Logotipo

Descripción generada automáticamente

**Entrega Evaluación Parcial 2 - Laboratorio práctico de Contenedores**

Integrantes:

Pedro Antor

Brian Lillo

Docente: Sandra Henríquez

Sección: 002D

Asignatura: TECNOLOGIAS DE VIRTUALIZACION

1. Realiza la instalación de Git en su equipo de laboratorio Windows.

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

2. Registra una cuenta de GitHub para poder gestionar un repositorio remoto para su código, crea un repositorio público en dicha cuenta.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico, Sitio web

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

<https://github.com/peantorduoc/duocuc>

3. Crea un directorio local para organizar los archivos Dockerfile disponibles creados previamente, versionando estos archivos con comandos básicos de Git.

mkdir -p ~/repos/duocuc

sudo apt-get install git -y

Reading package lists... Done

Building dependency tree... Done

Reading state information... Done

git is already the newest version (1:2.34.1-1ubuntu1.9).

git set to manually installed.

0 upgraded, 0 newly installed, 0 to remove and 20 not upgraded.

git config --global user.name "Pedro Antor"

git config --global user.email "pe.antor@duocuc.cl"

git config --global alias.lg "log --graph --abbrev-commit --decorate --format=format:'%C(bold blue)%h%C(reset) - %C(bold green)(%ar)%C(reset) %C(white)%s%C(reset) %C(dim white)- %an%C(reset)%C(bold yellow)%d%C(reset)' --all"

git config --global alias.s status --short

git config --global init.defaultBranch

cd ~/repos/duocuc/

git init

mkdir -p ~/repos/duocuc/TECNOLOGIAS\_DE\_VIRTUALIZACION\_002D/parcial2/

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

4. Carga en el repositorio que, creado en su cuenta de GitHub, Una vez realizado el commit de los archivos

ssh-keygen -t ed25519 -C "pe.antor@duocuc.cl" -f ~/pantor

mkdir ~/.ssh

mv ~/pantor\* ~/.ssh/

chmod 400 ~/.ssh/pantor

git remote add duocuc git@github.com:peantorduoc/duocuc.git

git remote -v

eval "$(ssh-agent -s)"

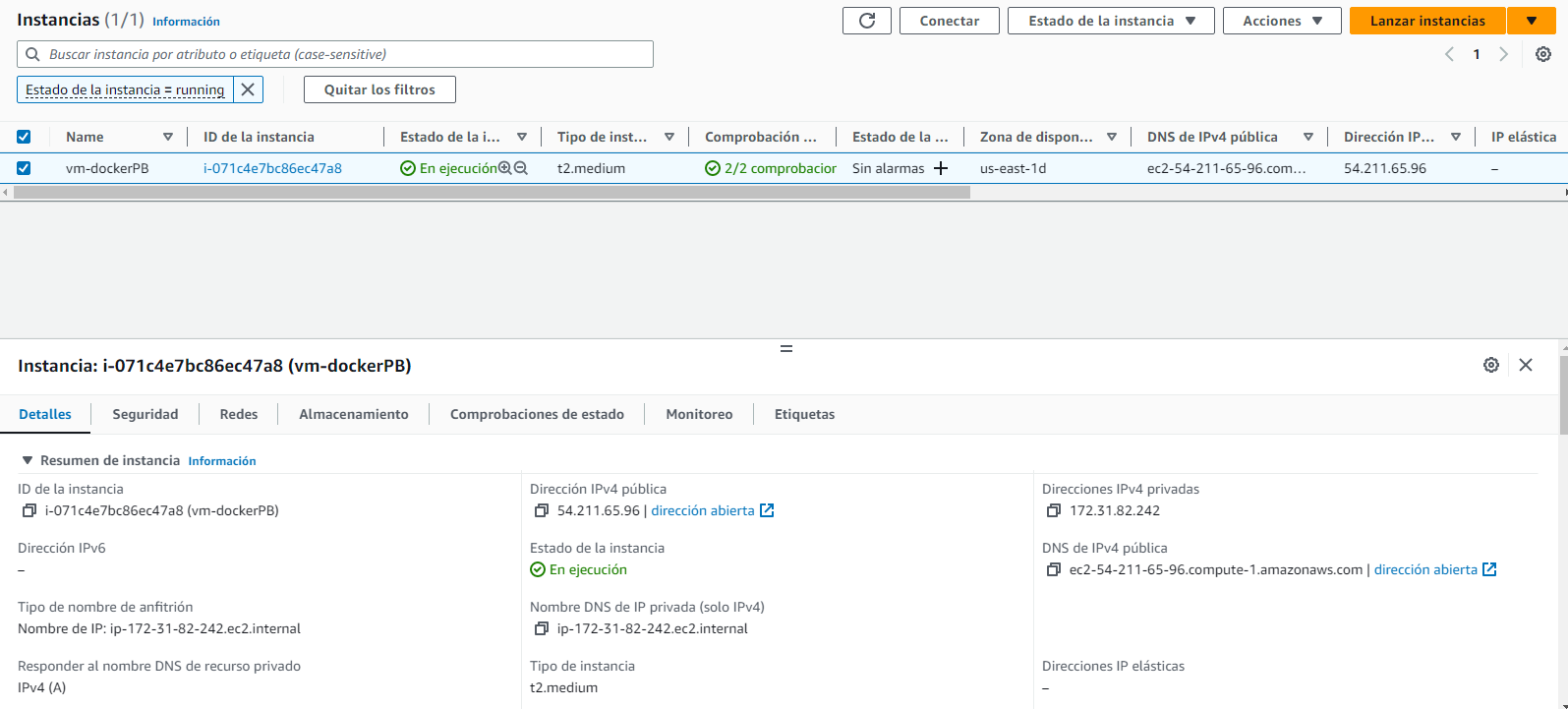
ssh-add ~/.ssh/pantor

ssh -T git@github.com

git push --set-upstream duocuc main

git push

5. Crea una instancia de distribución Linux en la nube pública (ej. AWS) y clona el código desde el repositorio de GitHub.



pantor@ip-172-31-82-242:~/repos$ git clone https://github.com/peantorduoc/duocuc.git

Cloning into 'duocuc'...

remote: Enumerating objects: 32, done.

remote: Counting objects: 100% (32/32), done.

remote: Compressing objects: 100% (21/21), done.

remote: Total 32 (delta 3), reused 32 (delta 3), pack-reused 0

Receiving objects: 100% (32/32), done.

Resolving deltas: 100% (3/3), done.

ubuntu@ip-172-31-82-242:~/repos$ ls -lR

.:

total 4

drwxrwxr-x 4 ubuntu ubuntu 4096 May 26 01:24 duocuc

./duocuc:

total 4

drwxrwxr-x 3 ubuntu ubuntu 4096 May 26 01:24 TECNOLOGIAS\_DE\_VIRTUALIZACION\_002D

./duocuc/TECNOLOGIAS\_DE\_VIRTUALIZACION\_002D:

total 4

drwxrwxr-x 4 ubuntu ubuntu 4096 May 26 01:24 parcial2

./duocuc/TECNOLOGIAS\_DE\_VIRTUALIZACION\_002D/parcial2:

total 8

drwxrwxr-x 2 ubuntu ubuntu 4096 May 26 01:24 dirdockefilefPB

drwxrwxr-x 2 ubuntu ubuntu 4096 May 26 01:24 www

./duocuc/TECNOLOGIAS\_DE\_VIRTUALIZACION\_002D/parcial2/dirdockefilefPB:

total 12

-rw-rw-r-- 1 ubuntu ubuntu 230 May 26 01:24 Comandos\_Docker.txt

-rw-rw-r-- 1 ubuntu ubuntu 845 May 26 01:24 Comandos\_Git.txt

-rw-rw-r-- 1 ubuntu ubuntu 285 May 26 01:24 docker-compose.yml

./duocuc/TECNOLOGIAS\_DE\_VIRTUALIZACION\_002D/parcial2/www:

total 4

-rw-rw-r-- 1 ubuntu ubuntu 334 May 26 01:24 index.html

6. Inicia un contenedor a partir de los archivos del repositorio y demuestra datos básicos de la instancia( Visualiza estadísticas del contenedor, reinicia, detiene, renombra, elimina el contenedor y la imagen)

sudo apt-get update

sudo apt-get install ca-certificates curl gnupg

sudo install -m 0755 -d /etc/apt/keyrings

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg

sudo chmod a+r /etc/apt/keyrings/docker.gpg

echo \

"deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \

"$(. /etc/os-release && echo "$VERSION\_CODENAME")" stable" | \

sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

sudo docker run hello-world

Unable to find image 'hello-world:latest' locally

latest: Pulling from library/hello-world

719385e32844: Pull complete

Digest: sha256:fc6cf906cbfa013e80938cdf0bb199fbdbb86d6e3e013783e5a766f50f5dbce0

Status: Downloaded newer image for hello-world:latest

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.

2. The Docker daemon pulled the "hello-world" image from the Docker Hub.

(amd64)

3. The Docker daemon created a new container from that image which runs the

executable that produces the output you are currently reading.

4. The Docker daemon streamed that output to the Docker client, which sent it

to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:

https://hub.docker.com/

For more examples and ideas, visit:

https://docs.docker.com/get-started/

sudo docker ps -a

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

d3c3db710d23 hello-world "/hello" 56 seconds ago Exited (0) 55 seconds ago kind\_beaver

Para poder desplegar un contenedor hay que instalar Docker Engine y eso lo pide en el paso 8.

$ sudo apt install docker-compose

$ sudo docker-compose up -d

Starting webserver ... done

sudo docker ps -a

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

430e970c89a6 nginx "/docker-entrypoint.…" 15 minutes ago Up 43 seconds 0.0.0.0:8000->80/tcp, :::8000->80/tcp webserver

d3c3db710d23 hello-world "/hello" 27 minutes ago Exited (0) 27 minutes ago

===> Detener contenedor

pantor:.ssh >> docker stop webserver

webserver

pantor:.ssh >> docker ps -a

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

e965fb73516e nginx "/docker-entrypoint.…" 2 hours ago Exited (0) 7 seconds ago webserver

===> Iniciar contenedor

pantor:.ssh >> docker start webserver

webserver

pantor:.ssh >> docker ps -a

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

e965fb73516e nginx "/docker-entrypoint.…" 2 hours ago Up 4 seconds 0.0.0.0:8000->80/tcp webserver

===> Renombrar contenedor

pantor:.ssh >> docker rename webserver webserver2

pantor:.ssh >> docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

e965fb73516e nginx "/docker-entrypoint.…" 2 hours ago Up 47 seconds 0.0.0.0:8000->80/tcp webserver2

===> Listar las imagenes de los contenedores

pantor:.ssh >> docker image ls

REPOSITORY TAG IMAGE ID CREATED SIZE

nginx latest f9c14fe76d50 27 hours ago 143MB

===> Viendo las estadisticas por contenedor o para todos los contenedores

pantor:.ssh >> docker stats webserver2

CONTAINER ID NAME CPU % MEM USAGE / LIMIT MEM % NET I/O BLOCK I/O PIDS

e965fb73516e webserver2 0.00% 6.762MiB / 7.475GiB 0.09% 1.02kB / 0B 0B / 0B 9

pantor:.ssh >> docker stats -a

CONTAINER ID NAME CPU % MEM USAGE / LIMIT MEM % NET I/O BLOCK I/O PIDS

e8b5400bd77e webserver3 0.00% 6.859MiB / 7.475GiB 0.09% 946B / 0B 0B / 0B 9

e965fb73516e webserver2 0.00% 6.762MiB / 7.475GiB 0.09% 1.16kB / 0B 0B / 0B 9

===> Eliminar un contenedor

pantor:.ssh >> docker kill webserver3

webserver3

pantor:.ssh >> docker ps -a

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

e8b5400bd77e nginx "/docker-entrypoint.…" About a minute ago Exited (137) 6 seconds ago webserver3

e965fb73516e nginx "/docker-entrypoint.…" 2 hours ago Up 5 minutes 0.0.0.0:8000->80/tcp webserver2

pantor:.ssh >> docker rm webserver3

webserver3

pantor:.ssh >> docker ps -a

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

e965fb73516e nginx "/docker-entrypoint.…" 2 hours ago Up 21 minutes 0.0.0.0:8000->80/tcp webserver2

7. Lanza una instancia de distribución Ubuntu en la nube pública de AWS utilizado una subred pública.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

8. Instala Docker Engine según lo recomendado para la distribución utilizada.

sudo apt-get update

sudo apt-get install ca-certificates curl gnupg

sudo install -m 0755 -d /etc/apt/keyrings

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg

sudo chmod a+r /etc/apt/keyrings/docker.gpg

echo \

"deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \

"$(. /etc/os-release && echo "$VERSION\_CODENAME")" stable" | \

sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

sudo docker run hello-world

Unable to find image 'hello-world:latest' locally

latest: Pulling from library/hello-world

719385e32844: Pull complete

Digest: sha256:fc6cf906cbfa013e80938cdf0bb199fbdbb86d6e3e013783e5a766f50f5dbce0

Status: Downloaded newer image for hello-world:latest

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.

2. The Docker daemon pulled the "hello-world" image from the Docker Hub.

(amd64)

3. The Docker daemon created a new container from that image which runs the

executable that produces the output you are currently reading.

4. The Docker daemon streamed that output to the Docker client, which sent it

to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:

https://hub.docker.com/

For more examples and ideas, visit:

https://docs.docker.com/get-started/

sudo docker ps -a

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

d3c3db710d23 hello-world "/hello" 56 seconds ago Exited (0) 55 seconds ago kind\_beaver

9. Inicia un contenedor con servicio de Nginx utilizando bind mounts.

===> Instalando Jquery para filtrar la salida del inspector

pantor:.ssh >> sudo apt-get install jq

Reading package lists... Done

Building dependency tree... Done

Reading state information... Done

The following additional packages will be installed:

libjq1 libonig5

The following NEW packages will be installed:

jq libjq1 libonig5

0 upgraded, 3 newly installed, 0 to remove and 11 not upgraded.

Need to get 353 kB of archives.

After this operation, 1096 kB of additional disk space will be used.

Do you want to continue? [Y/n] y

Get:1 http://archive.ubuntu.com/ubuntu lunar/main amd64 libonig5 amd64 6.9.8-1 [168 kB]

Get:2 http://archive.ubuntu.com/ubuntu lunar/main amd64 libjq1 amd64 1.6-2.1ubuntu3 [133 kB]

Get:3 http://archive.ubuntu.com/ubuntu lunar/main amd64 jq amd64 1.6-2.1ubuntu3 [52.5 kB]

Fetched 353 kB in 1s (302 kB/s)

Selecting previously unselected package libonig5:amd64.

(Reading database ... 38167 files and directories currently installed.)

Preparing to unpack .../libonig5\_6.9.8-1\_amd64.deb ...

Unpacking libonig5:amd64 (6.9.8-1) ...

Selecting previously unselected package libjq1:amd64.

Preparing to unpack .../libjq1\_1.6-2.1ubuntu3\_amd64.deb ...

Unpacking libjq1:amd64 (1.6-2.1ubuntu3) ...

Selecting previously unselected package jq.

Preparing to unpack .../jq\_1.6-2.1ubuntu3\_amd64.deb ...

Unpacking jq (1.6-2.1ubuntu3) ...

Setting up libonig5:amd64 (6.9.8-1) ...

Setting up libjq1:amd64 (1.6-2.1ubuntu3) ...

Setting up jq (1.6-2.1ubuntu3) ...

Processing triggers for man-db (2.11.2-1) ...

Processing triggers for libc-bin (2.37-0ubuntu2) ...

===> Bind mounts

pantor:.ssh >> sudo docker inspect webserver | jq -r ".[].HostConfig.Binds[]"

/home/pantor/repos/duocuc/TECNOLOGIAS\_DE\_VIRTUALIZACION\_002D/parcial2/www/index.html:/usr/share/nginx/html/index.html:rw

10. Confirma funcionamiento del servicio visualizando la interfaz del contenedor a través de la IP pública de la instancia.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

11. Utiliza herramienta Inspect, para revisar detalle del montaje y completa datos requeridos para actualizar "NetworkSettings".

pantor:.ssh >> sudo docker inspect webserver | jq -r ".[].NetworkSettings"

{

"Bridge": "",

"SandboxID": "7de4cf32453bad41b4a9035e4347dfe0855d9d57ca7de55a293b92abdfccf762",

"HairpinMode": false,

"LinkLocalIPv6Address": "",

"LinkLocalIPv6PrefixLen": 0,

"Ports": {

"80/tcp": [

{

"HostIp": "0.0.0.0",

"HostPort": "8000"

}

]

},

"SandboxKey": "/var/run/docker/netns/7de4cf32453b",

"SecondaryIPAddresses": null,

"SecondaryIPv6Addresses": null,

"EndpointID": "",

"Gateway": "",

"GlobalIPv6Address": "",

"GlobalIPv6PrefixLen": 0,

"IPAddress": "",

"IPPrefixLen": 0,

"IPv6Gateway": "",

"MacAddress": "",

"Networks": {

"dirdockefilefpb\_default": {

"IPAMConfig": null,

"Links": null,

"Aliases": [

"webserver",

"client",

"e965fb73516e"

],

"NetworkID": "8087668dd0aae0859105d96192c4ea9b634397c9c6cb3fef87dbad0bdac9ff2c",

"EndpointID": "a934fcbbe9164a1fd6f2ceda98b3eae09939cebbfe62d37c2371d9325f6077f6",

"Gateway": "172.18.0.1",

"IPAddress": "172.18.0.2",

"IPPrefixLen": 16,

"IPv6Gateway": "",

"GlobalIPv6Address": "",

"GlobalIPv6PrefixLen": 0,

"MacAddress": "02:42:ac:12:00:02",

"DriverOpts": null

}

}

}