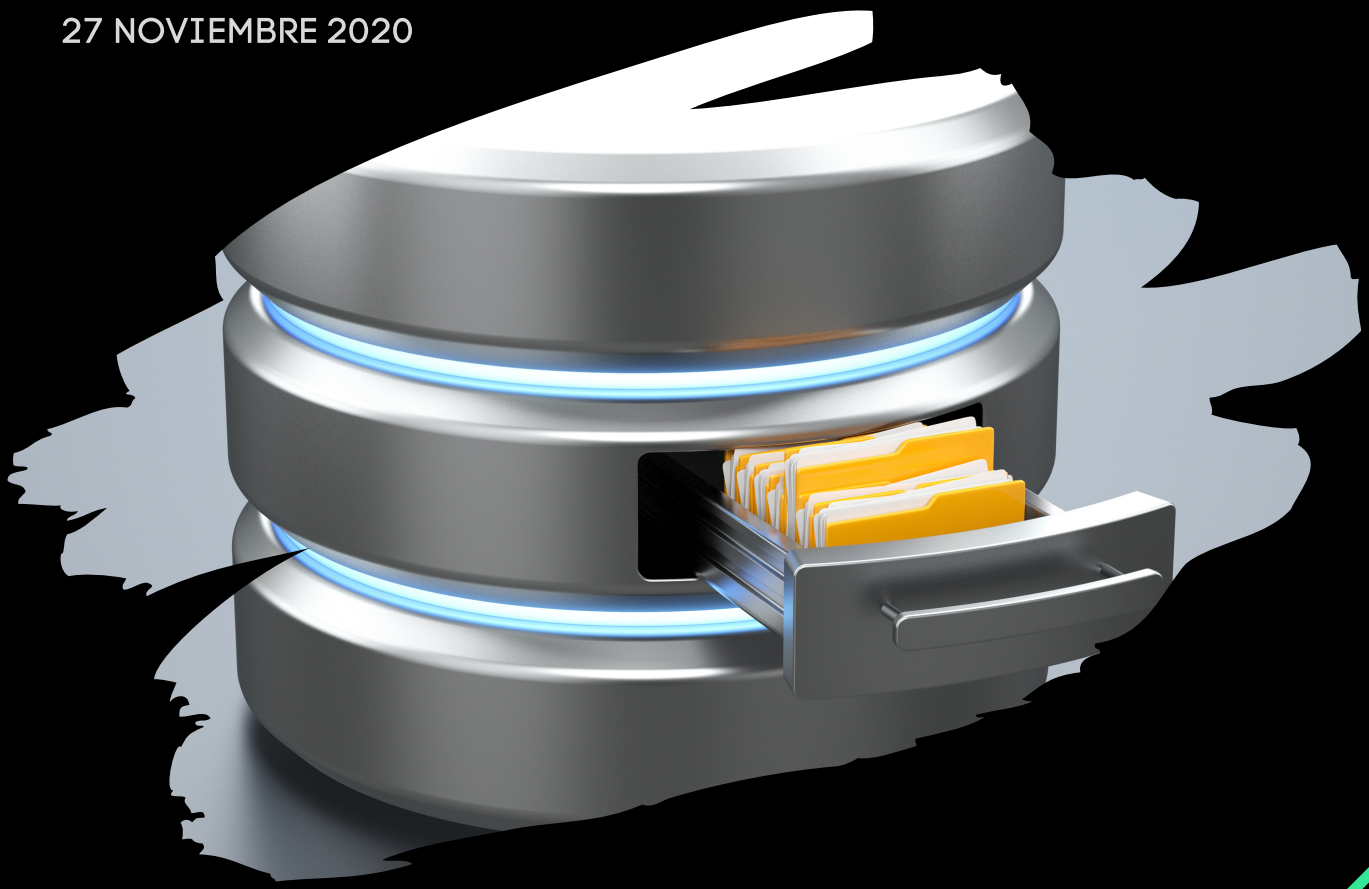
27 NOVIEMBRE 2020



AWS-UBUNTU

Hanna Siddharttha Lizarraga Ceballos

Objetivo

Se tiene como objetivo aprender a usar las instancias en amazon LightSail utilizando OS de Ubuntu y que en este por medio de docker podamos crear o importar base de datos.

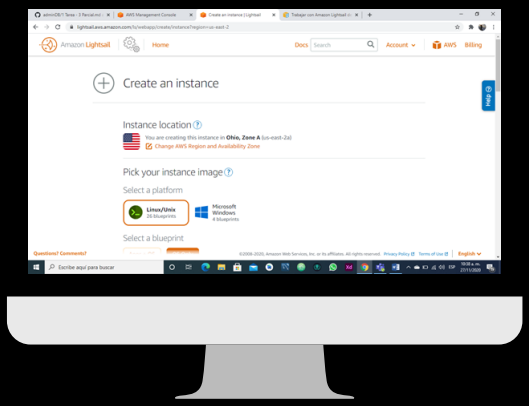
Alcance

En este caso al instalar Ubuntu e integrar docker a este, se hará un clon de repositorio de git hub para acceder a nuestra base de datos y crear tablas y contenido en esta.

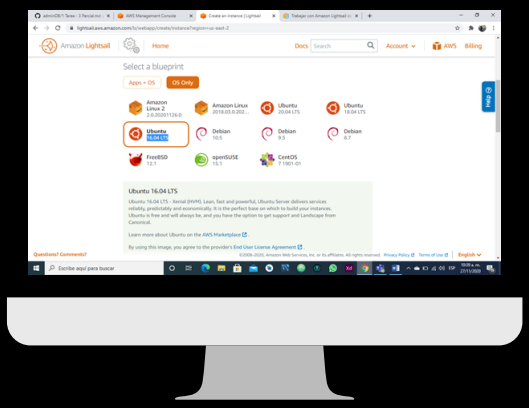
Como crear una instancia en Amazon LightSail

Primero ingresamos a la pagina de aws.amazon.com, seguido de esto iniciamos sesión para consola y seleccionamos Usuario raíz.

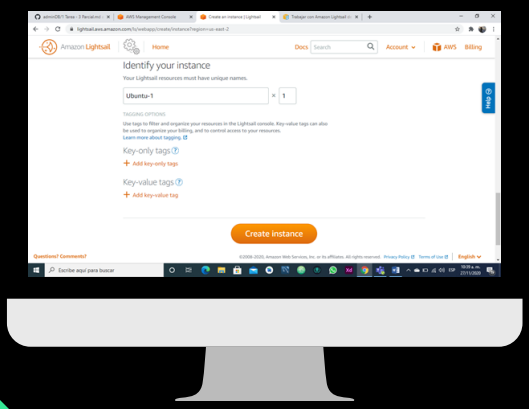
Una vez hecho esto damos click en Crear Instancia.



Seleccionamos Linux

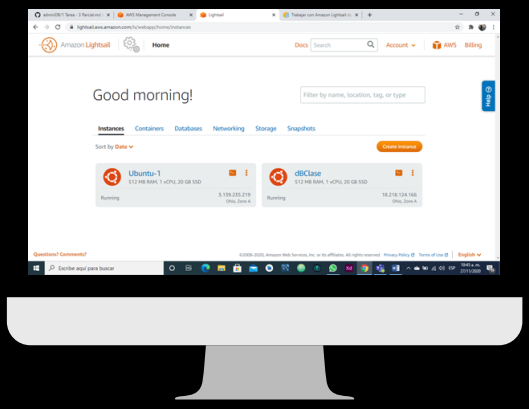


Seleccionamos la opción de OS y escogemos Ubuntu

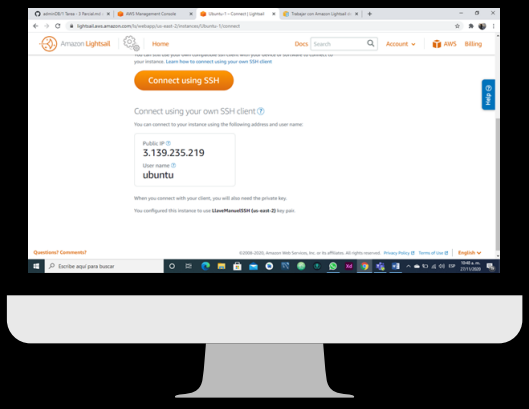


Creamos la instancia

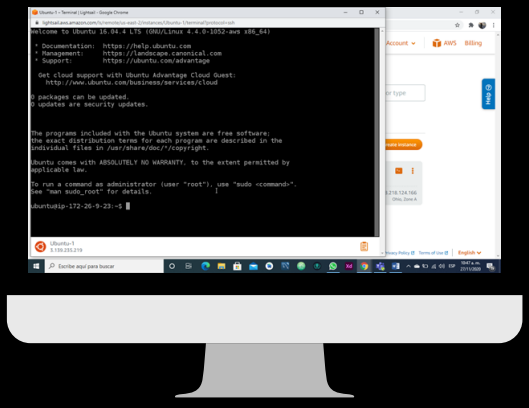
Como crear una instancia en Amazon LightSail



Al crear la instancia se vera así

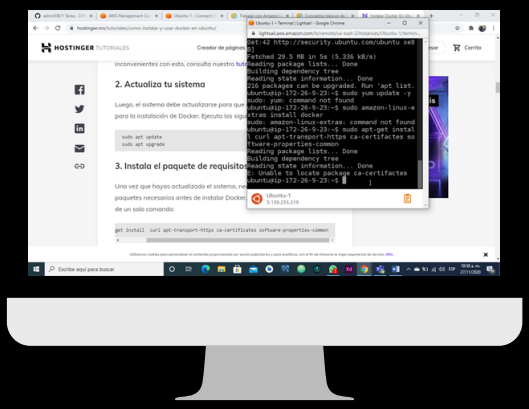


Al ingresar podemos observar nuestro IP el cual utilizaremos mas adelante.



Nos conectamos por SSH y actualizamos el sistema.

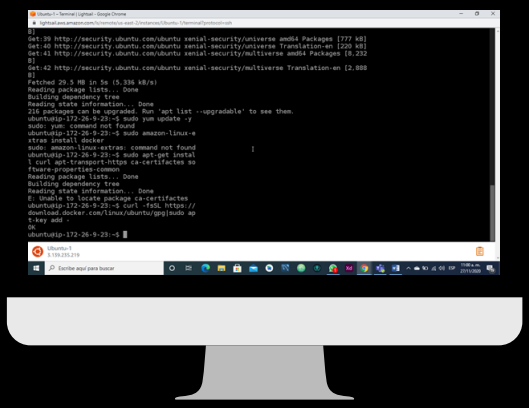
Instalando Docker en Ubuntu



Instalamos el paquete de requisitos previos

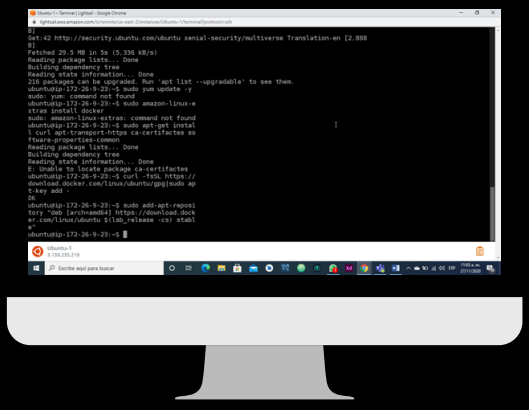
sudo apt-get install curl apt-transport-https ca-certificates software-properties-common

- apt-transport-https: permite que el administrador de paquetes transfiera datos a través de https
- ca-certificates: permite que el navegador web y el sistema verifiquen los certificados de seguridad
- curl: transfiere datos
- software-properties-common: agrega scripts para administrar el software



Agregamos los repositorios de Docker

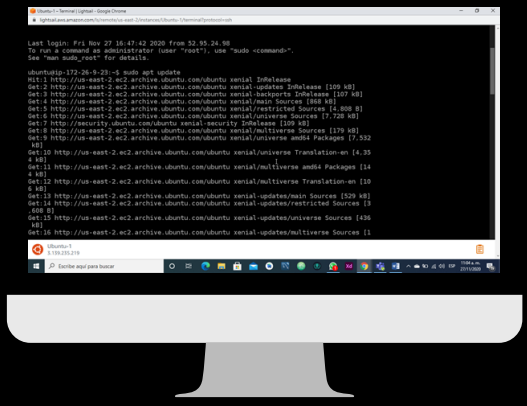
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -



Agregamos el repositorio

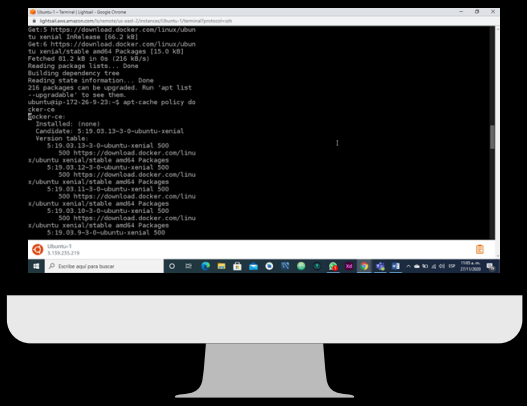
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"

Instalando Docker en Ubuntu



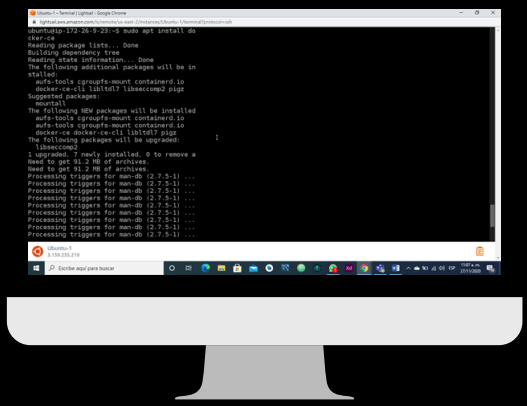
Actualizamos la información del repositorio

sudo apt update



Para saber si estamos instalando desde el repositorio de Docker en lugar del repositorio predeterminado de Ubuntu.

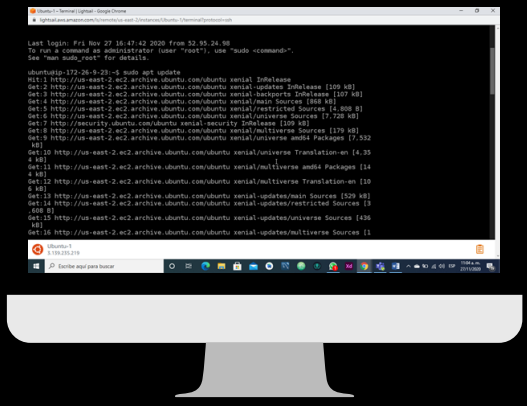
apt-cache policy docker-ce



Instalando Docker en Ubuntu 18.04

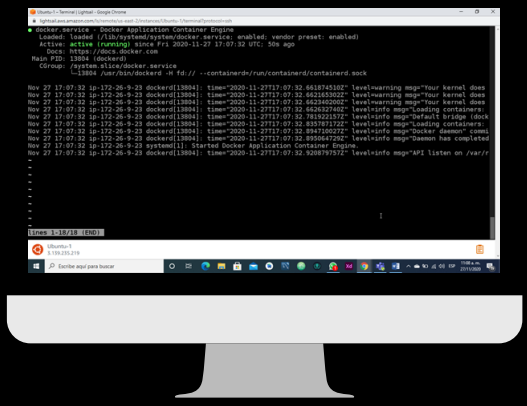
sudo apt install docker-ce

Instalando Docker en Ubuntu



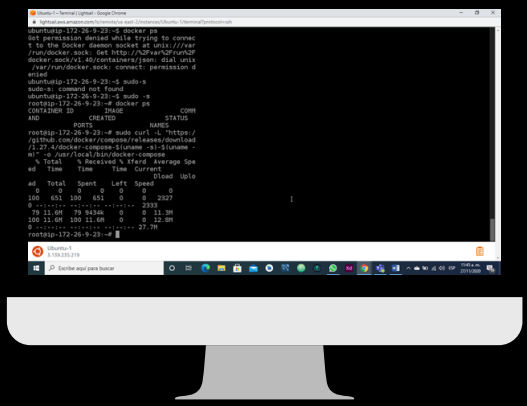
Actualizamos la información del repositorio

sudo apt update



Comprueba el estado de Docker.

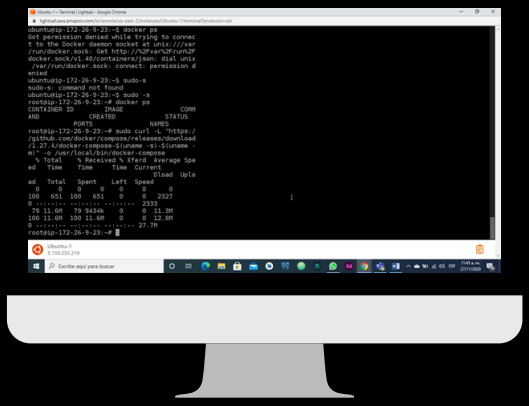
sudo systemctl status docker



Entramos como super usuario

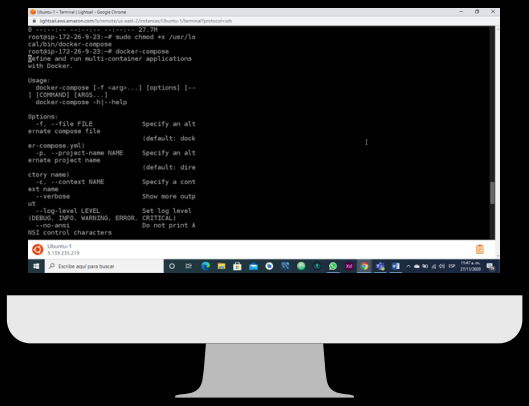
sudo -s

Instalar Compose en sistemas Linux



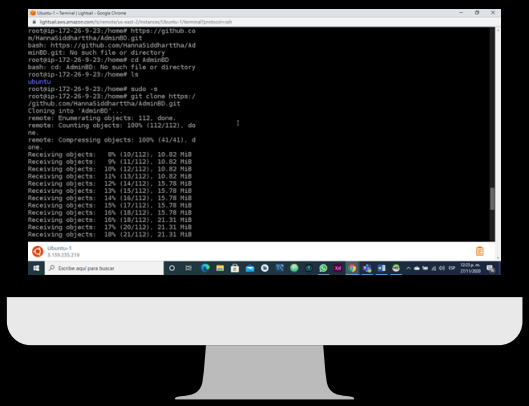
Descargar la versión estable actual de Docker Compose

sudo curl -L "https://github.com/docker/compose/releases/download/1.27.4/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose



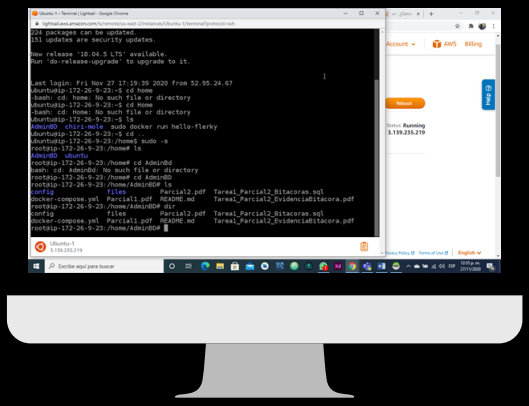
Aplicar permisos ejecutables al binario.

sudo chmod +x /usr/local/bin/docker-compose

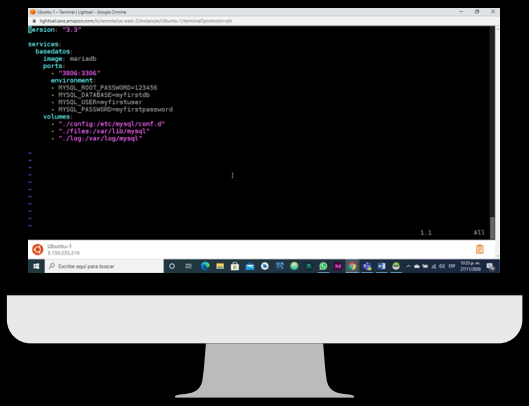


Clonamos nuestro repo.

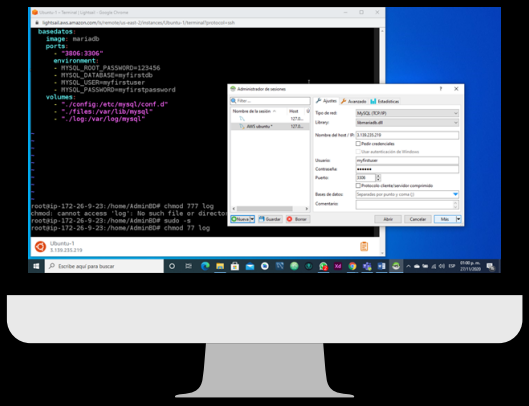
Accediendo a nuestra base de datos.



Entramos a nuestra carpeta en este caso AdminBD

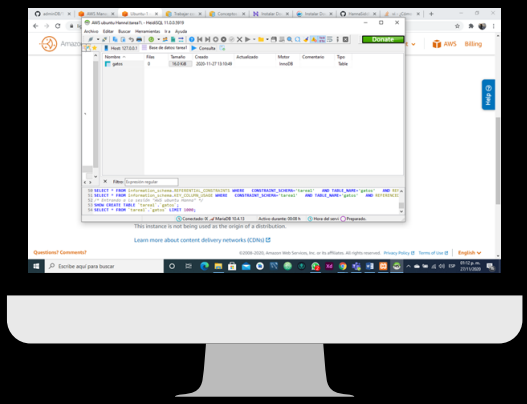


Vemos nuestro archivo de docker-compose.yml para checar sus datos.

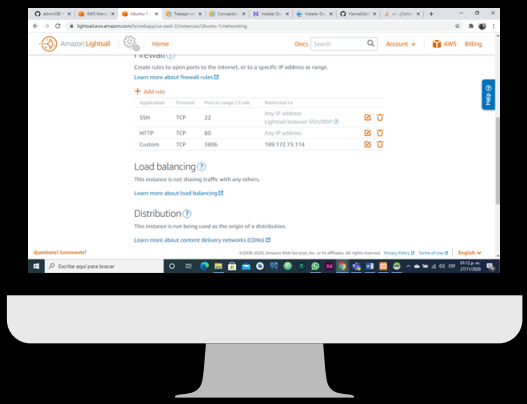


Abrimos nuestra base de datos, a la cual ingresaremos con nuestra IP de la instancia y los datos de nuestro archivo yml.

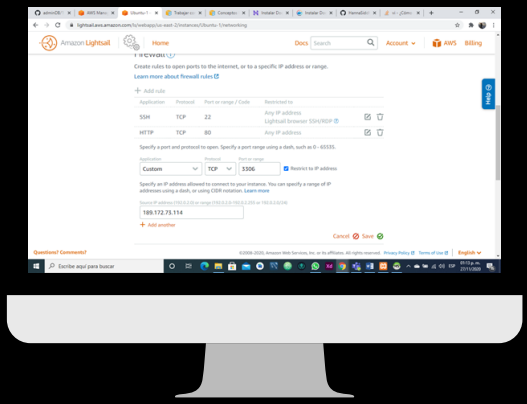
Contenido en nuestra base de datos y reglas y restricciones de firewall.



Al entrar a nuestra base de datos, creamos una tabla.



Después regresamos a nuestro panel y en Networking damos click y creamos una regla en firewall.
También podemos agregar o eliminar por medio de puerto o IP el acceso a nuestra instancia.



Para crear una restricción como bloqueo de acceso podemos utilizar nuestra IP para que solo nosotros podamos ingresar.

Conclusión

Es una práctica muy fácil la cual es indispensable saber ya que facilita mucho el trabajo de los desarrolladores una de las muchas ventajas de Docker es que hacen que la implementación de software sea mucho más eficiente que antes. Gracias a esto, los desarrolladores no tendrán problemas para saber cómo se ejecutará su aplicación fuera del entorno de prueba.