



Tecnológico de Estudios Superiores de Ecatepec

**División de Ingeniería en Sistemas
Computacionales**

Academia en Ciencias de la Ingeniería

Materia Programación de Base de Datos

Grupo 5701

Alumno

Campero Granados Luis Daniel

Profesor

Zapiain Cruz Cesar

Prácticas 3, 4, 5, 6, 7, 8 y 10

Practica 3 - Crear y configurar máquina virtual Linux Ubuntu 18 con Docker.

Una vez dentro del portal de azure creamos una nueva *Maquina Virtual*.

The image shows two screenshots of the Microsoft Azure portal interface. The top screenshot displays the main Azure dashboard with various service icons like 'Máquinas virtuales', 'App Services', and 'Cuentas de almacenamiento'. The bottom screenshot shows the 'Virtual machines' blade under the 'Compute' category. It includes filters for 'Subscription', 'Resource group', and 'Location', and a table header for columns like 'Nombre', 'Suscripción', 'Grupo de recursos', 'Ubicación', 'Estado', 'Sistema operativo', 'Tamaño', 'Dirección IP pública', and 'Discos'. A message at the bottom states 'No hay máquinas virtuales para mostrar' (No virtual machines available to show).

Esta maquina virtual le pondremos los siguientes datos:

Suscripción *: Azure para estudiantes
Grupo de recursos *: (Nuevo) Lina5_group
Nombre de máquina virtual *: Linux5
Región *: US West Central US
Opciones de disponibilidad *: No se requiere redundancia de la infraestructura
Tipo de seguridad *: Estándar
Imagen *: Ubuntu Server 18.04 LTS - Gen2
Instancia de Azure de acceso puntual *:
Tamaño *: Standard_B2s - 2 vCPU, 4 GiB de memoria (M0N 708.61/mes)
Cuenta de administrador
Tipo de autenticación *:
Nombre de usuario *: Luis
Contraseña *: masked
Confirmar contraseña *: masked

Reglas de puerto de entrada

Revisar y crear < Anterior Siguiente: Discos >

Validación superada

Datos básicos Discos Redes Administración Opciones avanzadas Etiquetas Revisar y crear

PRODUCT DETAILS
Standard_B2s by Microsoft Subscription credits apply 1,652 MN/hr Pricing for other VM sizes
[Terms of use](#) | [Privacy policy](#)

TERMS
By clicking "Create", (a) agrees to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorizes Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agrees that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third party offerings. See the Azure Marketplace terms for additional details.

Ha establecido los siguientes puertos abiertos para Internet: SSH. Esto solo se recomienda para las pruebas. Si quiere cambiar esta configuración, vuelve a la pestaña de aspectos básicos.

Datos básicos
Suscripción: Azure para estudiantes
Grupo de recursos: (Nuevo) Lina5_group
Nombre de máquina virtual: Linux5
Región: West Central US
Opciones de disponibilidad: No se requiere redundancia de la infraestructura
Tipo de seguridad: Estándar
Imagen: Ubuntu Server 18.04 LTS - Gen2

Crear < Anterior Siguiente > Descargar una plantilla para la automatización

Crear máquina virtual Canonical Ubuntu Server 18.04 LTS gen2-20220129182609 | Información general

Nombre de implementación: CreateVm-CanonicalUbuntuServer-18...
Suscripción: Azure para estudiantes
Grupo de recursos: Linux5_group

Detalles de la implementación (Descargar)

Recurso	Tipo	Estado	Detalles de la operación
No hay ningún resultado.			

La implementación está en curso

Nombre de implementación: CreateVm-CanonicalUbuntuServer-18...
Hora de inicio: 29/1/2022 18:29:19
Id. de correlación: 9302017d-4f63-4d73-9408-44b1a5b29b7d

Detalles de la implementación (Descargar)

Recurso	Tipo	Estado	Detalles de la operación
Linux5	Microsoft.Compute/virtualMachines	Created	Detalles de la operación
linux5_140	Microsoft.Network/networkInterfaces	Created	Detalles de la operación
Linux5-nsg	Microsoft.Network/networkSecurityGroups	OK	Detalles de la operación
Linux5-ip	Microsoft.Network/publicIPAddresses	OK	Detalles de la operación
Linux5_group-vnet	Microsoft.Network/virtualNetworks	OK	Detalles de la operación

Entramos al tutorial de *DigitalOcean* de como instalar Docker en *Ubuntu 18.*

The screenshot shows a web browser window with the URL www.digitalocean.com/community/tutorials/como-instalar-y-usar-docker-en-ubuntu-18-04-1-es. The page title is "Cómo instalar y usar Docker en Ubuntu 18.04". On the left, there's a sidebar with "CONTENTS" showing steps from "Paso 1 — Instalar Docker" to "Paso 8 — Hacer push de imágenes de Docker a un repositorio Docker". The main content area has a "TUTORIAL" section with a bio for "By Brian Hogan" published on April 26, 2019. It includes sections like "Introducción" and "Requisitos previos". A "RELATED" sidebar on the right offers links to other Docker tutorials and a "Join the DigitalOcean Community" button. At the bottom, there's a cookie consent banner.

Empezamos escribiendo los comandos que tendremos en este tutorial. Empezando por el comando `ssh IPDELAMAQUINAVIRTUAL -i NOMBREDELUSUARIO`.

The screenshot shows a Microsoft Azure Cloud Shell session titled "Linux5 - Máquina virtual". The terminal window displays the command `ssh 52.161.90.253 -i luis` being run. The output shows a warning about host key fingerprint mismatch and a permanent addition to known hosts. It then prompts for a password, which is entered as "luis". The session then lists system information and ends with a copyright notice. The Azure portal interface is visible in the background.

```
luis@52.161.90.253: ~ luis
The authenticity of host '52.161.90.253 (52.161.90.253)' can't be established.
ECDSA key fingerprint is SHA256:npADYXMoOrc2fKxNpluVxh1rbzWXRze8yldSPG0A.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added "52.161.90.253" (ECDSA) to the list of known hosts.
luis@52.161.90.253's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1067-azure x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

System information as of Sun Jan 30 00:36:07 UTC 2022

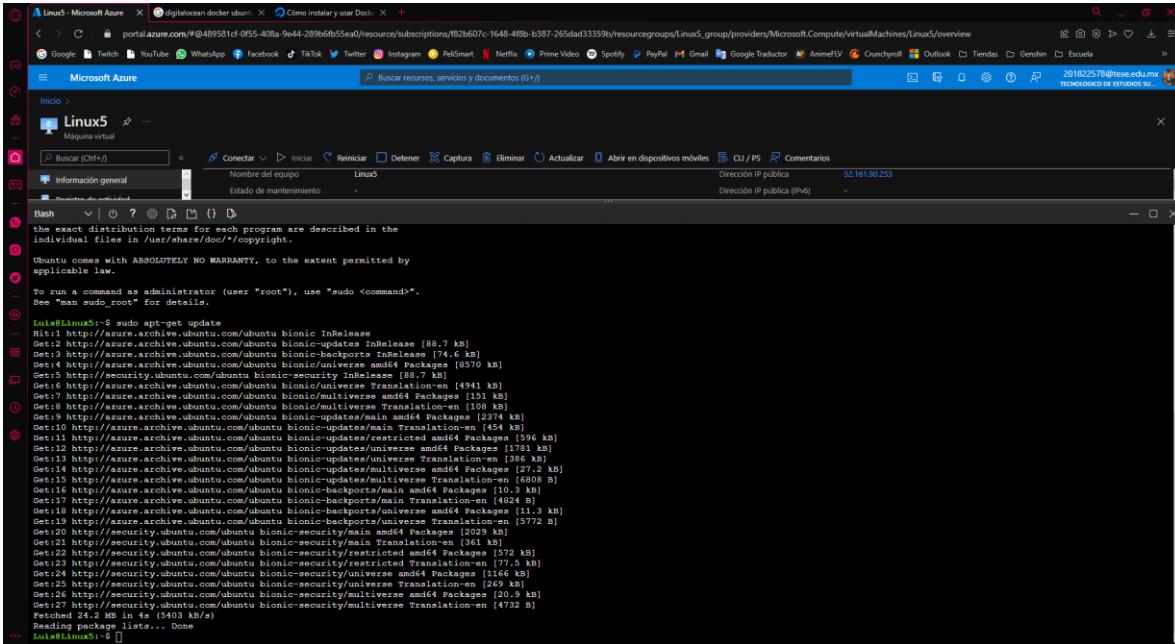
System load: 0.02      Processes:           118
Usage of /: 4.6% of 28.90GB   Users logged in: 0
Memory usage: 5%          IP address for eth0: 10.0.0.4
Swap usage: 0%           

0 updates can be applied immediately.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright*.

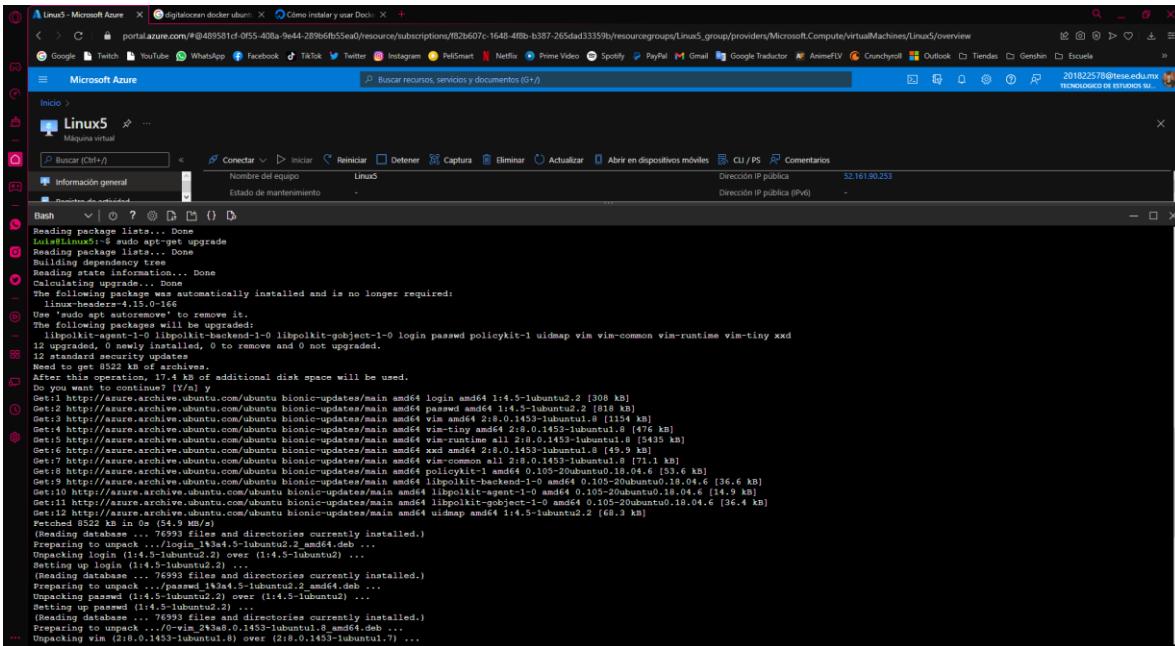
*** Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
```

Seguimos con el comando *sudo apt-get update*.



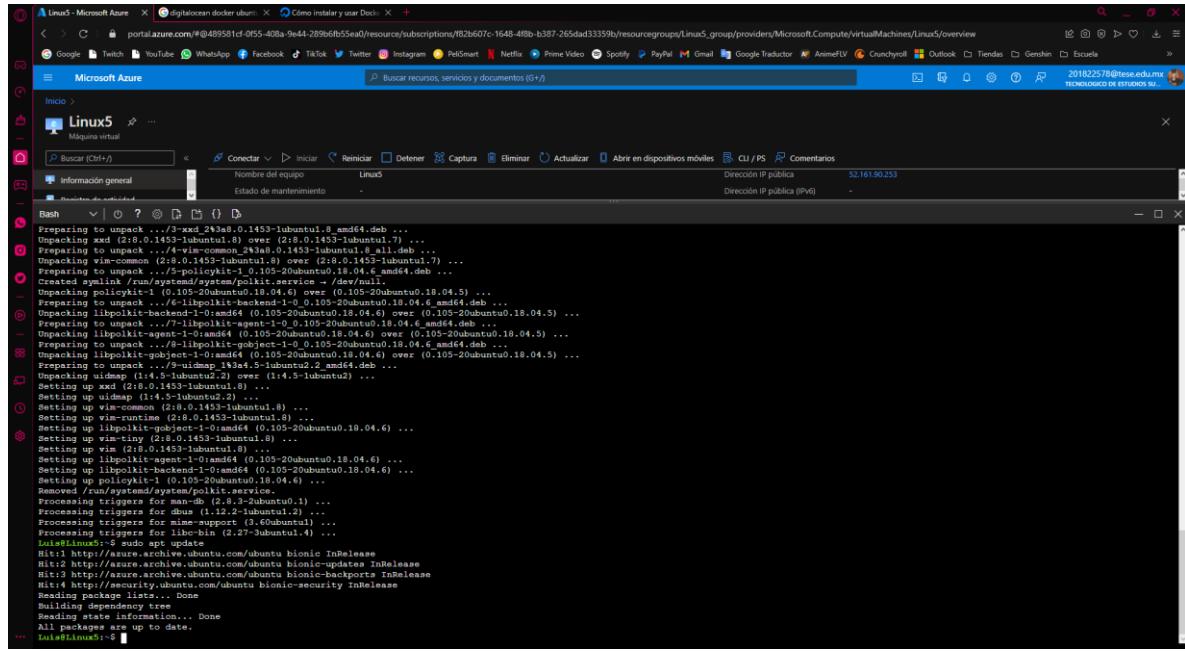
```
Luis@Linux5:~$ sudo apt-get update
Hit:1 http://archive.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://archive.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://archive.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:4 http://archive.archive.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:5 http://security.archive.ubuntu.com/ubuntu bionic/universe Translation-en [4941 kB]
Get:6 http://archive.archive.ubuntu.com/ubuntu bionic/universe Translation-en [108 kB]
Get:7 http://archive.archive.ubuntu.com/ubuntu bionic/universe Translation-en [108 kB]
Get:8 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [2374 kB]
Get:9 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main Translation-en [454 kB]
Get:10 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main Translation-en [454 kB]
Get:11 http://archive.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1596 kB]
Get:12 http://archive.archive.ubuntu.com/ubuntu bionic-updates/universe Translation-en [386 kB]
Get:13 http://archive.archive.ubuntu.com/ubuntu bionic-updates/universe Translation-en [27.2 kB]
Get:14 http://archive.archive.ubuntu.com/ubuntu bionic-updates/universe Translation-en [27.2 kB]
Get:15 http://archive.archive.ubuntu.com/ubuntu bionic-backports/main amd64 Packages [10.3 kB]
Get:16 http://archive.archive.ubuntu.com/ubuntu bionic-backports/main Translation-en [4824 B]
Get:17 http://archive.archive.ubuntu.com/ubuntu bionic-backports/main Translation-en [11.3 kB]
Get:18 http://archive.archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [11.3 kB]
Get:19 http://archive.archive.ubuntu.com/ubuntu bionic-backports/universe Translation-en [5772 B]
Get:20 http://security.archive.ubuntu.com/ubuntu bionic-security/main amd64 Packages [2029 kB]
Get:21 http://security.archive.ubuntu.com/ubuntu bionic-security/main Translation-en [361 kB]
Get:22 http://security.archive.ubuntu.com/ubuntu bionic-security/restricted amd64 Packages [22.8 kB]
Get:23 http://security.archive.ubuntu.com/ubuntu bionic-security/universe Translation-en [177.5 kB]
Get:24 http://security.archive.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [1166 kB]
Get:25 http://security.archive.ubuntu.com/ubuntu bionic-security/universe Translation-en [269 kB]
Get:26 http://security.archive.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [20.9 kB]
Get:27 http://security.archive.ubuntu.com/ubuntu bionic-security/universe Translation-en [4792 B]
Fetched 24.2 MB in 4s (5403 kB/s)
Reading package lists... Done
Luis@Linux5:~$
```

Posteriormente el comando *sudo apt-get upgrade*.



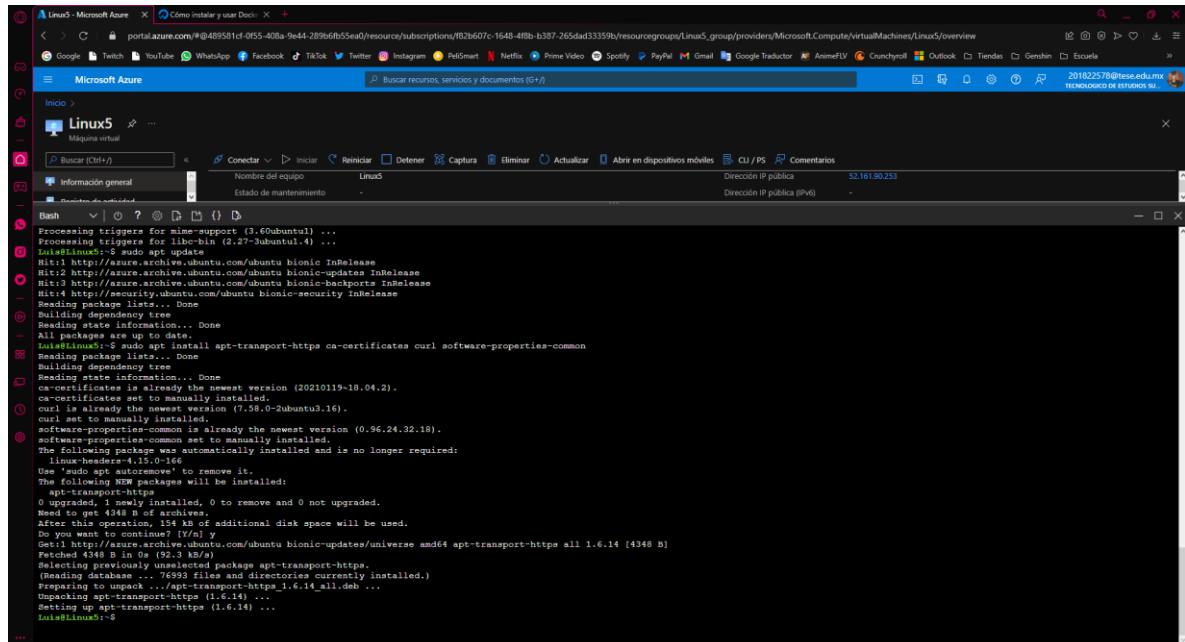
```
Luis@Linux5:~$ sudo apt-get upgrade
[sudo] password for Luis: 
Reading package lists...
Reading package lists... Done
Luis@Linux5:~$ sudo apt-get upgrade
[sudo] password for Luis: 
Building dependency tree
Reading state information...
Calculating upgrade...
Done
The following packages were automatically installed and are no longer required:
  linux-headers-4.15.0-166
Use 'sudo apt autoremove' to remove it.
The following packages will be upgraded:
  libgpg-error0 libgnutls-debian0 libgnutls28-openssl1-0 libkipk-backend1-0 login passwd policykit-1 vim vim-common vim-runtime vim-tiny xxd
12 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
12 standard security updates
12 to upgrade from 12 of another.
After this operation, 4 kB additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 login amd64 1:4.5-lubuntu2.2 [308 kB]
Get:2 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 passwd amd64 1:4.5-lubuntu2.2 [103 kB]
Get:3 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 vte-tiny amd64 2:8.0.1453-lubuntu1.8 [1154 kB]
Get:4 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 vte-runtime all 2:8.0.1453-lubuntu1.8 [476 kB]
Get:5 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 vim amd64 1:1.18.1-1ubuntu1.1 [5435 kB]
Get:6 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 vim-common amd64 1:1.18.1-1ubuntu1.1 [499 kB]
Get:7 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 vim-runtime amd64 1:1.18.1-1ubuntu1.1 [731.1 kB]
Get:8 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 policykit-1 amd64 1:105-20ubuntu18.04.6 [53.6 kB]
Get:9 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 libkipk-backend1-0 amd64 1:105-20ubuntu18.04.6 [36.6 kB]
Get:10 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 libkipk-agent1-0 amd64 1:105-20ubuntu18.04.6 [14.9 kB]
Get:11 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 libkipk-backend1-0 amd64 1:105-20ubuntu18.04.6 [36.4 kB]
Get:12 http://archive.archive.ubuntu.com/ubuntu bionic-updates/main amd64 vimmap amd64 1:4.5-lubuntu2.2 [60.3 kB]
Fetched 8522 kB in 0s (54.9 MB/s)
Preparing to unpack .../login_1:4.5-lubuntu2.2_amd64.deb...
Unpacking login (1:4.5-lubuntu2.2) over (1:4.5-lubuntu2) ...
Setting up login (1:4.5-lubuntu2.2) ...
Preparing to unpack .../passwd_1:4.5-lubuntu2.2_amd64.deb...
Unpacking passwd (1:4.5-lubuntu2.2) ...
Setting up passwd (1:4.5-lubuntu2.2) ...
Preparing to unpack .../vte-tiny_2:8.0.1453-lubuntu1.8_amd64.deb...
Unpacking vte-tiny (2:8.0.1453-lubuntu1.8) over (2:8.0.1453-lubuntu1.7) ...
Setting up vte-tiny (2:8.0.1453-lubuntu1.8) ...
```

Ahora si seguimos con los comandos del *DigitalOcean* empezando por el comando *sudo apt update*.



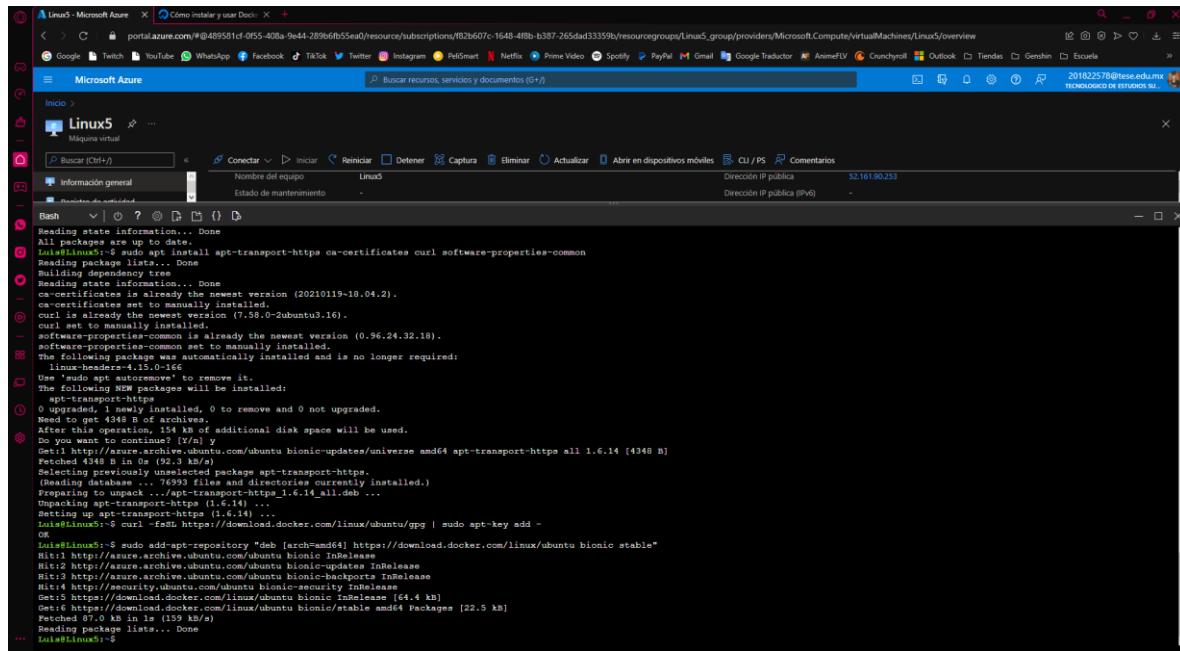
```
Luis@Linux5:~$ sudo apt update
Hit:1 http://aur.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://archive.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://archive.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
...
Luis@Linux5:~$
```

Después seguimos con *sudo apt install apt-transport-https ca-certificates curl software-properties-common*.



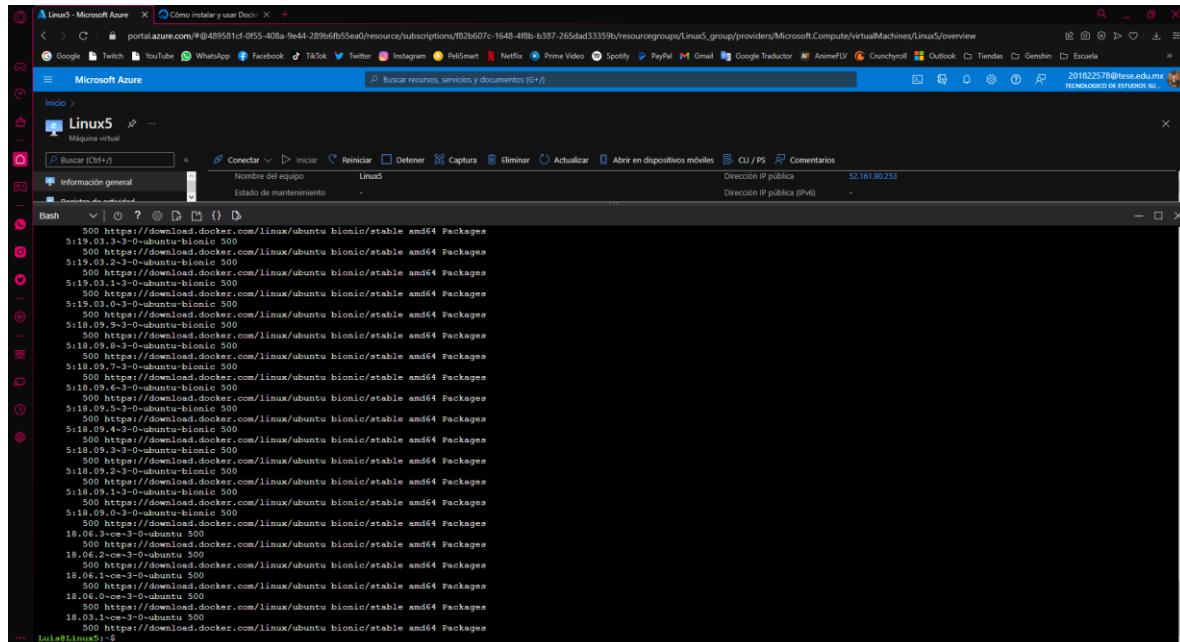
```
Luis@Linux5:~$ sudo apt install apt-transport-https ca-certificates curl software-properties-common
Processing triggers for mime-support (3.60ubuntu0.1) ...
Processing triggers for libhmac-sha512-perl (2.03-1) ...
Processing triggers for libhmac-sha512-perl (2.03-1) ...
Luis@Linux5:~$ sudo apt update
Hit:1 http://aur.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://archive.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://archive.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
...
Luis@Linux5:~$ sudo apt install apt-transport-https ca-certificates curl software-properties-common
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version (20210119-10.04.2).
curl is already the newest version (7.58.0-0ubuntu3.16).
curl set to manually installed.
software-properties-common is already the newest version (0.96.24.32.18).
The following package was automatically installed and is no longer required:
linux-headers-4.15.0-166
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 4348 B of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://aur.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 apt-transport-https all 1.6.14 [4348 B]
Fetched 4348 B in 0s (92.3 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 76993 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_1.6.14_all.deb ...
Unpacking apt-transport-https ...
Setting up apt-transport-https (1.6.14) ...
...
Luis@Linux5:~$
```

Seguimos con `curl -fsSL` <https://download.docker.com/linux/ubuntu/gpg>
| `sudo apt-key add -`



```
Bash | ... | D
Reading state information... Done
All packages are up to date.
0 upgraded, 0 newly installed, 0 to remove or upgrade.
0 not fully installed yet.
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version (20210119-10.04.2).
ca-certificates set to manually installed.
curl is already the newest version (7.58.0-2ubuntu1.16).
curl set to manually installed.
software-properties-common is already the newest version (0.96.24.32.18).
software-properties-common set to manually installed.
The following package was automatically installed and is no longer required:
libcurl4-openssl4:i386
Use 'apt autoremove' to remove it.
The following NEW packages will be installed:
apt-transport-https
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 4348 B of archives.
After this operation, 154 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Gett http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 apt-transport-https all 1.6.14 [4348 B]
Fetched 4348 B in 0s (92.3 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 76993 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_1.6.14_all.deb ...
Unpacking apt-transport-https (1.6.14) ...
Setting up apt-transport-https (1.6.14) ...
Linux5:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
[...]
Linux5:~$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable"
Hit:1 http://archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 https://security.ubuntu.com/ubuntu bionic-security InRelease
Get:5 https://download.docker.com/linux/ubuntu bionic InRelease [64.4 kB]
Get:6 https://download.docker.com/linux/ubuntu bionic/stable amd64 Packages [22.5 kB]
Fetched 159 kB in 1s (159 kB/s)
Reading package lists... Done
Linux5:~$
```

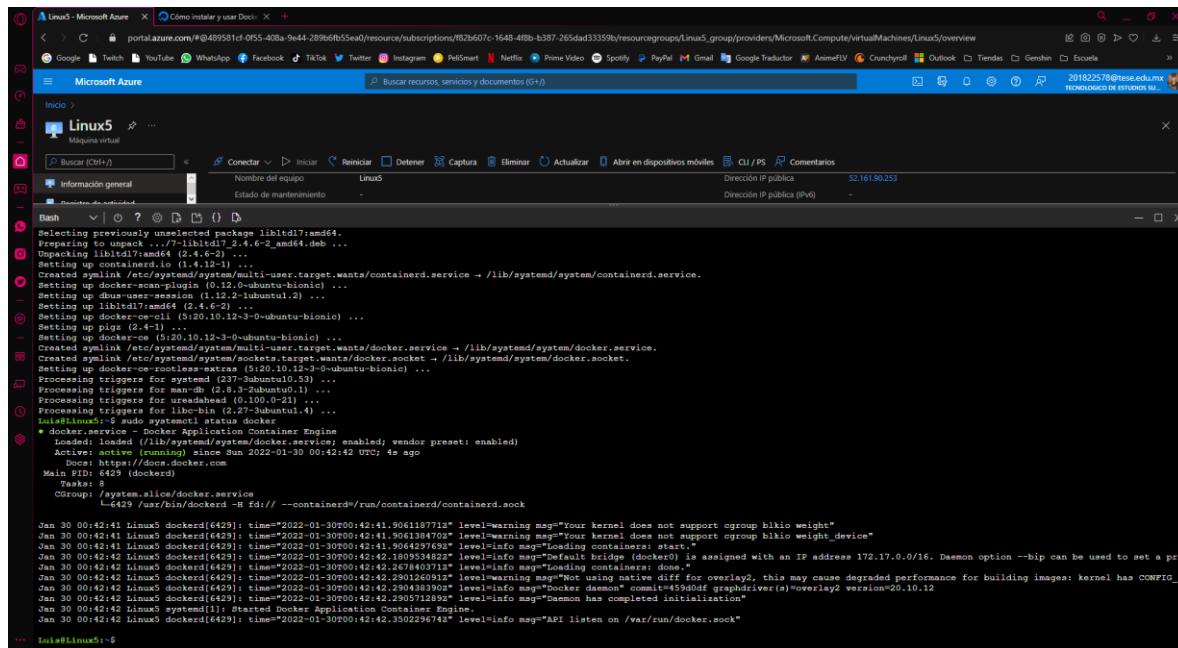
Después seguimos con `sudo add-apt-repository "deb [arch=amd64]`
<https://download.docker.com/linux/ubuntu bionic stable>"



```
Bash | ... | D
5:19.03.3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:19.03.2-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:19.03.1-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:19.03.0-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.5-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.4-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.3-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.7-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.6-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.5-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.4-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.3-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.2-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.1-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
5:18.09.3-3-0ubuntu-bionic 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
18.06.3-ce-3-0ubuntu 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
18.06.1-ce-3-0ubuntu 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
18.06.0-0ubuntu 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
18.03.1-ce-3-0ubuntu 500
500 https://download.docker.com/linux/ubuntu/bionic/stable amd64 Packages
Linux5:~$
```

Después seguirá *sudo apt update* y *sudo apt-cache policy docker-ce* y *sudo apt install docker-ce*

Y para finalizar *sudo systemctl status docker*



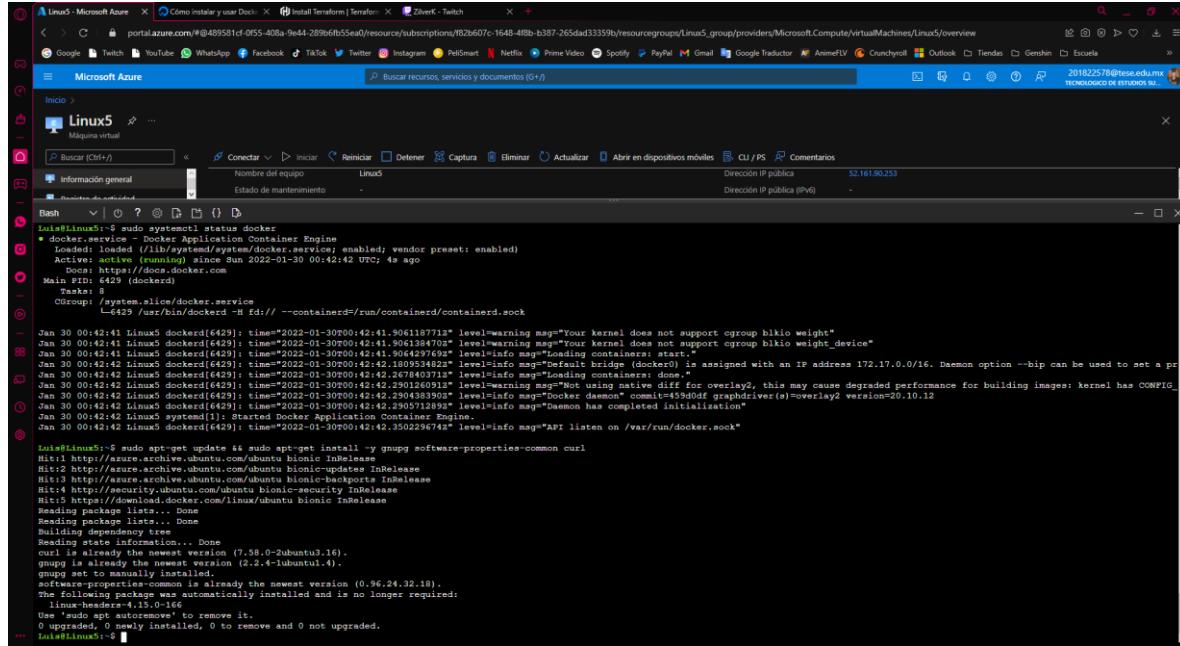
```
Luis@Linux5:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2022-01-30 00:42:52 UTC; 46 ago
     Docs: https://docs.docker.com
Main PID: 6429 (dockerd)
  Tasks: 8
 CGroup: /system.slice/docker.service
         └─ 6429 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Jan 30 00:42:41 Linux5 dockerd[6429]: time="2022-01-30T00:42:41.306118771Z" level=warning msg="your kernel does not support cgroup blkio weight"
Jan 30 00:42:41 Linux5 dockerd[6429]: time="2022-01-30T00:42:41.306138470Z" level=warning msg="your kernel does not support cgroup blkio weight_device"
Jan 30 00:42:41 Linux5 dockerd[6429]: time="2022-01-30T00:42:41.306429769Z" level=info msg="Loading containers: start."
Jan 30 00:42:42 Linux5 dockerd[6429]: time="2022-01-30T00:42:42.180953482Z" level=info msg="Default bridge (docker0) is assigned with an IP address 172.17.0.0/16. Daemon option --bip can be used to set a pr
Jan 30 00:42:42 Linux5 dockerd[6429]: time="2022-01-30T00:42:42.200126991Z" level=warning msg="Running native diff for overlay2, this may cause degraded performance for building images: kernel has CONFIG_
Jan 30 00:42:42 Linux5 dockerd[6429]: time="2022-01-30T00:42:42.290438390Z" level=info msg="Docker daemon" commit=d459dd0 graphdriver(s)=overlay2 version=20.10.12
Jan 30 00:42:42 Linux5 dockerd[6429]: time="2022-01-30T00:42:42.290571289Z" level=info msg="Daemon has completed initialization"
Jan 30 00:42:42 Linux5 systemd[1]: Started Docker Application Container Engine.
Jan 30 00:42:42 Linux5 dockerd[6429]: time="2022-01-30T00:42:42.300229707Z" level=info msg="API listen on /var/run/docker.sock"
...
Luis@Linux5:~$
```

Practica 4 - Configurar Terraform en Linux Ubuntu 18

Para instalar *Terraform* entramos a los tutoriales de *Terraform* y entramos en *Install Terraform*.

En la terminal empezaremos escribiendo los comandos; *sudo apt-get update && sudo apt-get install -y gnupg software-properties-common curl*

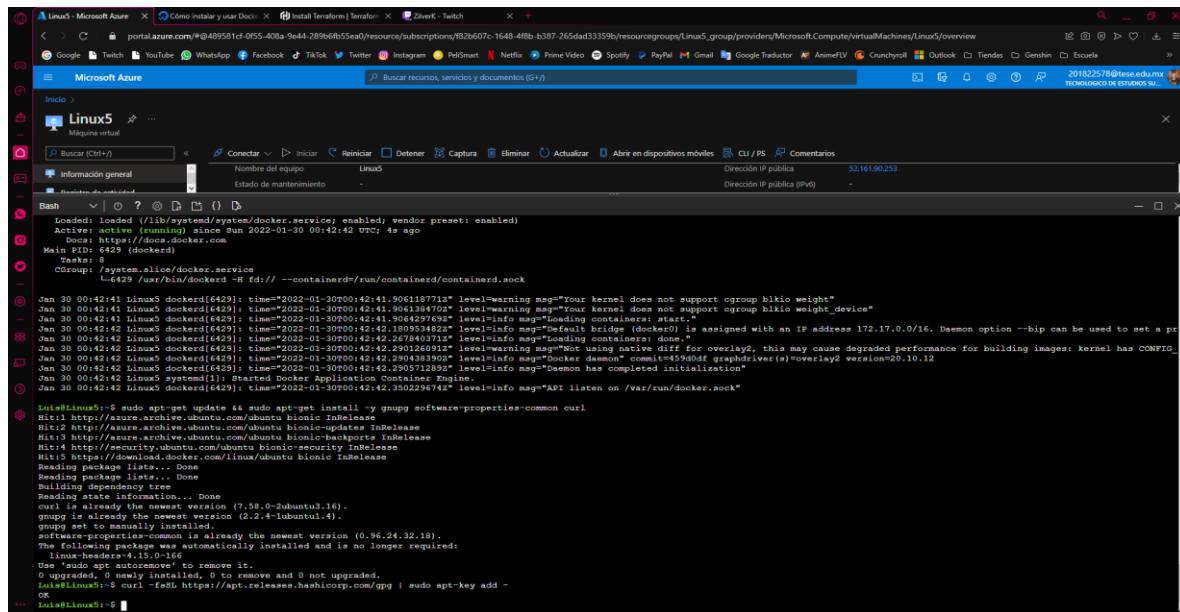


```
Luis@Linux5:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2022-01-30 00:42:42 UTC; 4s ago
     Docs: https://docs.docker.com
Main PID: 6429 (dockerd)
Tasks: 8
CGroup: /system.slice/docker.service
        └─6429 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Jan 30 00:42:41 Linux5 dockerd[6429]: time="2022-01-30T00:42:41.306118718Z" level=warning msg="your kernel does not support cgroup blkio weight"
Jan 30 00:42:41 Linux5 dockerd[6429]: time="2022-01-30T00:42:41.306138704Z" level=warning msg="your kernel does not support cgroup blkio weight_device"
Jan 30 00:42:41 Linux5 dockerd[6429]: time="2022-01-30T00:42:41.306429769Z" level=info msg="Loading containers: start."
Jan 30 00:42:42 Linux5 dockerd[6429]: time="2022-01-30T00:42:42.180953482Z" level=info msg="Default bridge (docker0) is assigned with an IP address 172.17.0.0/16. Daemon option --bip can be used to set a specific IP address on the bridge interface
Jan 30 00:42:42 Linux5 dockerd[6429]: time="2022-01-30T00:42:42.290126913Z" level=warning msg="Not using native diff for overlay2, this may cause degraded performance for building images: kernel has CONFIG_Overlay2_FS=y"
Jan 30 00:42:42 Linux5 dockerd[6429]: time="2022-01-30T00:42:42.290438390Z" level=info msg="Docker daemon" commit=45949dd graphdriver(s)=overlay2 version=20.10.12
Jan 30 00:42:42 Linux5 dockerd[6429]: time="2022-01-30T00:42:42.290471289Z" level=info msg="Docker has completed initialization"
Jan 30 00:42:42 Linux5 systemd[1]: Started Docker Application Container Engine.
Jan 30 00:42:42 Linux5 dockerd[6429]: time="2022-01-30T00:42:42.350229674Z" level=info msg="API listen on /var/run/docker.sock"

Luis@Linux5:~$ sudo apt-get update && sudo apt-get install -y gnupg software-properties-common curl
Hit:1 http://archive.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://archive.archive.ubuntu.com/ubuntu bionic-security InRelease
Hit:3 http://archive.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:5 https://download.docker.com/linux/ubuntu bionic InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
curl is already the newest version (7.58.0-2ubuntu3.16).
gnupg is already the newest version (2.2.4-1ubuntu1.4).
gnupg set to manually installed.
software-properties-common is already the newest version (0.96.24.32.18).
The following packages were automatically installed and are no longer required:
linux-headers-4.15.0-166
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
...
Luis@Linux5:~$
```

Despues *curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add -*



```
Luis@Linux5:~$ curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add -
Luis@Linux5:~$
```

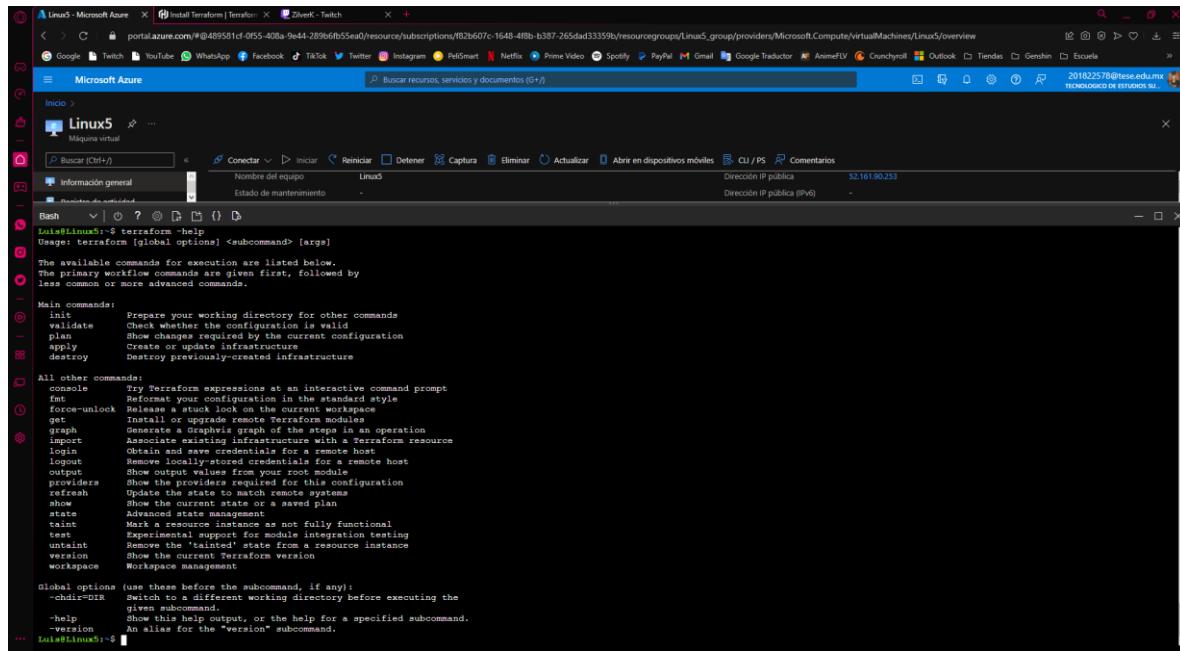
```
Despues    sudo    apt-add-repository    "deb    [arch=amd64]  
https://apt.releases.hashicorp.com $(lsb_release -cs) main"
```

```
Luis@Linux5:~$ sudo apt-get update
Hit:1 http://archive.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://archive.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://archive.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu bionic-security InRelease
Hit:5 https://apt.releases.hashicorp.com/ubuntu bionic InRelease
Reading package lists...
Done
Building dependency tree
Reading state information...
Done
curl is already the newest version (7.58.0-0ubuntu3.16).
gnupg is already the newest version (2.2.1-1ubuntu1.4).
gnupg2 is not installed.
software-properties-common is already the newest version (0.96.24.32.18).
The following package was automatically installed and is no longer required:
libcurl3
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Luis@Linux5:~$ curl -fastic https://apt.releases.hashicorp.com/gpg | sudo apt-key add -
OK
Luis@Linux5:~$ sudo apt-add-repository 'deb [arch=amd64] https://apt.releases.hashicorp.com $(lsb_release -cs) main'
Hit:1 http://archive.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://archive.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://archive.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu bionic InRelease
Get:5 https://apt.releases.hashicorp.com/ubuntu bionic InRelease [9497 B]
Get:7 https://apt.releases.hashicorp.com/ubuntu bionic/main amd64 Packages [46.9 kB]
Fetched 56.4 KB in 1s (102 kB/s)
Reading package lists... Done
```

Despues `sudo apt-get update && sudo apt-get install terraform`

```
Luis@Linux5:~$ curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add -
OK
Luis@Linux5:~$ sudo apt-add-repository "deb [arch=wasm64] https://apt.releases.hashicorp.com $(lsb_release -cs) main"
Hit:1 http://azuredash.azure.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://azuredash.azure.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://azuredash.azure.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://azuredash.azure.archive.ubuntu.com/ubuntu bionic-security InRelease
Get:5 https://apt.releases.hashicorp.com bionic InRelease [9497 B]
Hit:6 http://security.ubuntu.com/ubuntu bionic-security InRelease
Get:7 https://apt.releases.hashicorp.com bionic/main amd64 Packages [46.9 kB]
Fetched 564 kB in 1s (102 kB/s)
Reading package lists... Done
Luis@Linux5:~$ sudo apt-get update &> /dev/null
Hit:1 http://azuredash.azure.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://azuredash.azure.archive.ubuntu.com/ubuntu bionic-updates InRelease [89.7 kB]
Get:3 http://azuredash.azure.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease
Get:5 https://apt.releases.hashicorp.com bionic InRelease
Hit:6 https://apt.releases.hashicorp.com bionic/main InRelease
Fetched 163 kB in 1s (256 kB/s)
Reading package lists... Done
Luis@Linux5:~$ sudo apt-get install terraform
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
linux-image-4.15.0-166
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
terraform
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 18.7 MB of archives.
After this operation, 62.0 MB of additional disk space will be used.
Get:1 https://apt.releases.hashicorp.com/bionic/main amd64 terraform amd64 1.1.4 [18.7 MB]
Fetched 18.7 MB in 0s (39.2 MB/s)
Selecting previously unselected package terraform.
(Reading database ... 77261 files and directories currently installed.)
Preparing to unpack .../terraform_1.1.4_amd64.deb ...
Unpacking terraform (1.1.4) ...
Setting up terraform (1.1.4) ...
Luis@Linux5:~$
```

Despues lo verificamos que se instalara correctamente con *terraform -help*



```
Luis@Linux5:~$ terraform -help
Usage: terraform [<global options>] <subcommand> [<args>]

The available commands for execution are listed below.
The primary workflow commands are given first, followed by
less common or more advanced commands.

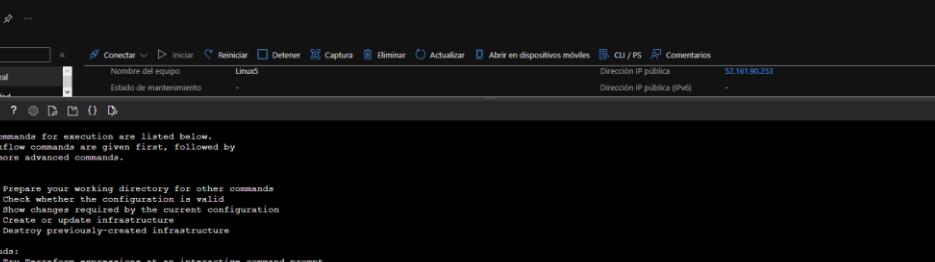
Main commands:
  init      Prepare your working directory for other commands
  validate   Check whether the configuration is valid
  plan      Show changes required by the current configuration
  apply     Create or update infrastructure
  destroy   Destroy previously-created infrastructure

All other commands:
  console   Try Terraform expressions at an interactive command prompt
  fmt       Reformat your configuration in the standard style
  force-unlock Remove a stuck lock on the current workspace
  get       Install or upgrade modules in your workspace
  graph     Generate a Graphviz graph of the steps in an operation
  import    Associate existing infrastructure with a Terraform resource
  login     Obtain and store credentials for a remote host
  logout    Logout locally and remove credentials for a remote host
  output    Show output values from your root module
  providers Show the providers required for this configuration
  refresh   Update state from one or more remote systems
  show      Show the current state of a saved plan
  state     Advanced state management
  taint     Mark a resource instance as not fully functional
  test      Register a module for module integration testing
  untaint  Remove the 'tainted' state from a resource instance
  version   Show the current Terraform version
  workspace Workspace management

Global options (use these before the subcommand, if any):
  -chdir=DIR  Switch to a different working directory before executing the
             given subcommand.
  -help       show this help output, or the help for a specified subcommand.
  -version    An alias for the "version" subcommand.
...
```

Practica 5 - Crear archivo IaC Terraform en Linux Ubuntu 18.

Empezamos con los comandos; `touch ~/.bashrc` y `terraform -install-autocomplete`



The screenshot shows a Microsoft Edge browser window with the following tabs:

- Linux5 - Microsoft Azure
- Install Terraform | Terraform
- SliverK - Twitch

The main content area displays the Azure portal interface for a virtual machine named "Linux5". The "Información general" tab is selected, showing details like the VM name, IP addresses, and connection status.

A terminal window titled "Linux5" is open, showing the following command-line session:

```
Linux5 ~ % Búsqueda: Buscar recursos, servicios y documentos (G+) Búsqueda: Buscar recursos, servicios y documentos (G+)
Microsoft Azure
Linux5
Máquina virtual
Buscar (Ctrl+F) Conectar Iniciar Detener Captura Eliminar Actualizar Abrir en dispositivos móviles CLI / PS Comentarios
Nombre del equipo Linux5 Estado de mantenimiento
Dirección IP pública $2.161.90.253 Dirección IP pública (IPv6) -
Bash
The available commands for execution are listed below.
The primary workflow commands are given first, followed by less common or more advanced commands.
Main commands:
  init      Prepare your working directory for other commands
  validate  Check whether the configuration is valid
  plan     Show changes required by the current configuration
  apply    Create or update infrastructure
  destroy   Destroy previously created infrastructure
All other commands:
  console   Run Terraform expressions at an interactive command prompt
  fmt       Reformat your configuration in the standard style
  force-unlock Release a stuck lock on the current workspace
  get      Install or upgrade remote Terraform modules
  graph    Generate a dependency graph for the current operation
  import   Associate existing infrastructure with a Terraform resource
  login    Obtain and save credentials for a remote host
  logout   Remove locally-stored credentials for a remote host
  refresh  Refresh the provider state
  providers Show the providers required for this configuration
  refresh  Update the state to match remote systems
  show     Show the current state or a saved plan
  state   Show the state file
  taint   Mark a resource instance as not fully functional
  test    Experimental support for module integration testing
  unlist  Remove the "tainted" state from a resource instance
  version Show the current Terraform version
  workspace Workspace management

Global options (use those before the subcommand, if any):
  -cdir<DIR> Switch to a different working directory before executing the given subcommand.
  -help     Show this help output, or the help for a specified subcommand.
  -version  An alias for the "version" subcommand.
  -t          Use the basic
  -terraform -install-autocomplete
```

Despues creamos un directorio llamado *terraform* y dentro de el un archivo tipo *terraform* llamado *main* con los siguientes comandos;

Mkdir terraform

Cd terraform

Nano main.tf

Peguamos el siguiente código dentro del archivo;

terraform {

required_providers {

docker = {

```
source = "kreuzwerker/docker"
```

version = " $\sim> 2.13.0$ "

}

}

```
}
```

```
provider "docker" {}
```

```
resource "docker_image" "nginx" {  
    name      = "nginx:latest"  
    keep_locally = false  
}
```

```
resource "docker_container" "nginx" {  
    image = docker_image.nginx.latest  
    name  = "tutorial"  
    ports {  
        internal = 80  
        external = 8000  
    }  
}
```

Guardamos con *ctrl+X*

A screenshot of the Microsoft Azure Cloud Shell interface. The title bar shows "Linux5 - Microsoft Azure". The main area displays a terminal window titled "Bash" with the command "nano main.tf" running. The terminal content is the Terraform configuration file:

```
#!/usr/bin/nano 2.9.3
main.tf

terraform {
    required_providers {
        docker = {
            source  = "kreuzwerker/docker"
            version = ">= 2.13.0"
        }
    }

    provider "docker" {}

    resource "docker_image" "nginx" {
        name     = "nginx:latest"
        keep_locally = false
    }

    resource "docker_container" "nginx" {
        image = docker_image.nginx.latest
        name  = "tutorial"
        ports {
            internal = 80
            external = 8000
        }
    }
}
```

Comprobamos con *cat main.tf*

A screenshot of the Microsoft Azure Cloud Shell interface. The title bar shows "Linux5 - Microsoft Azure". The main area displays a terminal window titled "Bash" with the command "cat main.tf" running. The terminal content shows the Terraform configuration file:

```
Luis@Linux5:~/Desktop$ cat main.tf
#!/usr/bin/nano 2.9.3
main.tf

terraform {
    required_providers {
        docker = {
            source  = "kreuzwerker/docker"
            version = ">= 2.13.0"
        }
    }

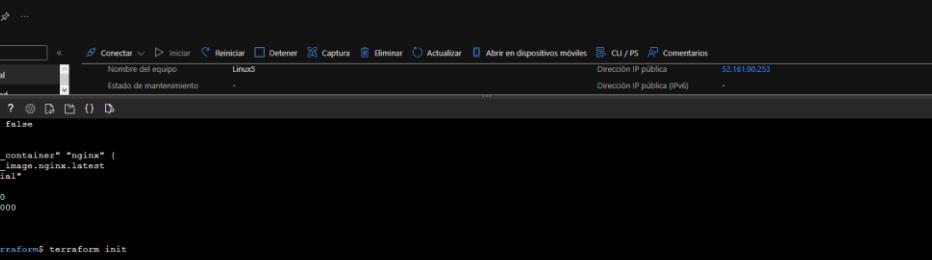
    provider "docker" {}

    resource "docker_image" "nginx" {
        name     = "nginx:latest"
        keep_locally = false
    }

    resource "docker_container" "nginx" {
        image = docker_image.nginx.latest
        name  = "tutorial"
        ports {
            internal = 80
            external = 8000
        }
    }
... Luis@Linux5:~/Desktop$
```

Practica 6 - Inicializar IaC de Terraform en Linux Ubuntu 18.

Ingresamos el comando *terraform init*



```
Luis@Linux5:~/terraform$ terraform init

Initializing the backend...

Initializing provider plugins...
- Finding kreuzwerker/docker versions matching ">~> 2.13.0"...
- Installing kreuzwerker/docker v2.13.0...
- Installed kreuzwerker/docker v2.13.0 (self-signed, key ID 24M54F214569A9A5)

Partitions and community providers are signed by their developers.
If you'd like to know more about provider signing, you can read about it here:
https://www.terraform.io/docs/cli/plugins/signing.html

Terraform has created a lock file, .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

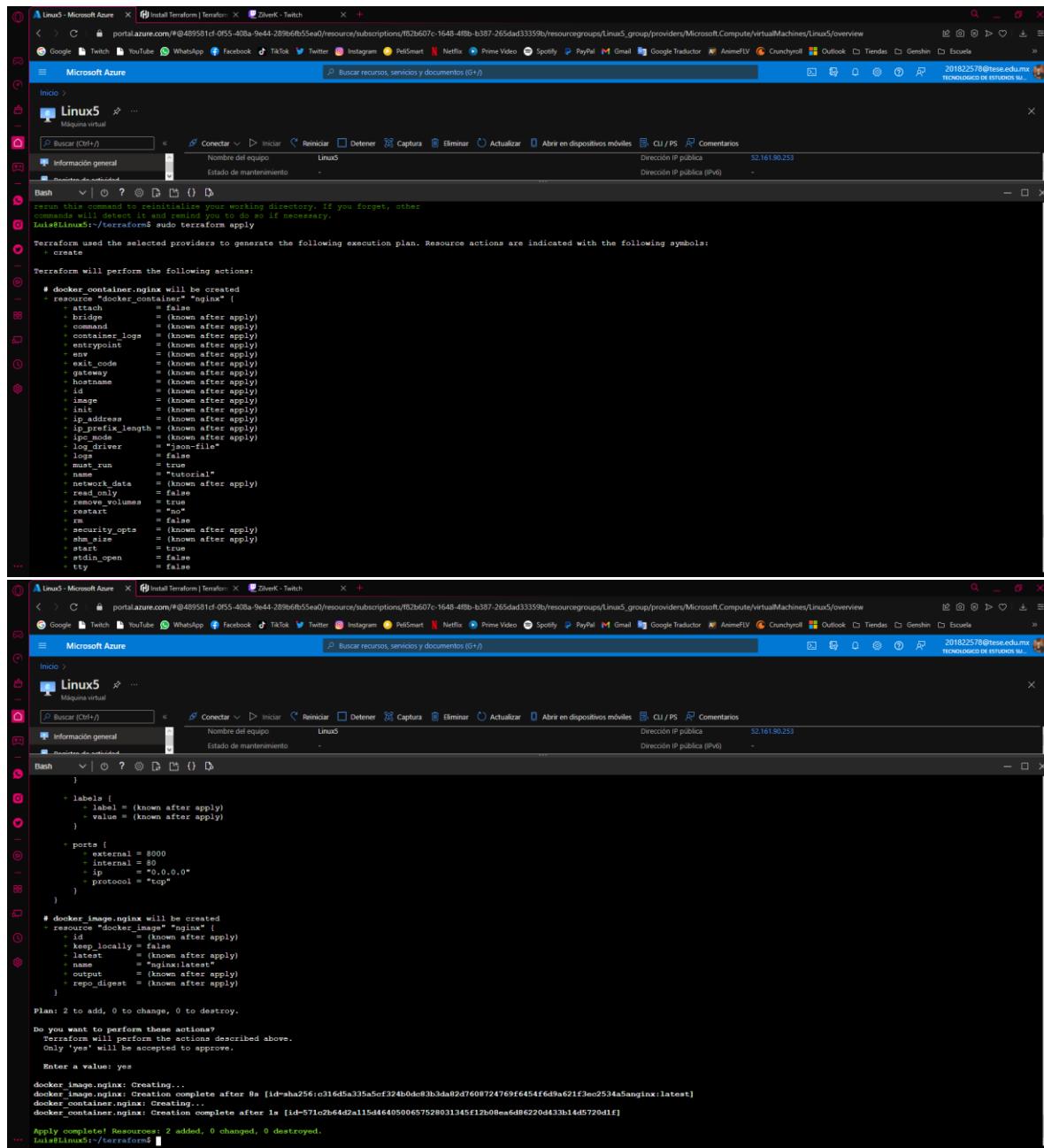
You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
run this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

Luis@Linux5:~/terraform$
```

Practica 7 - Crear Infraestructura en Terraform de un microservicio en Ubuntu 18.

Ingresamos el comando *sudo terraform apply*



```
Luis@Linux5:~/terraform$ sudo terraform apply
rerun this command to reinitialize your working directory. If you forget, other
commands will assume this is the state you want, so be careful!
Luis@Linux5:~/terraform$ sudo terraform apply
Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create
Terraform will perform the following actions:

# docker_container.nginx will be created
resource "docker_container" "nginx" {
  + attach = false
  + bridge = (known after apply)
  + command = (known after apply)
  + container_logs = (known after apply)
  + entrypoint = (known after apply)
  + env = (known after apply)
  + exit_code = (known after apply)
  + gateway = (known after apply)
  + hostname = (known after apply)
  + id = (known after apply)
  + image = (known after apply)
  + init = (known after apply)
  + ip_address = (known after apply)
  + ip_prefix_length = (known after apply)
  + log_driver = "json-file"
  + logs = false
  + max_rum = true
  + name = "tutorial"
  + network_data = (known after apply)
  + read_only = false
  + remove_volumes = true
  + restart = "no"
  + rm = false
  + security_opts = (known after apply)
  + std_in = (known after apply)
  + start = true
  + std_in_open = false
  + tty = false
}

# docker_image.nginx will be created
resource "docker_image" "nginx" {
  + id = (known after apply)
  + keep_locally = false
  + latest = (known after apply)
  + name = "nginx:latest"
  + repository = (known after apply)
  + repo_digest = (known after apply)
}

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

Enter a value: yes
docker image.nginx: Creating...
docker image.nginx: Creation complete after 8s [id=sha256:c316d5a335a5cf324b0de93b3da82d7608724769f6454f6d9a621f3ec2534a5nginx:latest]
docker container.nginx: Creating...
docker container.nginx: Creation complete after 1s [id=571c2b64d2a115d464050657528031345f12b08e6d86220d433b14d572d1f]
Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
Luis@Linux5:~/terraform$
```

Despues en nuestro portal de azure nos dirigimos a *Redes* y creamos una nueva regla que tendrá el puerto 8000

The screenshot shows the Azure portal interface for a Linux VM named 'Linux5'. In the left sidebar, under 'Configuración' (Configuration), 'Redes' (Networks) is selected. On the main page, under 'Redes' (Networks), there is a section for 'Reglas de puerto de entrada' (Inbound rules). A modal window titled 'Agregar regla de seguridad de entrada' (Add inbound security rule) is open, showing the configuration for a new rule:

Prioridad	Nombre	Puerto	Protocolo	Origen	Destino	Acción
300	SSH	22	TCP	Cualquier	Cualquier	Permitir
65000	AllowInbound	Cualquier	Cualquier	VirtualNetwork	Cualquier	Permitir
65001	AllowLoadBalancerInbound	Cualquier	Cualquier	AzureLoadBalancer	Cualquier	Permitir
65500	DenyAllInbound	Cualquier	Cualquier	Cualquier	Cualquier	Denegar
310	Port_8000	8000	Cualquier	Cualquier	Cualquier	Permitir

The 'Nombre' (Name) field is set to 'Port_8000'. The 'Puerto' (Port) field is set to '8000'. The 'Protocolo' (Protocol) dropdown is set to 'Any'. The 'Origen' (Source) dropdown is set to 'Cualquier'. The 'Destino' (Destination) dropdown is set to 'Any'. The 'Acción' (Action) dropdown is set to 'Permitir'.

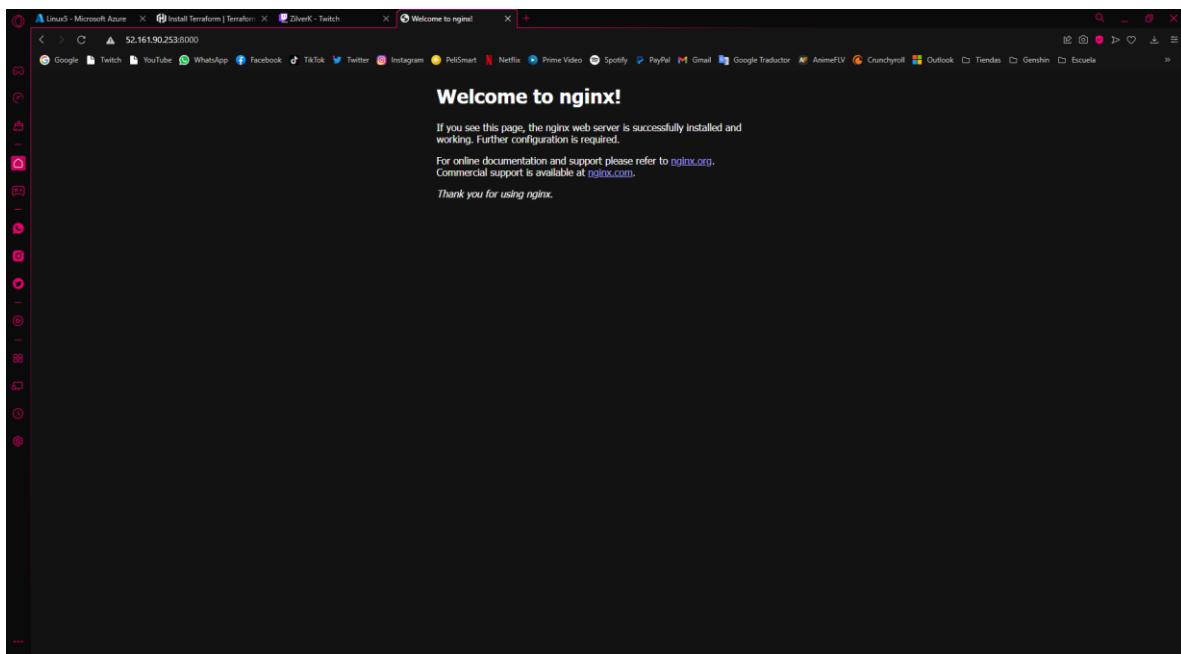
Verificamos que se haya creado.

The screenshot shows the same view as the previous one, but now the newly created rule is visible in the list:

Prioridad	Nombre	Puerto	Protocolo	Origen	Destino	Acción
300	SSH	22	TCP	Cualquier	Cualquier	Permitir
310	Port_8000	8000	Cualquier	Cualquier	Cualquier	Permitir
65000	AllowInbound	Cualquier	Cualquier	VirtualNetwork	Cualquier	Permitir
65001	AllowLoadBalancerInbound	Cualquier	Cualquier	AzureLoadBalancer	Cualquier	Permitir
65500	DenyAllInbound	Cualquier	Cualquier	Cualquier	Cualquier	Denegar

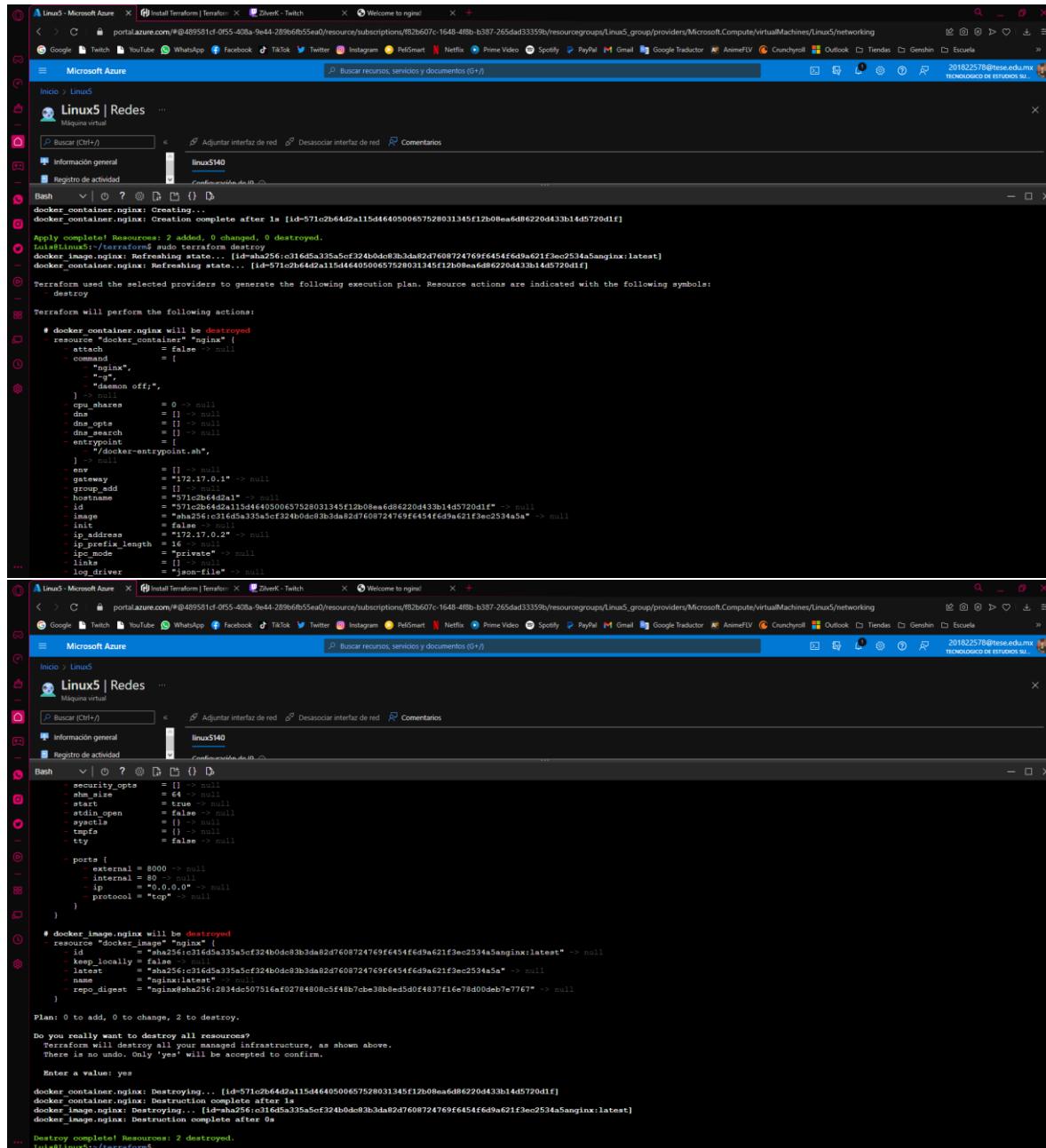
The rule 'Port_8000' has been successfully added with a priority of 310, allowing traffic on port 8000 from any source to any destination via TCP.

Comprobamos la conexión con la dirección de IP y el número de puerto.



Practica 8 - Destruir Infraestructura en Terraform de un microservicio en Ubuntu 18.

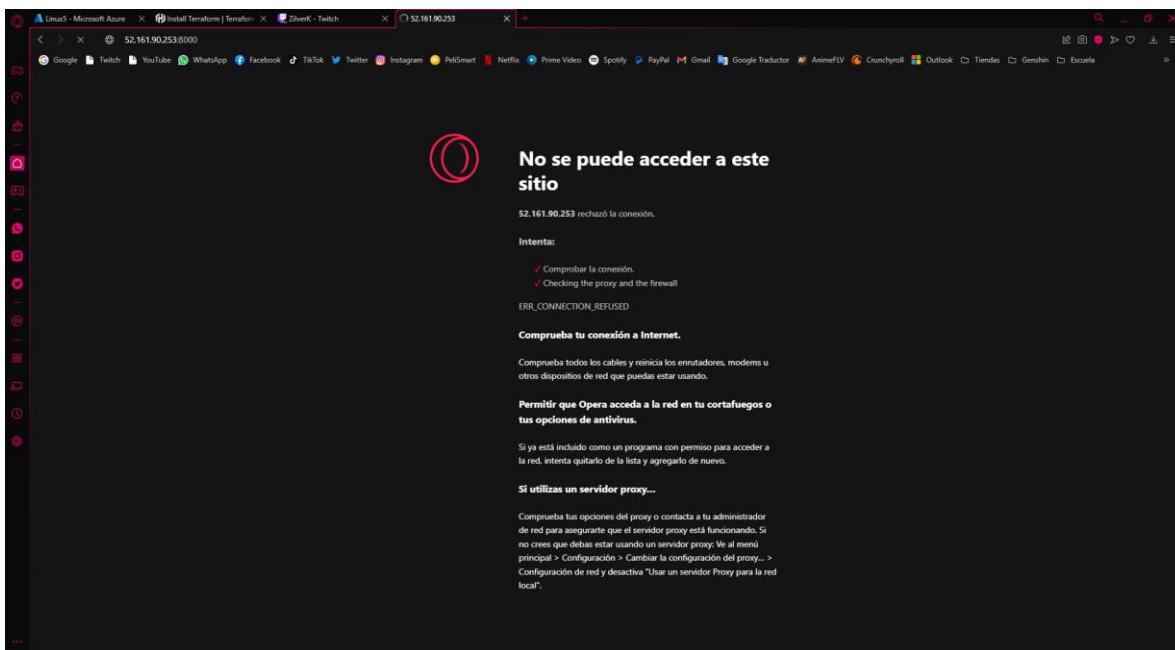
Para destruirlo solamente ingresamos el comando *sudo terraform destroy* y nos pedirá nuestra contraseña.



The image shows two terminal windows side-by-side, both titled "Linux5 - Microsoft Azure". Both terminals are running the command "terraform destroy".

```
Linux5 | Redes ...  
Buscador (Ctrl+F) < > Ajustar interfaz de red Desasociar interfaz de red Comentarios  
Información general  
Registro de actividad  
Bash < ? > [ ] < > docker_container.nginx: Creating...  
docker_container.nginx: Creation complete after 1s [id=571c2b64d2a115d4640500657528031345f12b08ea6d86220d433b14d5720d1f]  
Apply complete! Resources: 2 added, 0 changed, 0 destroyed.  
Outputs:  
  - /etc/hosts: /etc/hosts -> terraform destroy  
  - docker_image.nginx: Refreshing state... [id=sha256:c16d5a335a5cf324b0dc83b3da82d7608724769f645ff6d9a621f3ec2534a5anginx:latest]  
  - docker_container.nginx: Refreshing state... [id=571c2b64d2a115d4640500657528031345f12b08ea6d86220d433b14d5720d1f]  
Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:  
  destroy  
Terraform will perform the following actions:  
# docker_container.nginx will be destroyed  
resource "docker_container" "nginx" {  
  attach = false => null  
  command = [  
    "-nginx",  
    "-g",  
    "-demon off";  
  ] =>  
  cpu_shares = 0 => null  
  dns = [] => null  
  dns_opts = [] => null  
  dns_search = [] => null  
  entrypoint = [  
    "./docker-entrypoint.sh",  
  ] => null  
  env = [] => null  
  gateway = "172.17.0.1" => null  
  group_add = [] => null  
  hostname = "571c2b64d2a115d4640500657528031345f12b08ea6d86220d433b14d5720d1f" => null  
  id = "sha256:c16d5a335a5cf324b0dc83b3da82d7608724769f645ff6d9a621f3ec2534a5a" => null  
  image = "sha256:c16d5a335a5cf324b0dc83b3da82d7608724769f645ff6d9a621f3ec2534a5a" => null  
  init = false => null  
  ip_address = "172.17.0.2" => null  
  ip_prefix_length = 16 => null  
  ipc_mode = "private" => null  
  links = [] => null  
  log_driver = "json-file" => null  
  
# docker_image.nginx will be destroyed  
resource "docker_image" "nginx" {  
  id = "sha256:c16d5a335a5cf324b0dc83b3da82d7608724769f645ff6d9a621f3ec2534a5anginx:latest"  
  keep_locally = true => null  
  latest = "sha256:c16d5a335a5cf324b0dc83b3da82d7608724769f645ff6d9a621f3ec2534a5a" => null  
  name = "nginx:latest" => null  
  repo_digest = "nginx@sha256:2834dc507516af0278480c5f48b7cb38bbed5d0f4837f1e78d0deb7e7767" => null  
}  
Plan: 0 to add, 0 to change, 2 to destroy.  
Do you really want to destroy all resources?  
Terraform will destroy all your managed infrastructure, as shown above.  
There is no undo. Only 'yes' will be accepted to confirm.  
Enter a value: yes  
docker_container.nginx: Destroying... [id=571c2b64d2a115d4640500657528031345f12b08ea6d86220d433b14d5720d1f]  
docker_container.nginx: Destruction complete after 1s  
docker_image.nginx: Destroying... [id=sha256:c16d5a335a5cf324b0dc83b3da82d7608724769f645ff6d9a621f3ec2534a5anginx:latest]  
docker_image.nginx: Destruction complete after 0s  
Destroy complete! Resources: 2 destroyed.  
tut@Linux5:~/terraform$
```

Comprobamos la conexión.



Practica 10 - Resumen sobre videos de Azure Data Fundamentals.

Empieza hablando sobre la cantidad de datos que han generado los sistemas, aplicaciones y dispositivos que ha aumentado mucho en su tecnología de hoy en día, ya que los datos están en casi todos los lugares.

Podemos encontrarlos con diferentes estructuras y formatos. Tenemos que entender los datos y explorarlos para así poder revelar hechos interesantes que ayuda a obtener información significativa sobre lo que se quiera obtener

Posteriormente nos enseña a organizar y procesar dichos datos, así como también las bases de datos relacionales y no relacionales, el manejo de los datos mediante el procesamiento transaccional y también por lotes y de transmisión, agregando el uso de herramientas y técnicas comprendidas en el mundo real, imaginando a un analista de datos para una gran organización de consumidores,

También menciona las bases de datos, donde nos explican porque son tan simples como una hoja de cálculo de escritorio o tan compleja como un sistema global que contiene datos enormes de información altamente estructurada, y cómo se mencionaba al inicio un enfoque común es almacenar datos en un formato tabular, con filas y columnas. Podemos definir relaciones entre tablas. A estas últimas se les llaman bases de datos relacionales algo que incluso he llegado a cursar en materias pasadas y que también pueden ser semiestructuradas o no estructuradas, con datos semiprocesados o no procesados.

Las bases de datos son gestionadas por un sistema de gestión de bases de datos (DBMS del inglés *Data base manager system*). Este maneja los aspectos físicos de una base de datos, como dónde y cómo se almacena, quién puede acceder a ella y cómo garantizar que esté disponible cuando sea necesario, además de que muchas organizaciones dependen de la información almacenada en sus bases de datos para tomar decisiones comerciales, ya que en el pasado, estas organizaciones ejecutaban sus DBMS en las instalaciones, sin embargo, este enfoque requiere que la organización mantenga su propia infraestructura de hardware, por lo tanto, un número cada vez

mayor de empresas está cambiando sus bases de datos a la nube, donde los costos de configuración y mantenimiento de la infraestructura se reducen considerablemente.

Posteriormente en el segundo video se nos explica como los datos vienen en todas las formas y tamaños, y que se pueden usar para una gran cantidad de propósitos, sin embargo, el modelo relacional podría no ser el esquema más apropiado, ya que la estructura de los datos puede ser demasiado variada para modelar fácilmente como un conjunto de tablas relacionales.

Los requisitos de procesamiento de datos no son los mejores intentar convertir estos datos al formato relacional. Sale más rentable el uso de repositorios no relacionales que puedan almacenar datos en su formato original, pero que permitan un rápido almacenamiento y recuperación de esos datos.