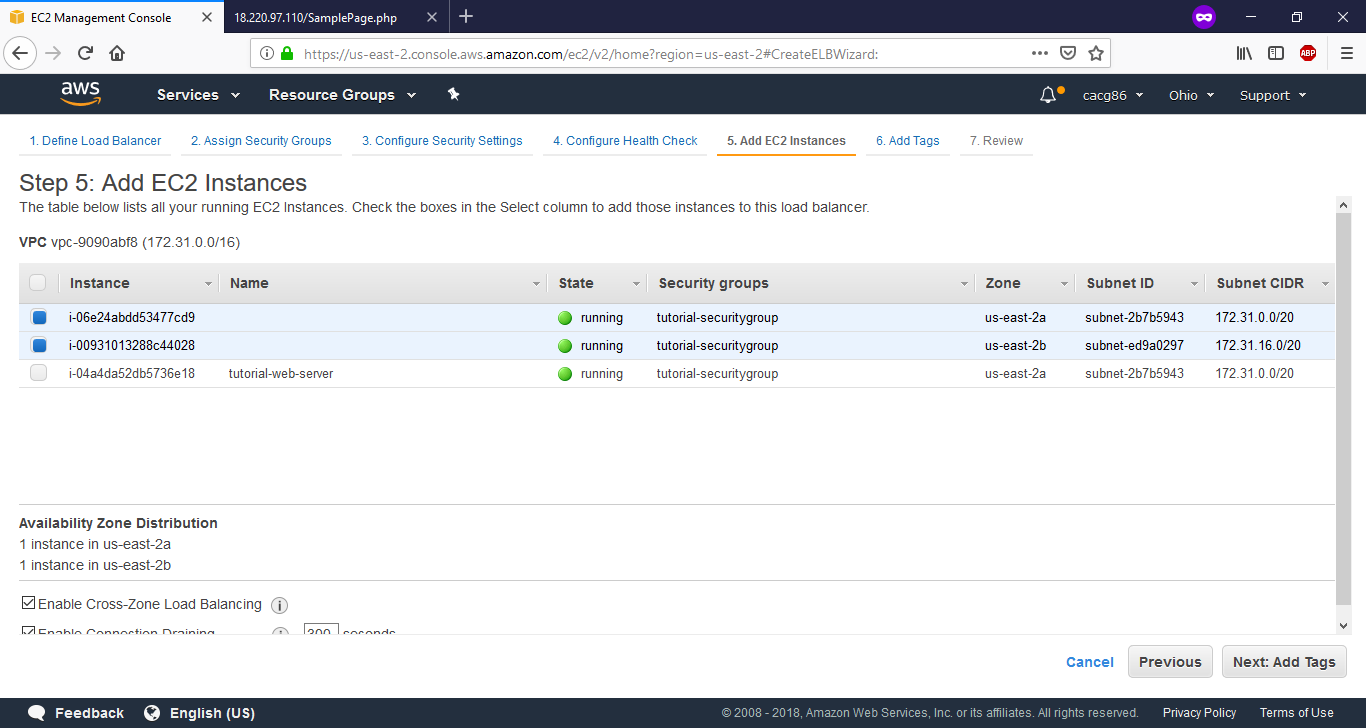
1. Dar siguiente



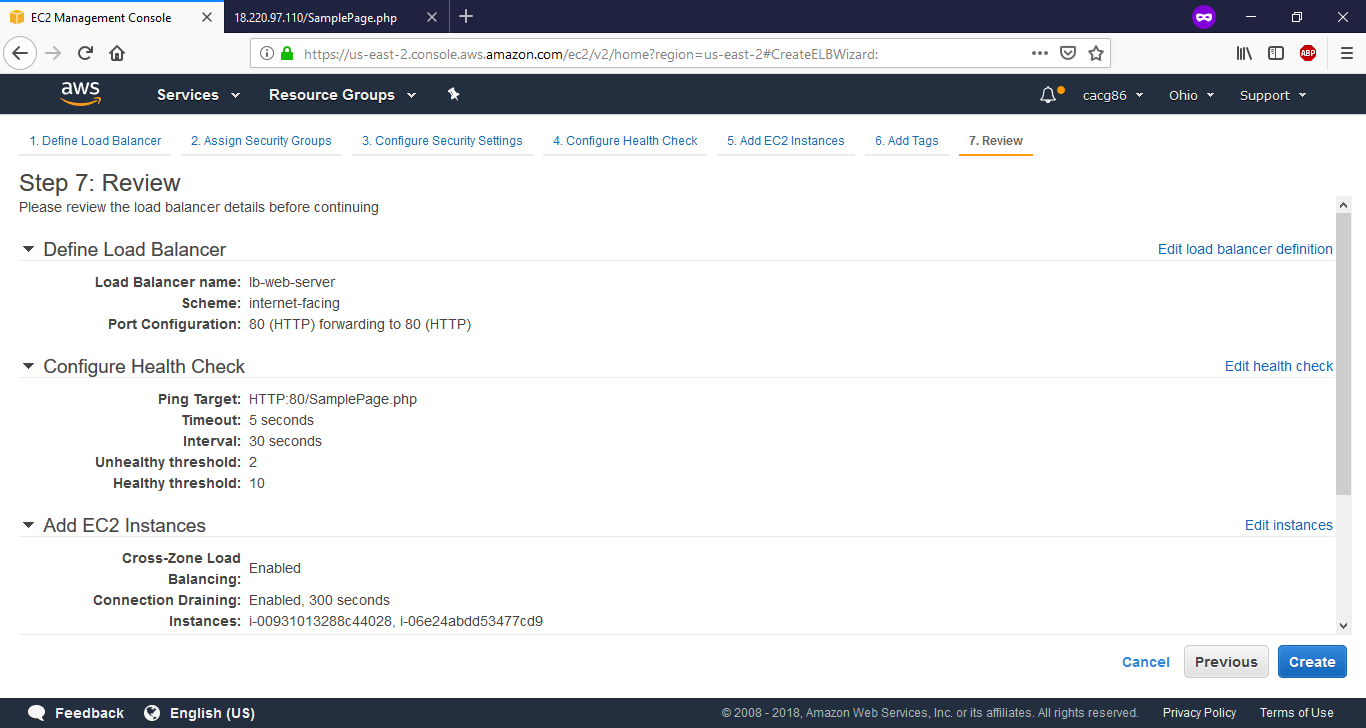
1. En health check poner en ping path “SamplePage.php”



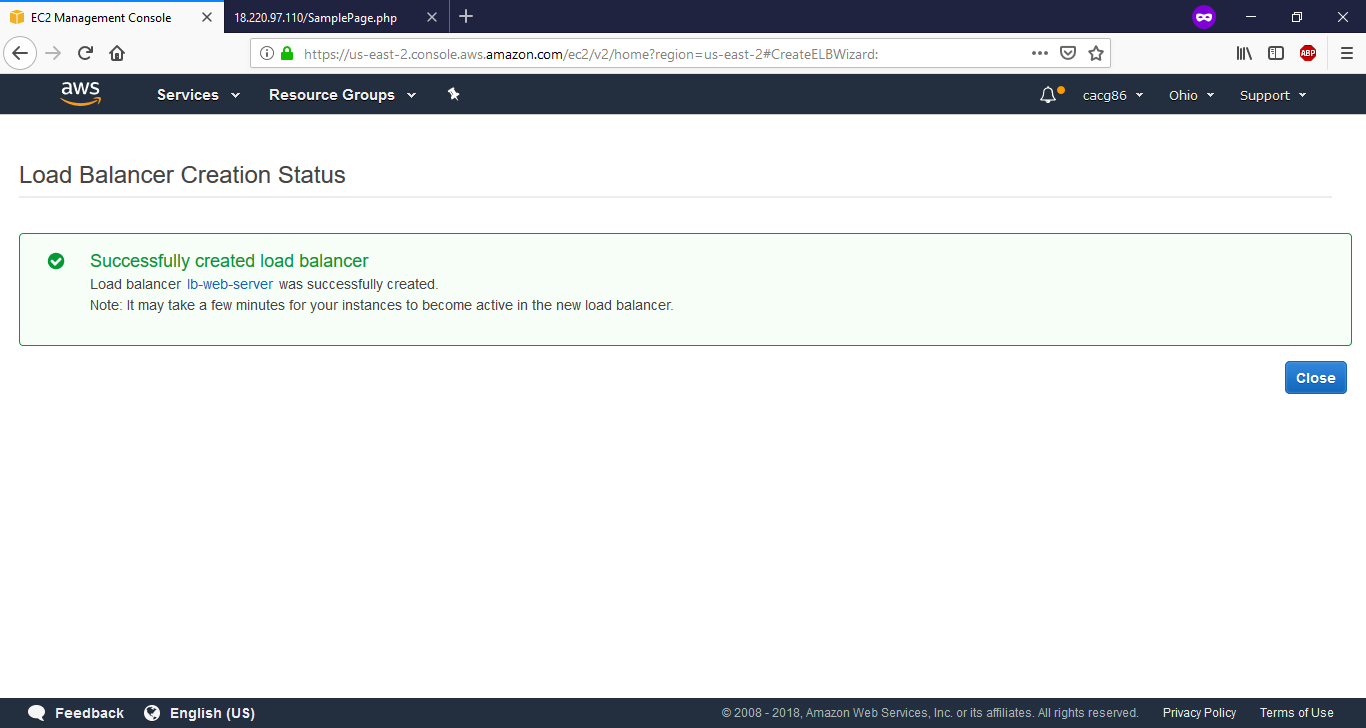
1. Agregar la VM creada por el autoscaling group

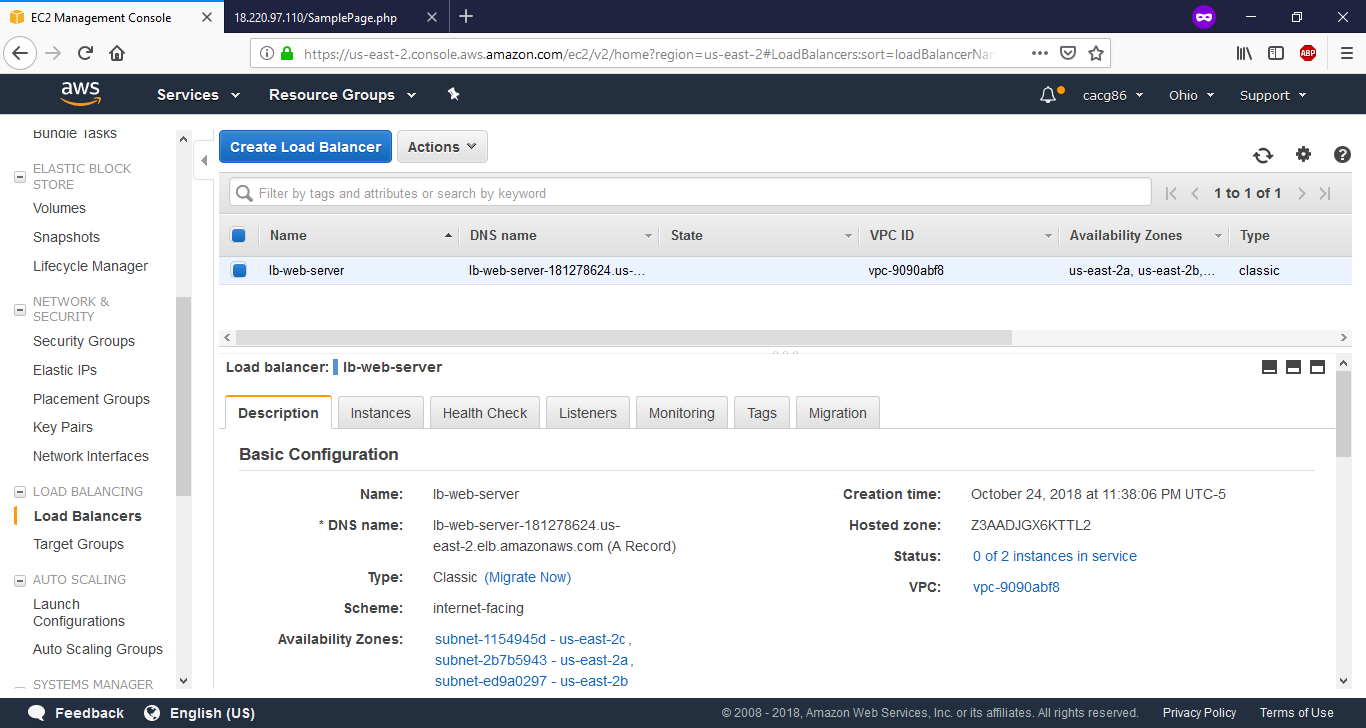


1. Dar click a “Create”



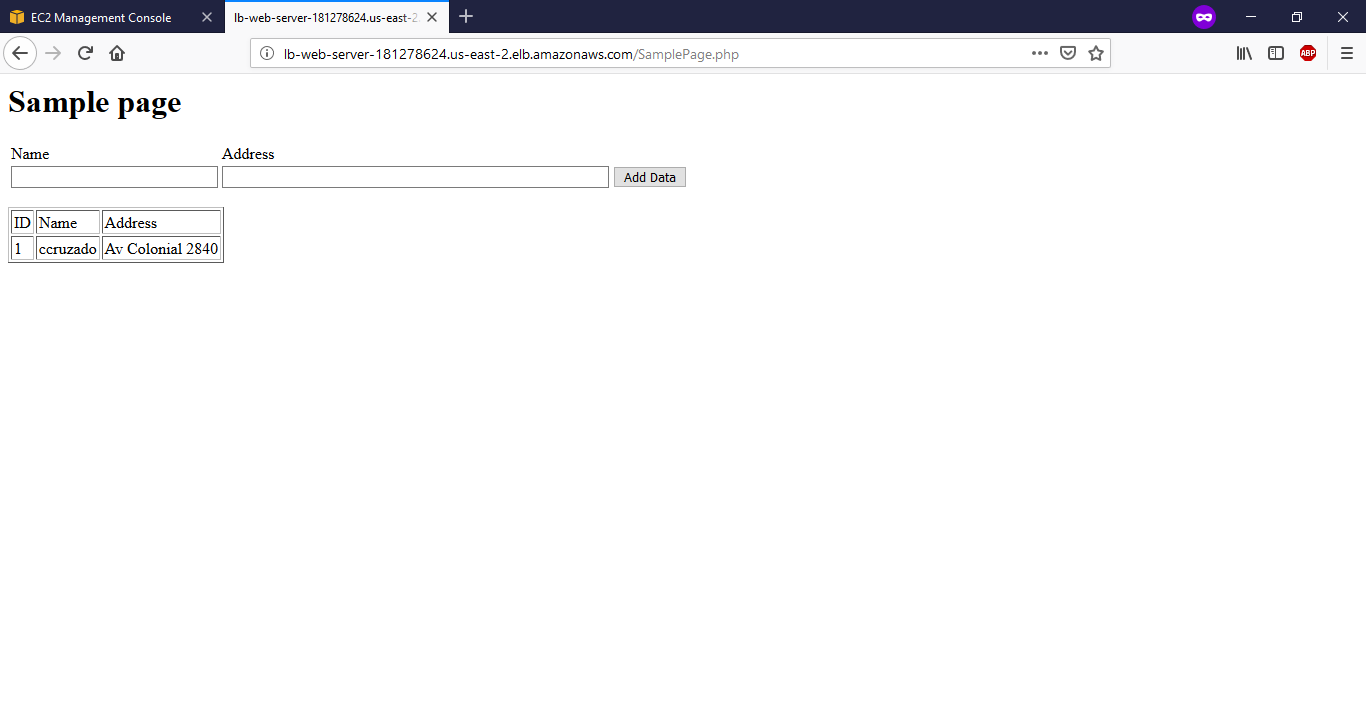
1. Verificar creación de ELB



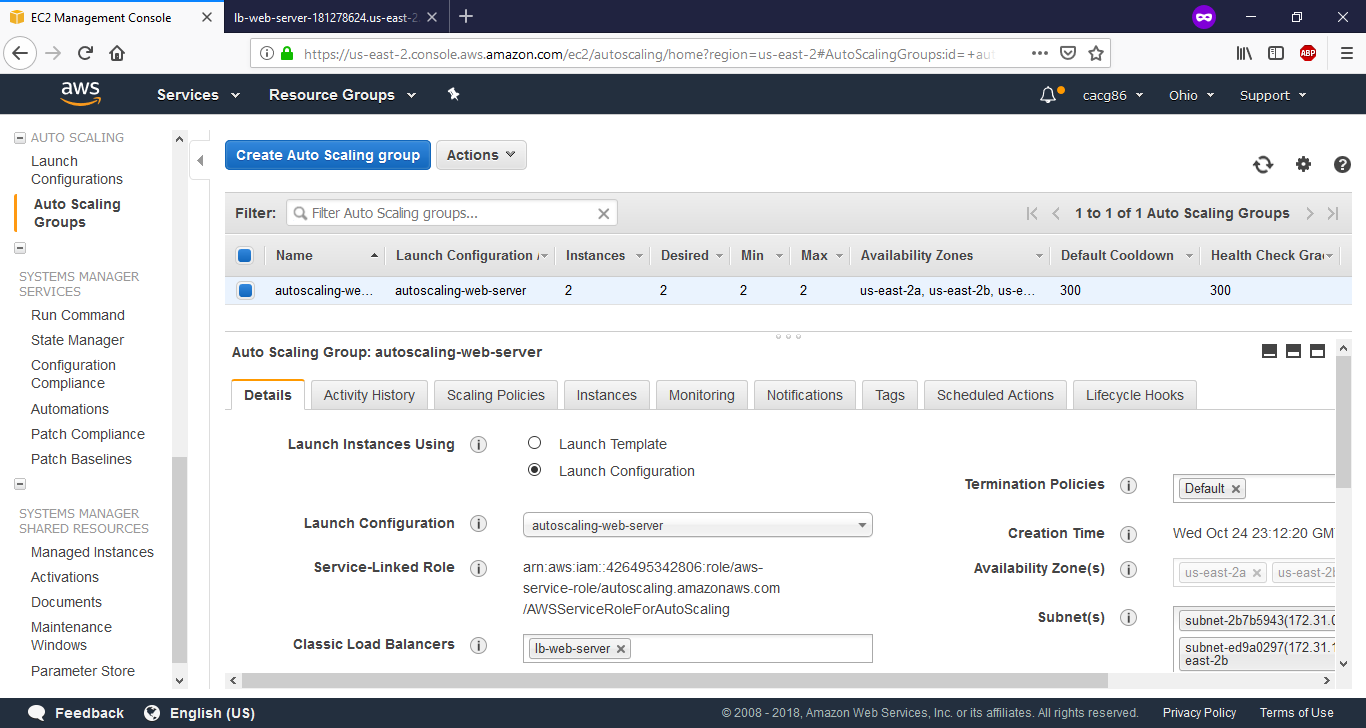


1. Copiar DNS de load balancer e ingresar desde un browser:

<http://lb-web-server-181278624.us-east-2.elb.amazonaws.com/SamplePage.php>



1. Asociar el ELB al autoscaling group



1. Modificando la cantidad de instancias, podemos ver como estas se crean y son registradas en el balanceador de carga

