

Manual para instalar y configurar Host Rancher

Y administración de servicios con balanceador

2018

Integrantes:

- Berenice
- Ingrid
- Miguel
- Omar

CONTENIDO

PARTE 1 Creación de la vertical	2
Creación de Dockerfile con ENTRYPOINT.....	4
Creación del sh y dar permisos de acceso al sh	6
Construcción de la imagen con el archivo sh y jar	6
Ejecución de la vertical de manera local.....	7
Subir imagen del servicio a DockerHub	8
Comprobación de la imagen en DockerHub	9
 PARTE 2 Instalación y configuración de Host Rancher.....	10
Agregar servicios	13
Creación del balanceador.....	15
Funcionamiento del balanceador	17
 PARTE 3 Los agentes de Rancher	19
Cambiar nombre de usuario a la computadora	22
Crear y correr el agente de Rancher	24
Información adicional.....	25

PARTE 1

CREACIÓN DE LA VERTICAL

1. Se tiene una vertical en la carpeta vertx-sample con la siguiente ruta:
`/home/gustavo/Documents/vertx-sample`

2. Compilación de vertical en maven

Con ayuda de una imagen de docker que tenga maven creamos un volumen a la carpeta que contiene la vertical de la siguiente forma:

```
root@834f56ce1e48: /  
Archivo Editar Ver Buscar Terminal Ayuda  
gustavo@serverMike:~/Documents/vertx-sample$ docker run -it -v /home/gustavo/ Documents/vertx-sample:/vertx-sample kebbelar/jdk18-utf8-debug-maven bash  
root@834f56ce1e48: /#
```

Accedemos a la carpeta creada

```
root@834f56ce1e48: /vertx-sample  
Archivo Editar Ver Buscar Terminal Ayuda  
root@834f56ce1e48: /# ls  
bin dev home lib64 mnt proc run srv tmp usr vertx-sample  
boot etc lib media opt root sbin sys traspaso var  
root@834f56ce1e48: /# cd vertx-sample/  
root@834f56ce1e48: /vertx-sample# ls  
Dockerfile LICENSE mike_hoy README.md src test.sh  
doc.txt log pom.xml sample-1.0-SNAPSHOT-fat.jar target  
root@834f56ce1e48: /vertx-sample#
```

Una vez dentro de la carpeta vertx-sample ejecutamos el comando

```
mvn clean package
```

```
root@834f56ce1e48: /vertx-sample x
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@834f56ce1e48:/vertx-sample# mvn clean package
[INFO] Scanning for projects...
[INFO]
[INFO] -----
--
[INFO] Building sample 1.0-SNAPSHOT
[INFO] -----
--
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/httpcomponents/httpclient/4.5.2/httpclient-4.5.2.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/httpcomponents/httpclient/4.5.2/httpclient-4.5.2.pom (0 B at 0 B/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/httpcomponents/httpcomponents-client/4.5.2/httpcomponents-client-4.5.2.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/httpcomponents/httpcomponents-client/4.5.2/httpcomponents-client-4.5.2.pom (0 B at 0 B/s)
```

Si el código está bien construido no generará errores

```
root@834f56ce1e48: /vertx-sample x
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
the shaded jar.
[INFO] Including io.vertx:vertx-web:jar:3.2.1 in the shaded jar.
[INFO] Including io.vertx:vertx-auth-common:jar:3.2.1 in the shaded jar.
[INFO] Including org.apache.httpcomponents:httpclient:jar:4.5.2 in the shaded jar.
[INFO] Including org.apache.httpcomponents:httpcore:jar:4.4.4 in the shaded jar.
[INFO] Including commons-logging:commons-logging:jar:1.2 in the shaded jar.
[INFO] Including commons-codec:commons-codec:jar:1.9 in the shaded jar.
[INFO] -----
--
[INFO] BUILD SUCCESS
[INFO] -----
--
[INFO] Total time: 11.720 s
[INFO] Finished at: 2018-06-14T18:09:51Z
[INFO] Final Memory: 25M/67M
[INFO] -----
--
root@834f56ce1e48:/vertx-sample#
```

NOTA: Se puede tener errores en el código en este caso en lugar de un **BUILD SUCCESS** será un **BUILD FAILURE**

A continuación en la carpeta target verificamos que exista el jar generado **sample-1.0-SNAPSHOT-fat.jar**

```
root@834f56ce1e48: /vertx-sample/target
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@834f56ce1e48:/vertx-sample# cd target/
root@834f56ce1e48:/vertx-sample/target# ls
classes                maven-archiver        test-classes
generated-sources      sample-1.0-SNAPSHOT-fat.jar
generated-test-sources sample-1.0-SNAPSHOT.jar
root@834f56ce1e48:/vertx-sample/target# ll
total 7876
drwxr-xr-x 7 root root    4096 jun 14 18:09 ./
drwxr-xr-x 6 root root    4096 jun 14 18:09 ../
drwxr-xr-x 3 root root    4096 jun 14 18:09 classes/
drwxr-xr-x 3 root root    4096 jun 14 18:09 generated-sources/
drwxr-xr-x 3 root root    4096 jun 14 18:09 generated-test-sources/
drwxr-xr-x 2 root root    4096 jun 14 18:09 maven-archiver/
-rw-r--r-- 1 root root 8027345 jun 14 18:09 sample-1.0-SNAPSHOT-fat.jar
-rw-r--r-- 1 root root  6841 jun 14 18:09 sample-1.0-SNAPSHOT.jar
drwxr-xr-x 3 root root    4096 jun 14 18:09 test-classes/
root@834f56ce1e48:/vertx-sample/target#
```

CREACIÓN DE DOCKERFILE CON ENTRYPOINT

Con ayuda de una imagen vamos a ejecutar la vertical, para esto empezamos creando un Dockerfile, que tendrá los comandos necesarios para tener el archivo .jar y .sh

En el contenedor creado, en nuestro caso tendremos una carpeta llamada **rancher** donde tendremos los siguientes archivos:

```
gustavo@serverMike: ~/Documents/rancher
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
gustavo@serverMike:~/Documents/rancher$ pwd
/home/gustavo/Documents/rancher
gustavo@serverMike:~/Documents/rancher$ ll
total 7856
drwxr-xr-x 2 gustavo gustavo    4096 Jun 14 11:50 ./
drwxr-xr-x 4 gustavo gustavo    4096 Jun 14 10:44 ../
-rw-r--r-- 1 gustavo gustavo    121 Jun 12 17:22 Dockerfile
-rw-r--r-- 1 gustavo gustavo     55 Jun 12 16:33 java.sh
-rw-r--r-- 1 gustavo gustavo 8027345 Jun 14 11:09 sample-1.0-SNAPSHOT-fat.jar
gustavo@serverMike:~/Documents/rancher$
```

Damos el comando **docker images**

```
gustavo@serverMike: ~/Documents/rancher
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
gustavo@serverMike:~/Documents/rancher$ docker images
REPOSITORY          TAG          IMAGE ID          CREATED
SIZE
```

NOTA: En caso de que no existan imágenes recientes para uso de nuestro proyecto usamos **dgpecurso04/calculadora:v0.6** teniendo en cuenta que dicho contenedor tiene **java** instalado.

```
gustavo@serverMike: ~/Documents/rancher
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
gustavo@serverMike:~/Documents/rancher$ docker pull dgpecurso04/calculadora:v0.6
v0.6: Pulling from dgpecurso04/calculadora
72b39c1d4615: Pull complete
46a2d5ede4a6: Pull complete
d7caf6e91ad4: Pull complete
c7ac9f284354: Pull complete
a3ed95caeb02: Pull complete
bc46a0d5ee71: Pull complete
7fbaf168dc95: Pull complete
1b6f5bd66616: Pull complete
1da7dea888ba: Pull complete
aee217abbb32: Pull complete
Digest: sha256:b7d4ad5a14eae7eba07cc5f496e4feee1ea6ca9d4d852d1ea612d3b83dbf4228
Status: Downloaded newer image for dgpecurso04/calculadora:v0.6
gustavo@serverMike:~/Documents/rancher$ docker images
REPOSITORY              TAG                IMAGE ID            CREATED
dgpecurso04/calculadora v0.6              63e9a1a44f82       42 hours ago
o                        550 MB
```

Para la creación del Dockerfile agregamos los siguientes comandos:

nano Dockerfile

```
gustavo@serverMike: ~/Documents/rancher
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
gustavo@serverMike:~/Documents/rancher$ nano Dockerfile
```

Y agregamos las líneas:

```
gustavo@serverMike: ~/Documents/rancher
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
GNU nano 2.5.3      Archivo: Dockerfile      Modificado
FROM dgpecurso04/calculadora:v0.6
COPY java.sh /home
COPY sample-1.0-SNAPSHOT-fat.jar /home
ENTRYPOINT ["./home/java.sh"]

^G Ver ayuda  ^O Guardar  ^W Buscar  ^K Cortar Tex  ^J Justificar  ^C Posición
^X Salir      ^R Leer fich.  ^\ Reemplazar  ^U Pegar txt  ^T Ortografía  ^_ Ir a línea
```

Lo que hacemos es basarnos en un contenedor ya hecho que es **dgpecurso04/calculadora:v0.6** y copiamos los archivos **java.sh** y **sample-1.0-SNAPSHOT-fat.jar** a la carpeta **/home** del contenedor a crear.

CREACIÓN DEL SH Y DAR PERMISOS DE ACCESO AL SH

El ENTRYPOINT ejecutará el archivo **java.sh** que estará en la carpeta **/home** del contenedor a crear, para esto si el archivo no está creado damos el siguiente comando

```
gustavo@serverMike: ~/Documents/rancher
Archivo Editar Ver Buscar Terminal Ayuda
gustavo@serverMike:~/Documents/rancher$ nano java.sh
```

Añadimos las líneas:

```
gustavo@serverMike: ~/Documents/rancher
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.5.3 Archivo: java.sh
#!/bin/bash
java -jar home/sample-1.0-SNAPSHOT-fat.jar

[ 2 líneas leídas ]
^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar Tex ^J Justificar ^C Posición
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar txt ^T Corrector ^_ Ir a línea
```

Ahora damos los permisos necesarios a ese archivo **.sh** con la siguiente línea

```
gustavo@serverMike: ~/Documents/rancher
Archivo Editar Ver Buscar Terminal Ayuda
gustavo@serverMike:~/Documents/rancher$ sudo chmod 777 java.sh
gustavo@serverMike:~/Documents/rancher$
```

CONSTRUCCIÓN DE LA IMAGEN CON EL ARCHIVO SH Y JAR

Ahora construiremos la imagen con el siguiente comando:

```
sudo docker build -t dgpecurso04/calculadora:v0.7 .
```

```
gustavo@serverMike: ~/Documents/rancher
Archivo Editar Ver Buscar Terminal Ayuda
gustavo@serverMike:~/Documents/rancher$ sudo docker build -t dgpecurso04/calculadora:v0.7 .
Sending build context to Docker daemon 8.031 MB
Step 1/4 : FROM dgpecurso04/calculadora:v0.6
---> 63e9a1a44f82
Step 2/4 : COPY java.sh /home
---> 5d60d8b8ce01
Removing intermediate container c3fe4d0e235c
Step 3/4 : COPY sample-1.0-SNAPSHOT-fat.jar /home
---> 432a35425dd0
Removing intermediate container de238c38d279
Step 4/4 : ENTRYPOINT /home/java.sh
---> Running in bf07c1e0c1af
---> 7235540cb844
Removing intermediate container bf07c1e0c1af
Successfully built 7235540cb844
gustavo@serverMike:~/Documents/rancher$ █
```

El comando se compone de:

sudo docker build -t dgpecurso04/calculadora:v0.7 .

Donde **-t** es el tag que puede ser opcional que sería **v0.7**

dgpecurso04 es el usuario en Docker Hub

calculadora es el nombre del proyecto

Si damos el comando **docker images** veremos nuestro contenedor con el tag (v0.7)

```
gustavo@serverMike: ~/Documents/rancher
Archivo Editar Ver Buscar Terminal Ayuda
gustavo@serverMike:~/Documents/rancher$ docker images
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
dgpecurso04/calculadora v0.7        7235540cb844     6 minutes ago   558 MB
dgpecurso04/calculadora v0.6        63e9a1a44f82     43 hours ago    550 MB
gustavo@serverMike:~/Documents/rancher$ █
```

EJECUCIÓN DE LA VERTICAL DE MANERA LOCAL

Antes de subir nuestro proyecto a Docker Hub es necesario realizar una prueba local de la siguiente manera:

Comando:

docker run -d -p 8081:8080 -e PBA=pruebaServer1 dgpecurso04/calculadora:v0.7


```
gustavo@serverMike: ~/Documents/rancher
Archivo Editar Ver Buscar Terminal Ayuda

gustavo@serverMike:~/Documents/rancher$ docker run -d -p 8081:8080 -e PBA=pruebaServer1 dgpecurso04/calculadora:v0.7
e163f82d8f04216509ea661c8ef3cf69a8a3ea4a79823876a454063007801832
gustavo@serverMike:~/Documents/rancher$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
e163f82d8f04        dgpecurso04/calculadora:v0.7  "/home/java.sh"    8 seconds ago      Up 6 seconds       0.0.0.0:8081->8080/tcp   reverent_hugle
gustavo@serverMike:~/Documents/rancher$
```

Accedemos a las siguientes URL:

<http://localhost:8081/api/suma?numA=98&numB=5>

<http://localhost:8081/api/primero?mode=abs>

```
Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:8081/api/suma?numA=98&numB=5 localhost:8081/api/primero?mode=abs
localhost:8081/api/suma?numA=98&numB=5
JSON Raw Data Headers
Save Copy Filter JSON
servidor: "172.17.0.2"
resultado: 103
numA: 98
operacion: "SUMA"
numB: 5
```

```
Mozilla Firefox
localhost:8081/api/suma?numA=98&numB=5 localhost:8081/api/primero?mode=abs
localhost:8081/api/primero?mode=abs
JSON Raw Data Headers
Save Copy Filter JSON
variable: "pruebaServer1"
decoded: "abs"
autos:
  primero: "Ferrari"
  segundo: "Lamborghini"
  tercero: "Bugatti"
  nombre: "gustavo"
  edad: "21"
```

SUBIR IMAGEN DEL SERVICIO A DOCKERHUB

Ya que se logró la ejecución del contenedor en dicho puerto lo siguiente es subir el archivo a Docker Hub, para esto tenemos que ingresar nuestro usuario de la siguiente forma:

Comando:

docker login

```
gustavo@serverMike: ~/Documents/rancher
Archivo Editar Ver Buscar Terminal Ayuda

gustavo@serverMike:~/Documents/rancher$ docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head
over to https://hub.docker.com to create one.
Username: dgpecurso04
Password:
Login Succeeded
gustavo@serverMike:~/Documents/rancher$
```

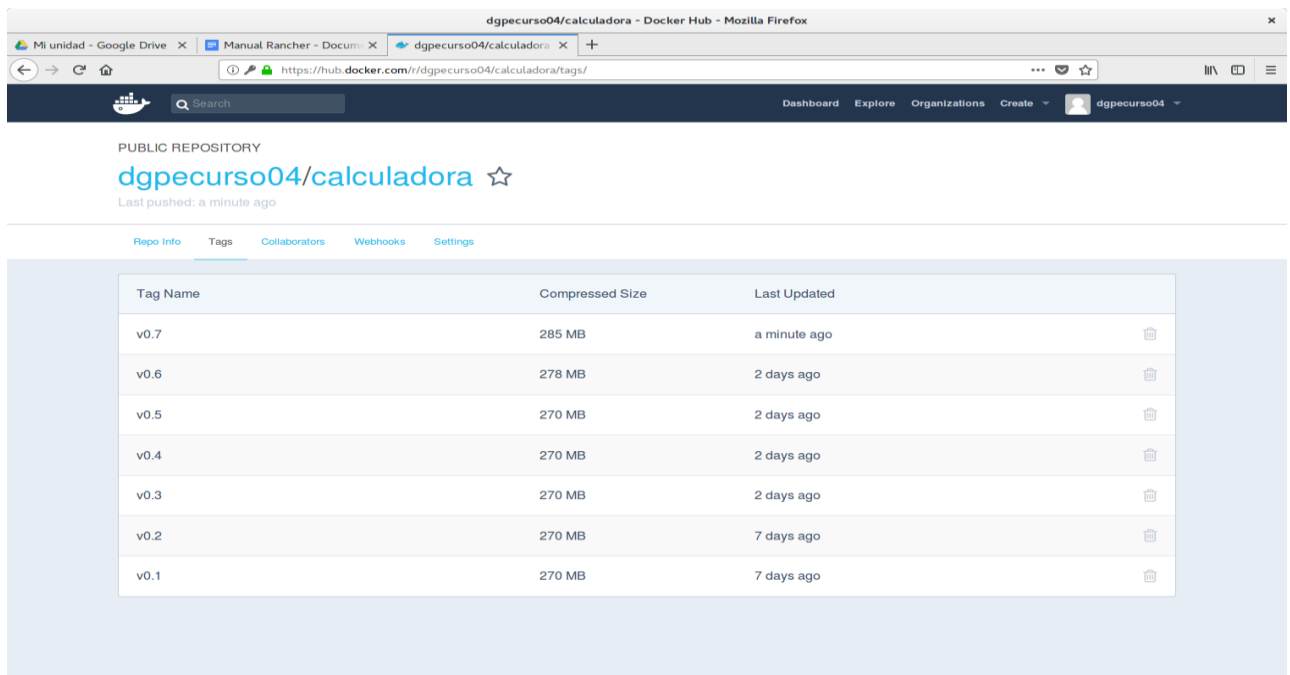
Subimos la imagen con el siguiente comando:

```
docker push dgpecurso04/calculadora:v0.7
```

```
gustavo@serverMike: ~/Documents/rancher
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
gustavo@serverMike:~/Documents/rancher$ docker push dgpecurso04/calculadora:v0.7
The push refers to a repository [docker.io/dgpecurso04/calculadora]
7ca9e58313dd: Pushed
9a871ce53f67: Pushed
baa1ca57ac01: Preparing
7ca9e58313dd: Pushing 5.801 MB/8.027 MB
7ca9e58313dd: Pushing 5.408 MB/8.027 MB
891ad5124517: Layer already exists
0dd2d2815662: Layer already exists
5f70bf18a086: Layer already exists
918dbf1cf3de: Layer already exists
9e1fe90ee292: Layer already exists
d97fd2c5d8e1: Layer already exists
c1cc34424286: Layer already exists
v0.7: digest: sha256:620a36141e2858b54e1875ed1f4bca877c04081140357f11b95833c8cd45e8cd size: 2823
gustavo@serverMike:~/Documents/rancher$
```

COMPROBACIÓN DE LA IMAGEN EN DOCKERHUB

Accedemos a Docker Hub con usuario y contraseña verificando que exista la imagen que creamos:



The screenshot shows the Docker Hub interface for the repository `dgpecurso04/calculadora`. The page displays a list of tags with their respective compressed sizes and last updated times. The tags range from `v0.1` to `v0.7`, with `v0.7` being the most recent, pushed a minute ago.

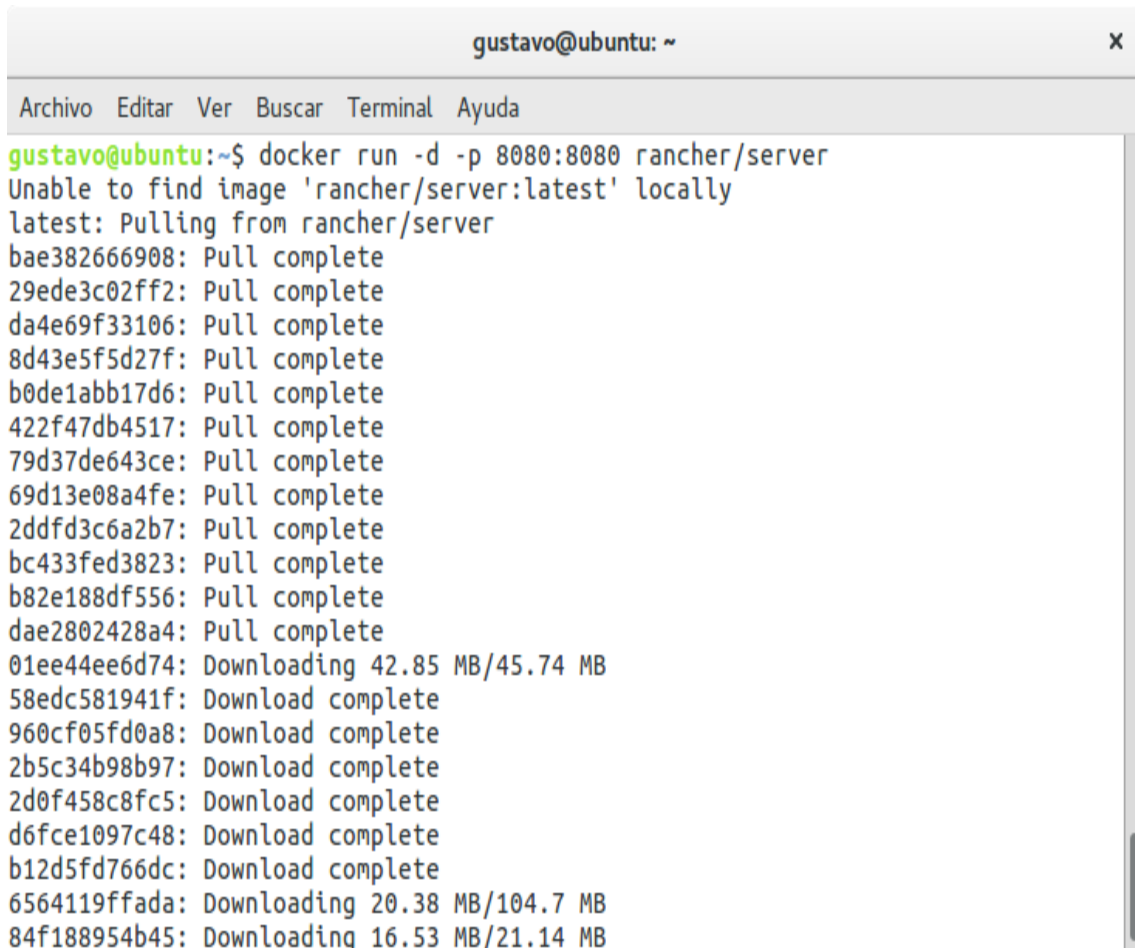
Tag Name	Compressed Size	Last Updated
v0.7	285 MB	a minute ago
v0.6	278 MB	2 days ago
v0.5	270 MB	2 days ago
v0.4	270 MB	2 days ago
v0.3	270 MB	2 days ago
v0.2	270 MB	7 days ago
v0.1	270 MB	7 days ago

PARTE 2

INSTALACIÓN Y CONFIGURACIÓN DE HOST RANCHER

Ejecutar el Rancher con el comando

```
docker run -d -p 8080:8080 rancher/server
```

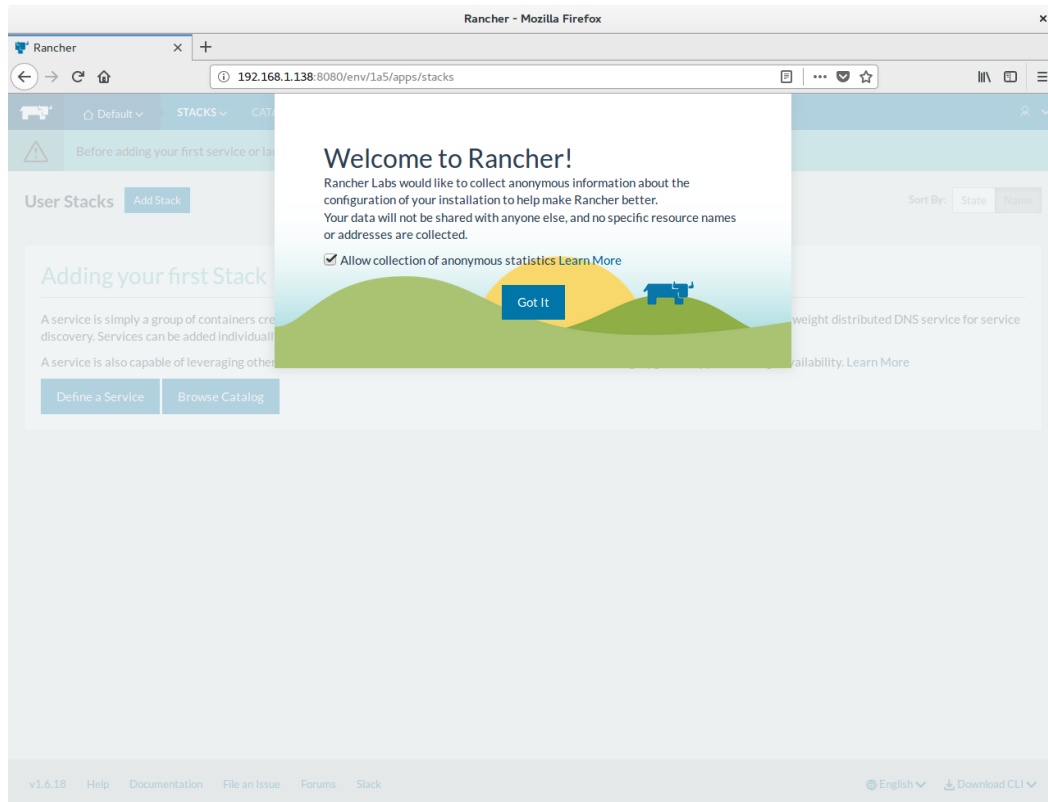


```
gustavo@ubuntu: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
gustavo@ubuntu:~$ docker run -d -p 8080:8080 rancher/server  
Unable to find image 'rancher/server:latest' locally  
latest: Pulling from rancher/server  
bae382666908: Pull complete  
29ede3c02ff2: Pull complete  
da4e69f33106: Pull complete  
8d43e5f5d27f: Pull complete  
b0de1abb17d6: Pull complete  
422f47db4517: Pull complete  
79d37de643ce: Pull complete  
69d13e08a4fe: Pull complete  
2ddfd3c6a2b7: Pull complete  
bc433fed3823: Pull complete  
b82e188df556: Pull complete  
dae2802428a4: Pull complete  
01ee44ee6d74: Downloading 42.85 MB/45.74 MB  
58edc581941f: Download complete  
960cf05fd0a8: Download complete  
2b5c34b98b97: Download complete  
2d0f458c8fc5: Download complete  
d6fce1097c48: Download complete  
b12d5fd766dc: Download complete  
6564119ffada: Downloading 20.38 MB/104.7 MB  
84f188954b45: Downloading 16.53 MB/21.14 MB
```

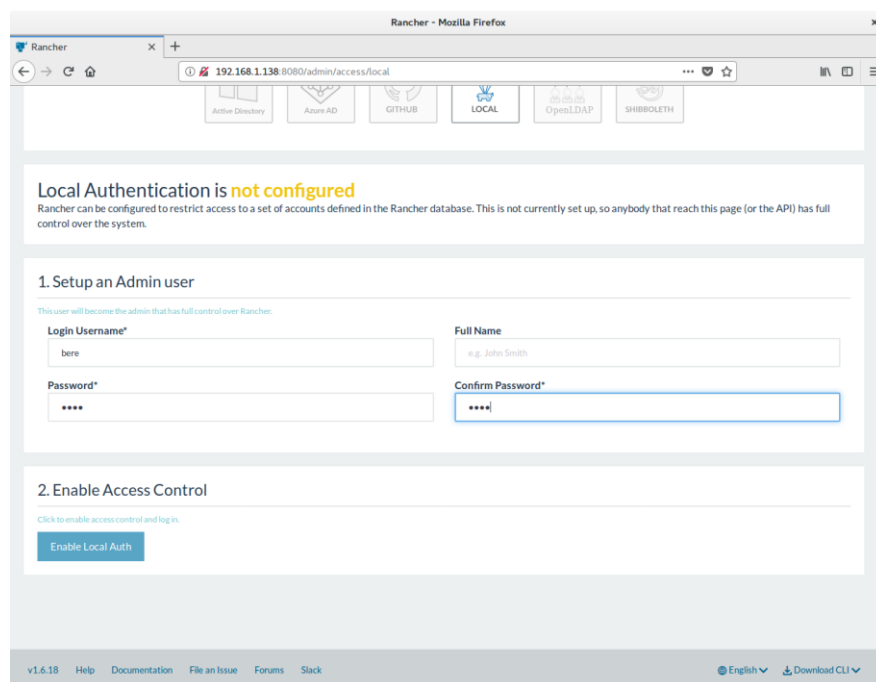
Para comprobar que esté corriendo Rancher:

En el navegador con la dirección compuesta por la dirección ip de la máquina host y el puerto destinado durante la instalación de Rancher, se muestra la pantalla de bienvenida.

En este caso es: 192.168.1.138:8080



En la pestaña de ADMIN y después seleccionando “LOCAL”
Configurar un nombre de usuario y contraseña.



En la pestaña “INFRASTRUCTURE” guardamos la ip de la máquina host como la determinada para usar la API de Rancher

Rancher - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Rancher Compartido conmigo - Go X Manual Rancher - Docum X +

192.168.1.138:8080/env/1a5/infra/hosts/add?driver=custom

Default STACKS CATALOG INFRASTRUCTURE ADMIN API

Before adding your first service or launching a container, you'll need to add a Linux host with a supported version of Docker. Add a host

Hosts: Add Host

Host Registration URL

What base URL should hosts use to connect to the Rancher API?

☒ This site's address:

☐ Something else:

Don't include `/v1` or any other path, but if you are doing **SSL termination** in front of Rancher, be sure to use `https://`.

Are you sure all the hosts you will create will be able to reach `http://192.168.1.138:8080`? It looks like a private IP or local network.

Save

v1.6.18 Help Documentation File an Issue Forums Slack

Al quedar guardada nos mostrará el comando que tendrán que ejecutar los agentes que deseen usar nuestro host.

Rancher - Mozilla Firefox

Rancher Compartido conmigo - Go X Manual Rancher - Docum X New Tab +

192.168.1.138:8080/env/1a5/infra/hosts/add?driver=custom

Custom AMAZON EC2 Azure DigitalOcean packet

Manage available machine drivers

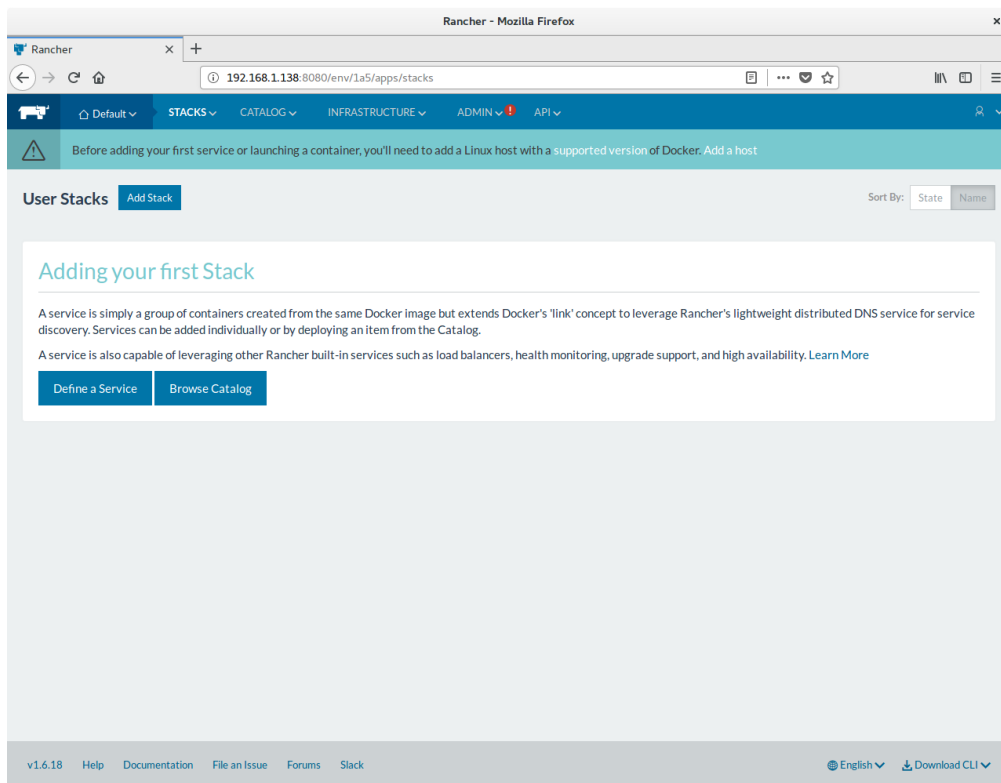
- 1 Start up a Linux machine somewhere and install a supported version of Docker on it.
- 2 Make sure any security groups or firewalls allow traffic:
 - From and To all other hosts on UDP ports 500 and 4500 (for IPsec networking)
- 3 Optional: Add labels to be applied to the host.
 - + Add Label
- 4 Specify the public IP that should be registered for this host. If left empty, Rancher will auto-detect the IP to use. This generally works for machines with unique public IPs, but will not work if the machine is behind a firewall/NAT or if it is the same machine that is running the `rancher/server` container.
- 5 Copy, paste, and run the command below to register the host with Rancher:

```
sudo docker run --rm --privileged -v /var/run/docker.sock:/var/run/docker.sock -v /var/lib/rancher:/var/lib/rancher rancher/agent:v1.2.10 http://192.168.1.138:8080/v1/scripts/93f7872958c4bf6d4a315146784000081v0212pck75w3jzdk8tMzkaVgQ
```
- 6 Click close below. The new host should pop up on the Hosts screen within a minute.

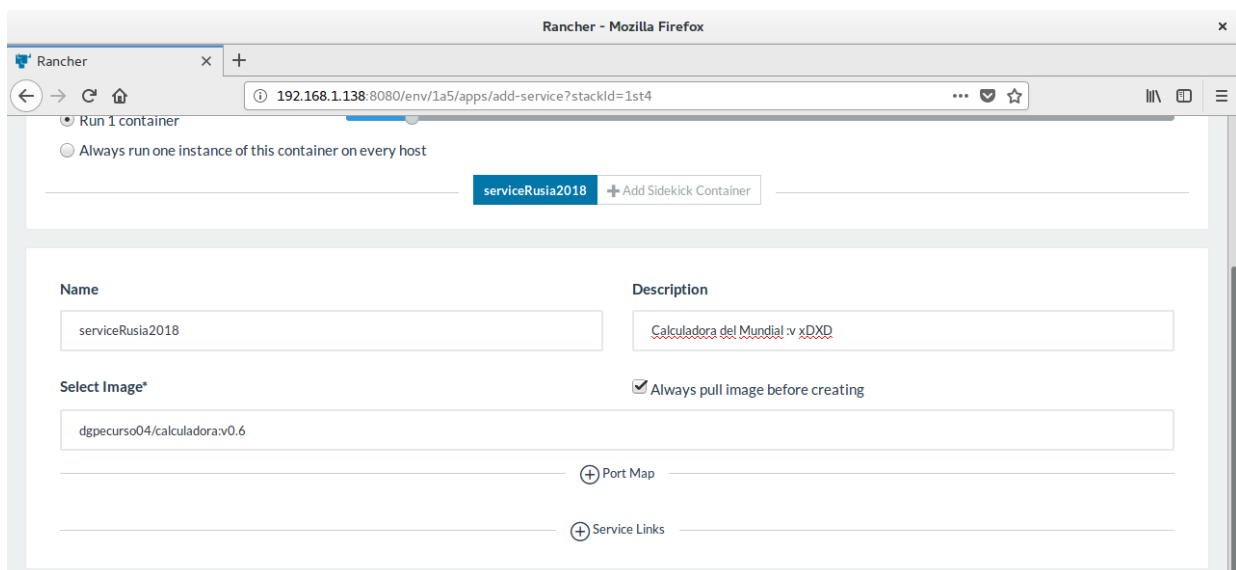
Close

AGREGAR SERVICIOS

Para agregar un servicio nos vamos a la pestaña “DEFAULT” y después en “Define a Service”



Le asignamos un nombre, una descripción y el nombre de la imagen que utilizaremos para dar el servicio, en este caso es dgpecurso04/calculadora:v0.6



Seleccionamos en “Console” la opción “None” ya que no tendrá una vista interactiva.

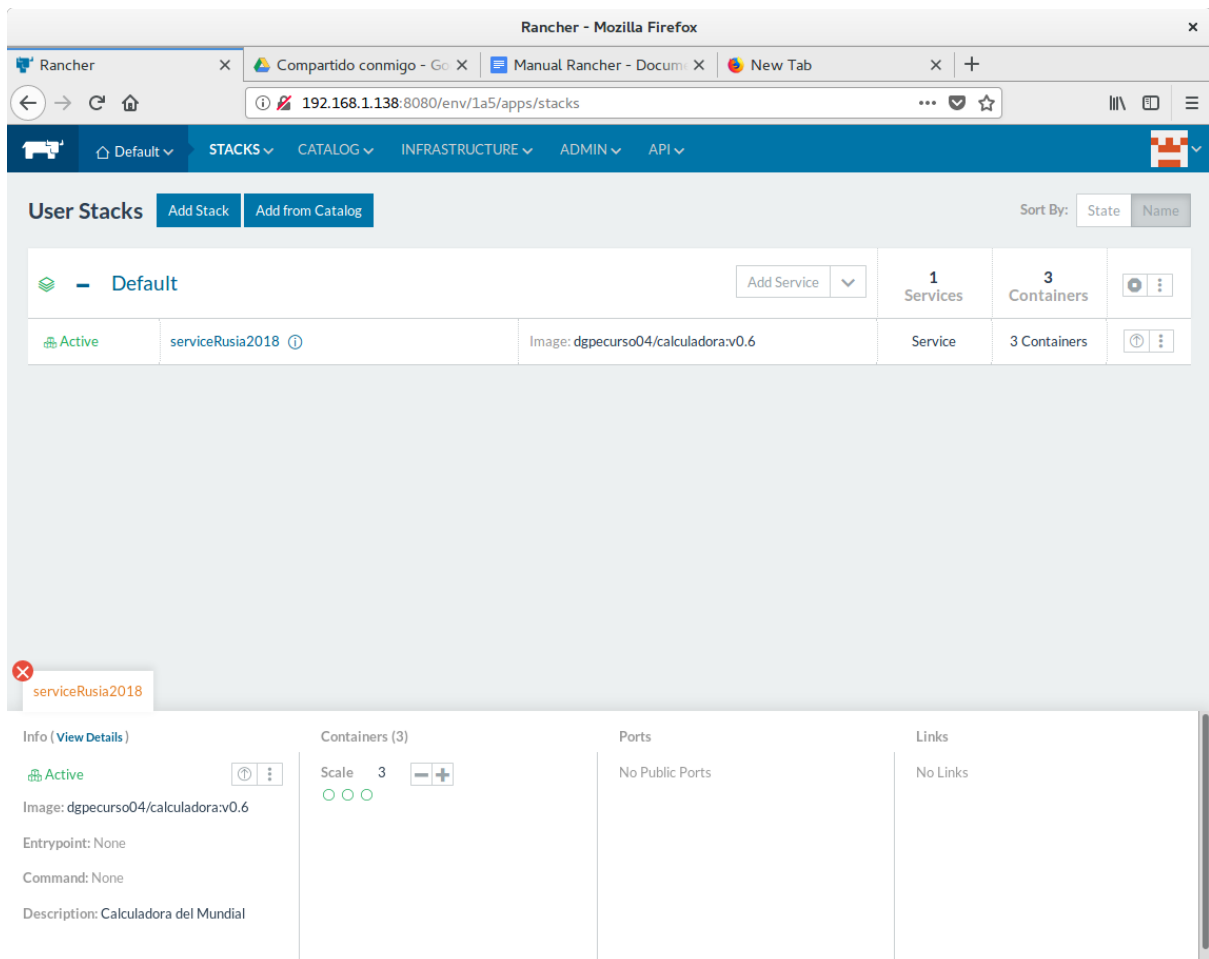
The screenshot shows the 'Add Service' form in the Rancher console, specifically the 'Security/Host' tab. The form is titled 'Rancher - Mozilla Firefox' and the URL is '192.168.1.138:8080/env/1a5/apps/add-service?stackId=1st5'. The form includes sections for 'Port Map' and 'Service Links' at the top. Below these are tabs for 'Command', 'Volumes', 'Networking', 'Security/Host' (selected), 'Secrets', 'Health Check', 'Labels', and 'Scheduling'. The 'Security/Host' tab contains various configuration options: 'Privileged' (checkbox for 'Full access to the host'), 'PID Mode' (checkbox for 'Host'), 'Memory Limit' (Unlimited MB), '+ Swap Limit' (Unlimited MB), 'Memory Reservation' (Unlimited MB), 'mCPU Reservation' (empty field), 'CPU Pinning' (e.g. 0,3; Default: All), 'Shares' (e.g. 1024), 'Capabilities' (Add: None, Drop: None), 'Device Binding' (+ Add Binding), 'Log Driver' (json-file), and 'Log Options' (+ Add Option). At the bottom are 'Create' and 'Cancel' buttons.

Presionamos el botón “Create”

Veremos cómo se inicializa y activa el servicio

The screenshot shows the Rancher console after the service has been created. The URL is '192.168.1.138:8080/env/1a5/apps/stacks/1st5'. The console shows a message: 'Before adding your first service or launching a container, you'll need to add a Linux host with a supported version of Docker. Add a host'. Below this is a 'Stack' section with a dropdown set to 'Default' and an 'Add Service' button. The service 'serviceRusia2018' is shown in progress, with the image 'dgpecurso04/calculadora:v0.6'. The status is 'Activating' and there are '0 Containers'.

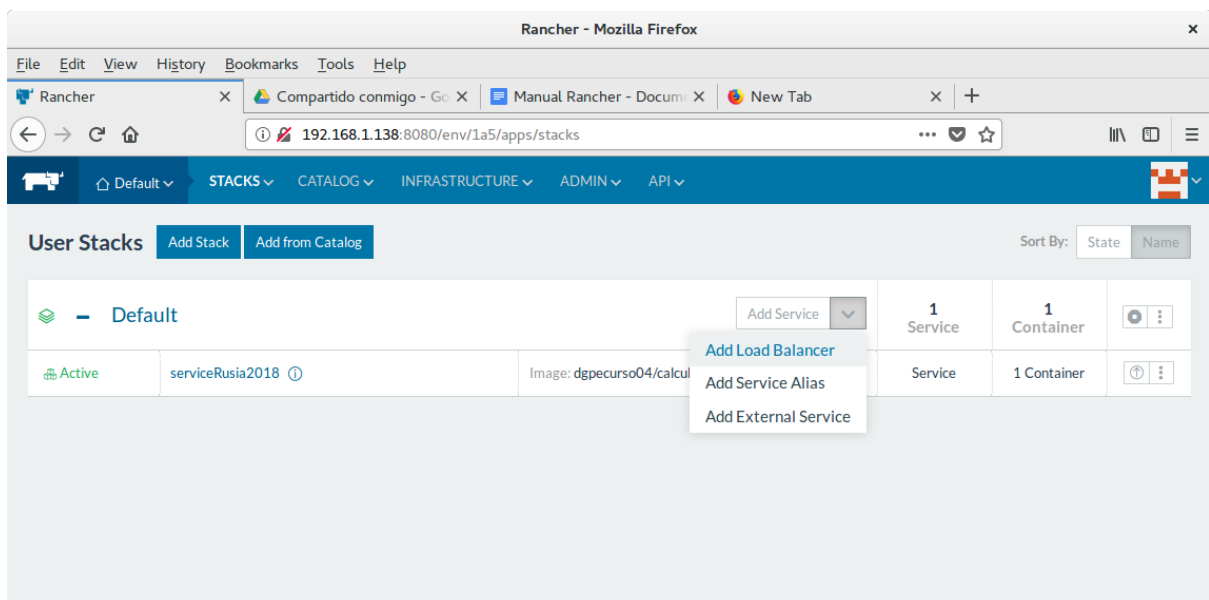
Podemos aumentar el número de contenedores que replicarán el servicio



The screenshot shows the Rancher web interface in a Mozilla Firefox browser. The address bar displays the URL `192.168.1.138:8080/env/1a5/apps/stacks`. The navigation bar includes tabs for Default, STACKS, CATALOG, INFRASTRUCTURE, ADMIN, and API. The main content area is titled 'User Stacks' and features buttons for 'Add Stack' and 'Add from Catalog'. A table lists the stacks, with 'Default' selected. Below the table, the details for the 'serviceRusia2018' service are shown. The service is in an 'Active' state, using the image 'dgpecurso04/calculadora:v0.6'. The 'Containers (3)' section shows a scale of 3 with three green circles. The 'Ports' section indicates 'No Public Ports', and the 'Links' section indicates 'No Links'. A dropdown menu for 'Add Service' is open, showing options: 'Add Load Balancer', 'Add Service Alias', and 'Add External Service'.

CREACIÓN DEL BALANCEADOR

Para crear el balanceador le damos clic en “Add Service” y después “Add Load Balancer”



This screenshot shows the Rancher web interface with the 'Add Service' dropdown menu open. The service 'serviceRusia2018' is selected, and the 'Add Load Balancer' option is highlighted. The service is currently scaled to 1 container. The 'Ports' section indicates 'No Public Ports', and the 'Links' section indicates 'No Links'.

Le asignamos nombre y descripción. El número de puerto en el que se alojará el balanceador, en este caso es el 10001 y el número de puerto en donde se encuentra el host que es el 8080.

Le damos clic en “Create”

Rancher - Mozilla Firefox

192.168.1.138:8080/env/1a5/apps/add-balancer?stackId=1st5

Add Load Balancer

Scale

☒ Run 1 container

☐ Always run one instance of this container on every host

Name

Rusia2018-lb

Description

Balanceador del Mundial de Rusia 2018

Port Rules

+ Add Service Rule + Add Selector Rule

Access*	Protocol*	Request Host	Port*	Path	Target*	Port*
Public	HTTP	e.g. example.com	10001	e.g. /foo	Default/serviceRusia2	8080

Host and Path rules are matched top-to-bottom in the order shown. Backends will be named randomly by default; to customize the generated backends, provide a name and then refer to that in the custom haproxy.cfg. Show custom backend names. Show host IP address options.

SSL Termination Stickiness Custom haproxy.cfg Labels Scheduling

There are no SSL/TLS ports configured.

Create Cancel

Cuando los agentes se conectan al Host, podemos visualizar como los asigna.

Rancher - Mozilla Firefox

192.168.1.138:8080/env/1a5/apps/stacks/1st5/services/1s9/containers

Service: serviceRusia2018 in Default

Degraded (Failed to acquire lock [SERVICE.9])

Description: Calculadora del Mundial

Type: Service

Scale: 5

Image: dgpecurso04/calculadora:v0.6

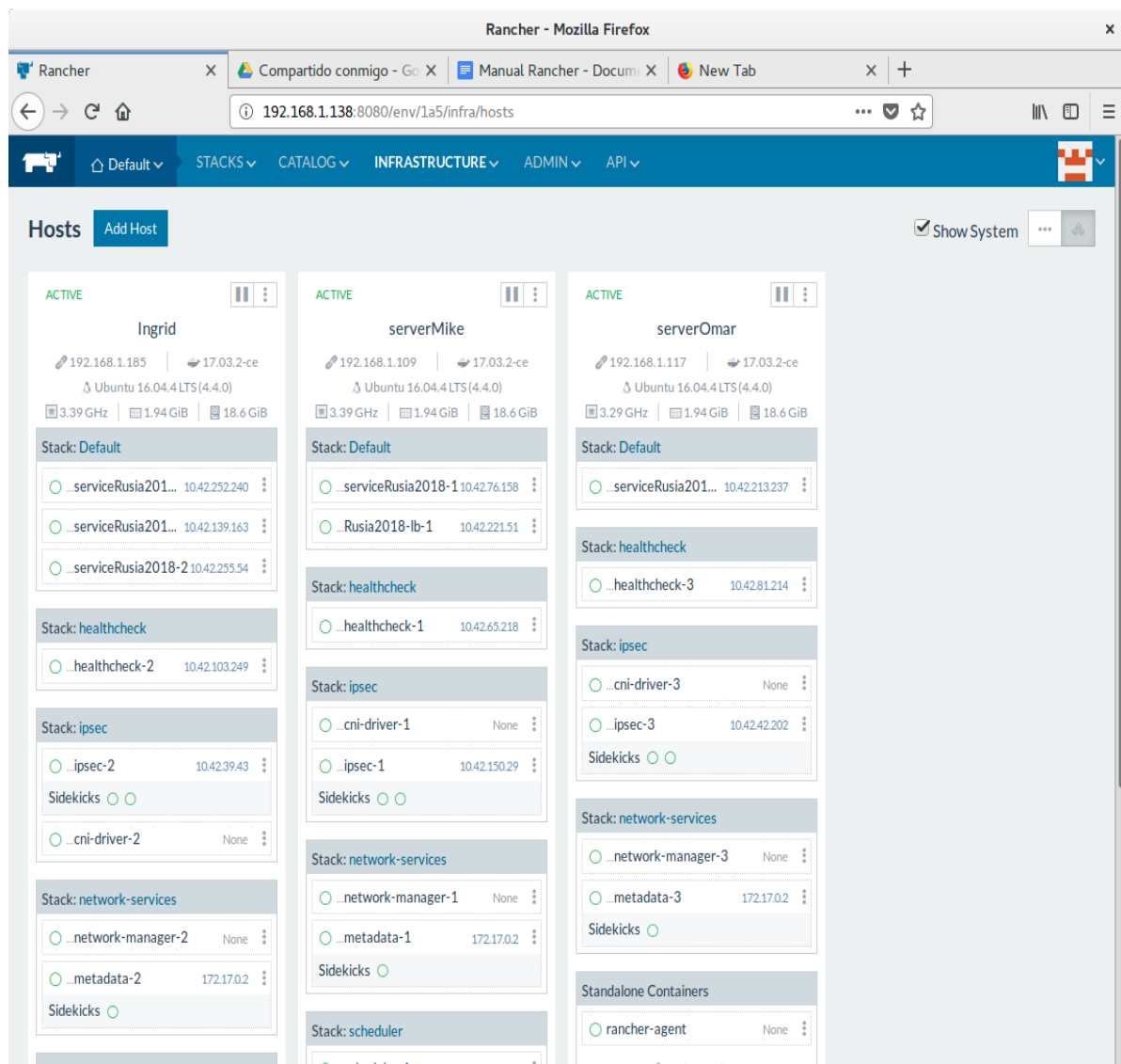
Entrypoint: None

Command: None

State	Name	IP Address	Host	Image	Stats
Running	Default-servi...	10.42.76.158	serverMike	dgpecurso04/cal	
Running	Default-servi...	10.42.213.237	serverOmar	dgpecurso04/cal	n/a
Starting	Default-servi...	None	Ingrid	dgpecurso04/cal	n/a

En "HOSTS" se reflejan la información de los agentes conectados al host, como el nombre de su computadora, su dirección IP, y los servicios que tiene asignados.

Como podemos observar solo a uno de ellos se le asigna el balanceador.

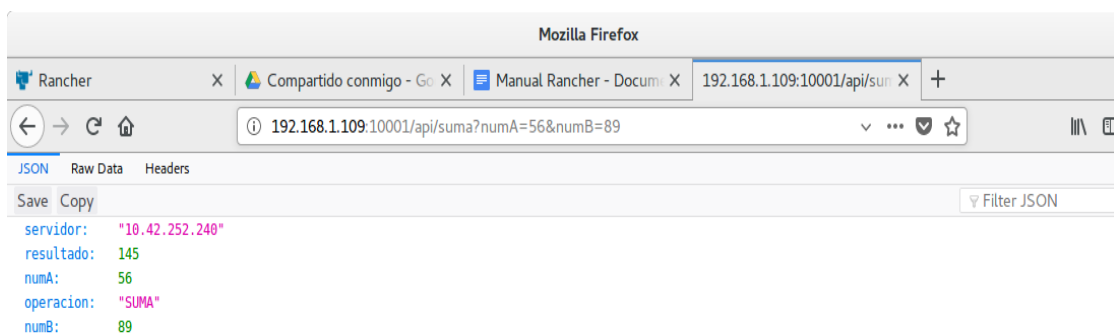
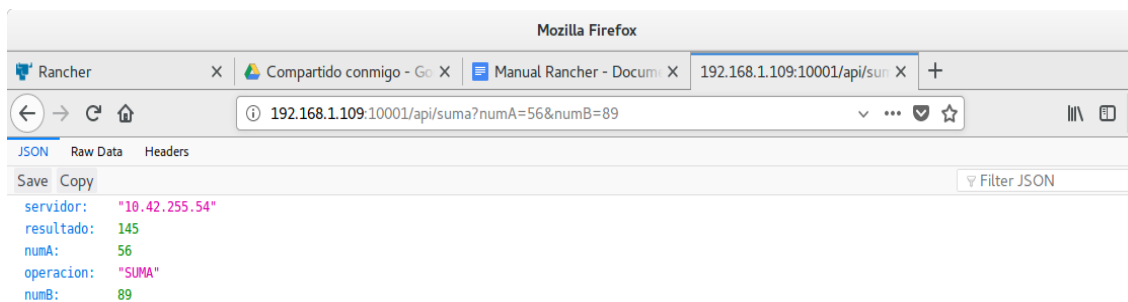
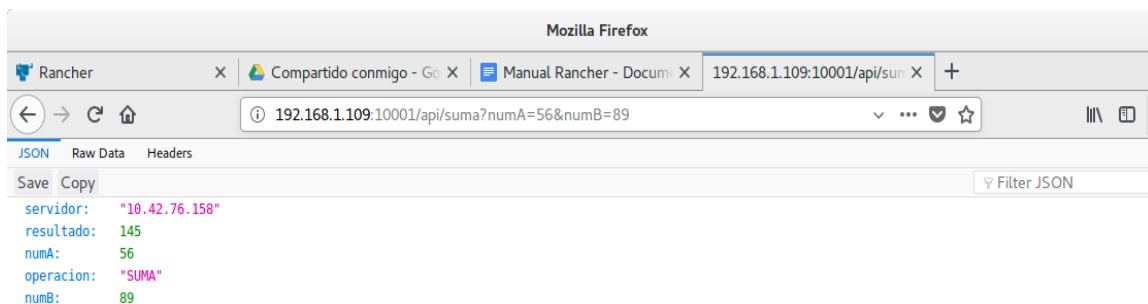
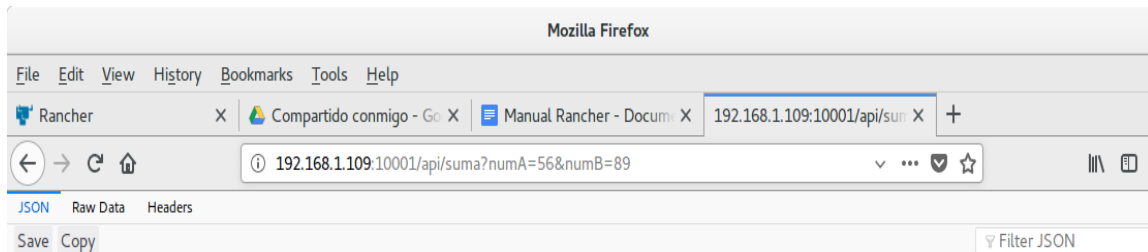


FUNCIONAMIENTO DEL BALANCEADOR

Para probar y ver el funcionamiento del balanceador, en el navegador mandamos llamar el servicio GET, usando el puerto asignado al balanceador.

192.168.1.109:10001/api/suma?numA=56&numB=89

Al actualizar en repetidas ocasiones el navegador observaremos en “servidor” como cambiar la dirección ip desde la que se da respuesta a nuestra petición, así comprobamos que el balanceador funciona.

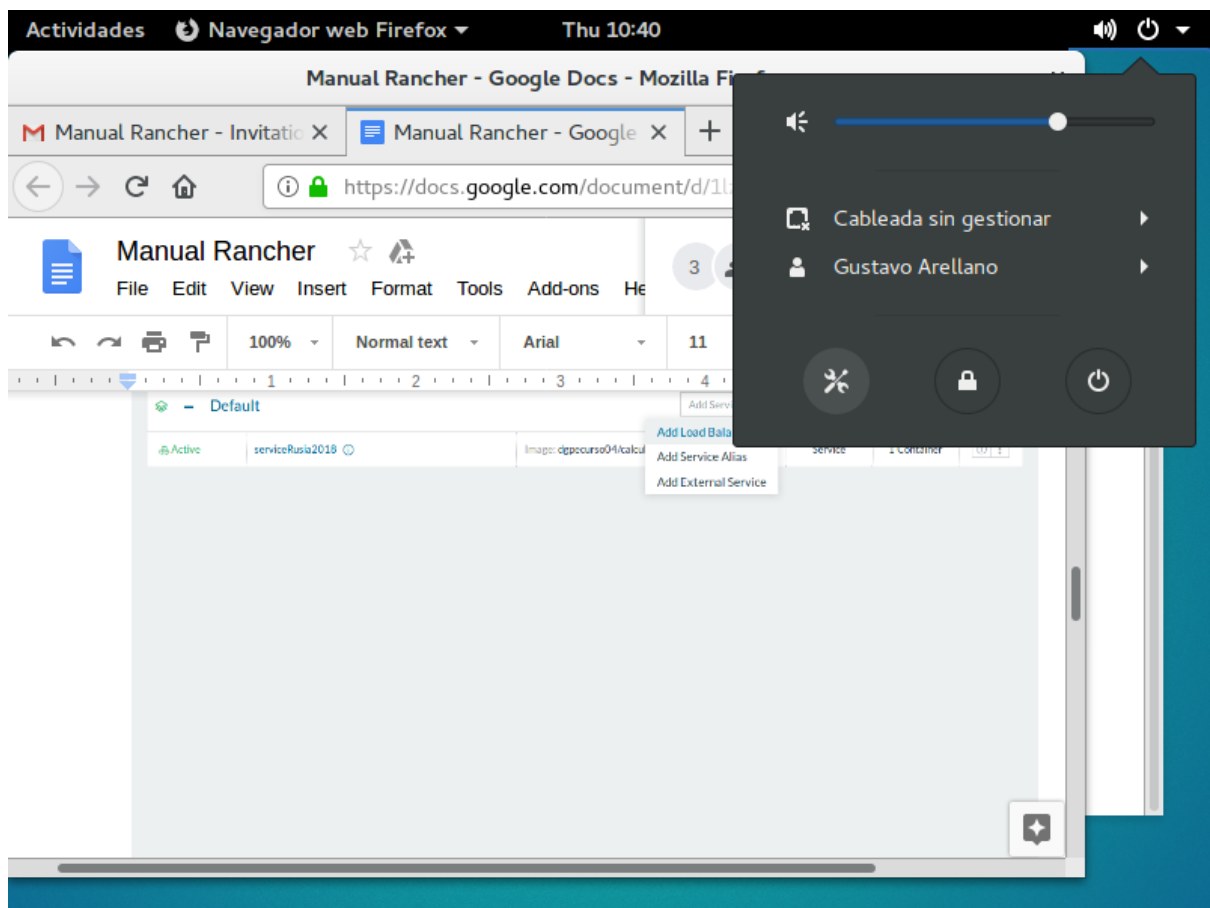


PARTE 3

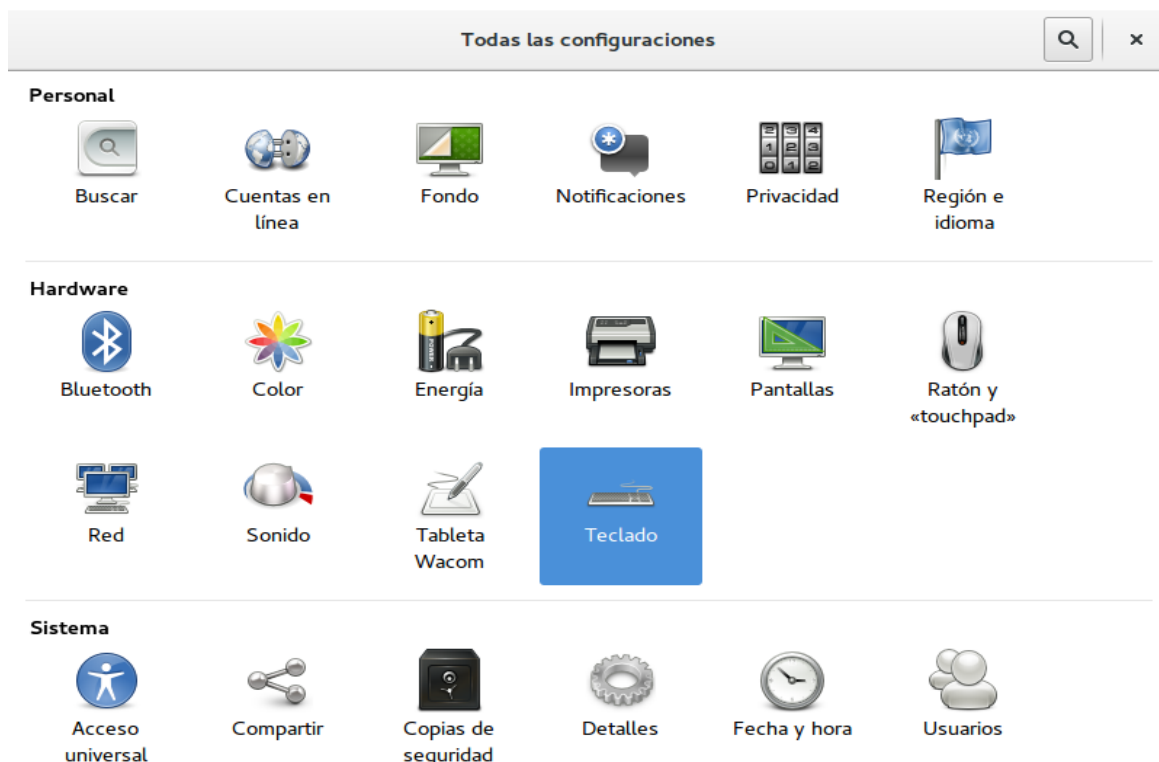
LOS AGENTES DE RANCHER

OPCIONAL: Sobre la imagen ub que compartió el profesor se puede instalar un entorno gráfico GNOME y cambiar la configuración del teclado a Español (México)

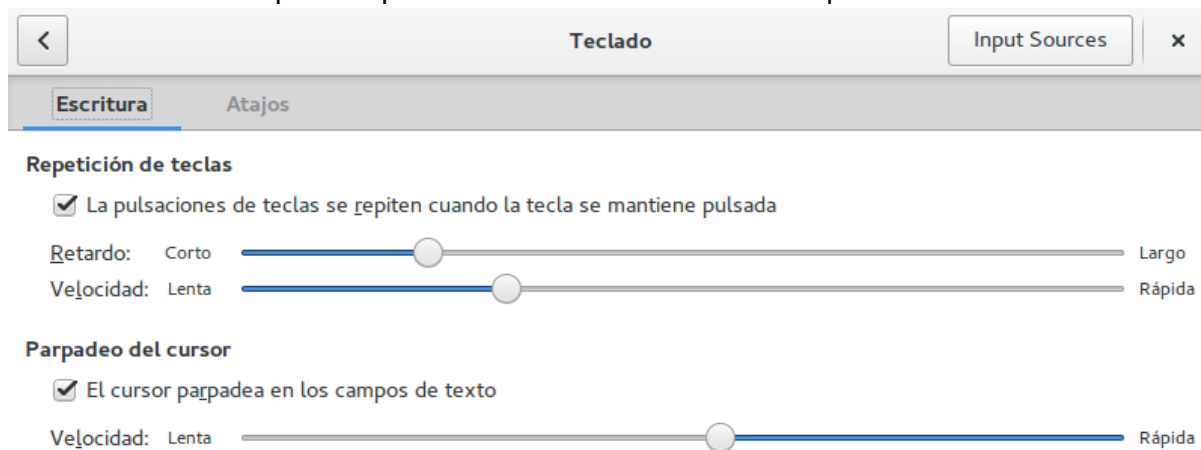
1. Para instalar el ambiente gráfico GNOME se usan los siguientes comandos:
 - 1.1. `sudo apt-get update`
 - 1.2. `sudo apt-get install console-data`
 - 1.3. `sudo apt-get install ubuntu-gnome-desktop`
2. Una vez instalado GNOME, para cambiar la configuración del teclado:
 - 2.1. En el extremo superior izquierdo abrir las opciones de configuración



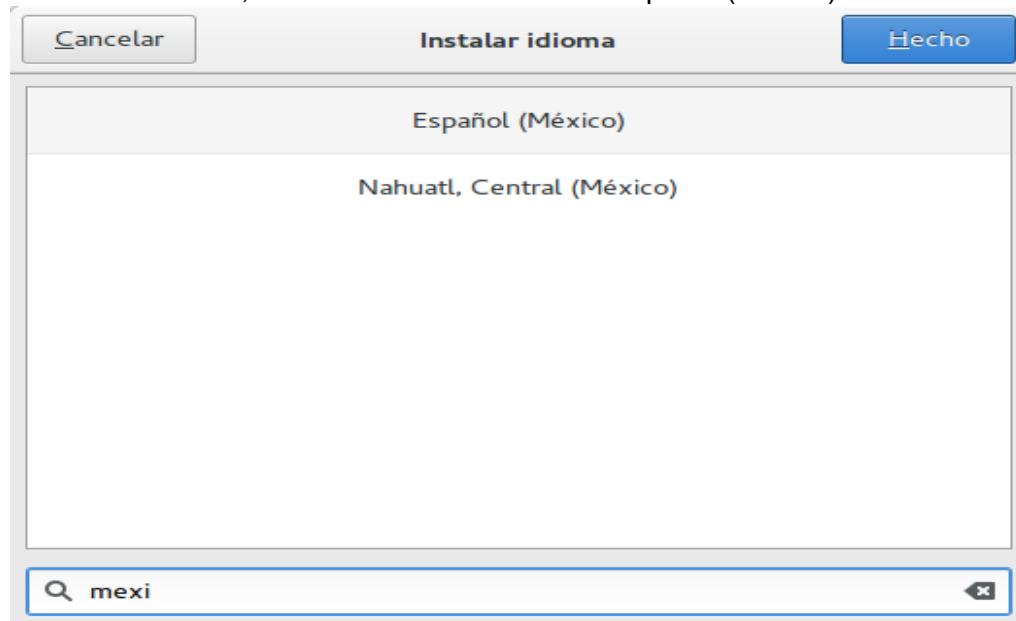
2.2. En la sección de Hardware seleccionar “Teclado”



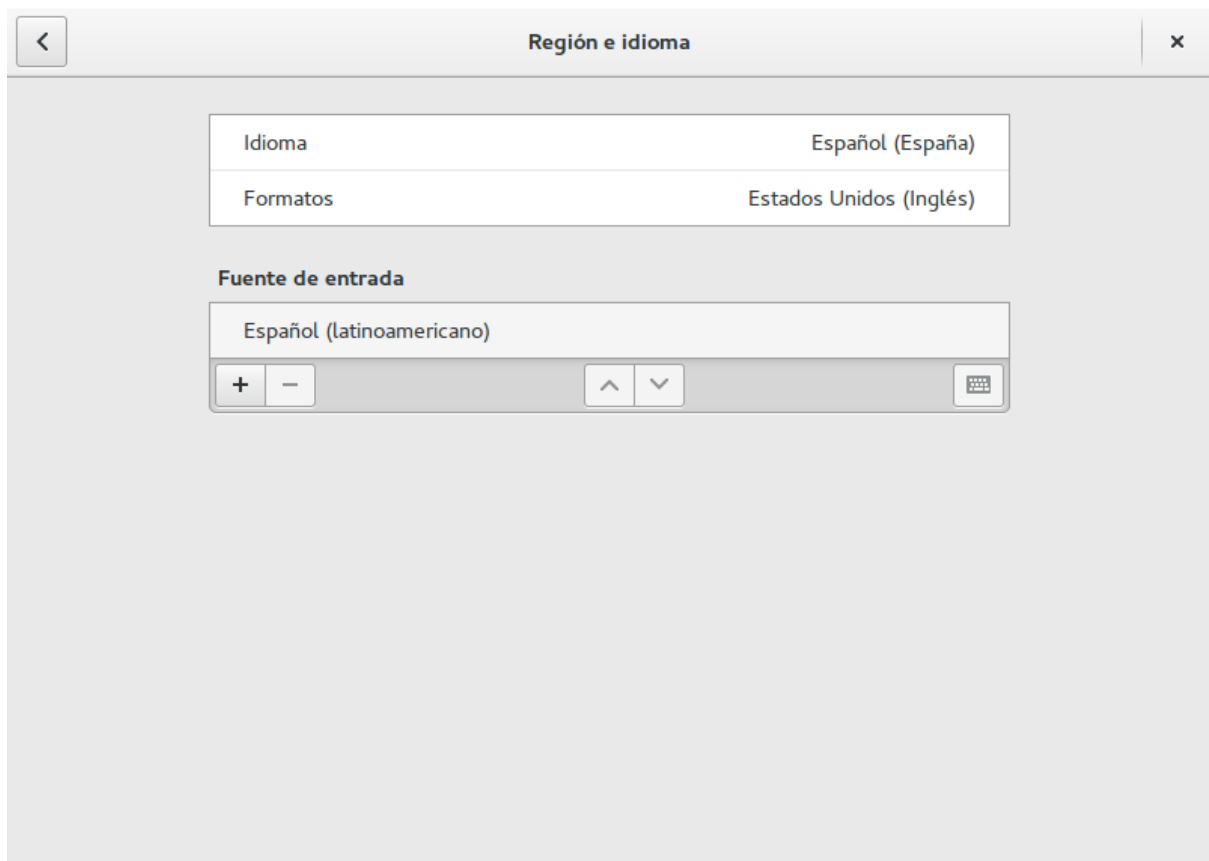
2.3. En la parte superior derecha dar clic al botón “Input Sources”



2.4. En Idioma, dar clic e instalar el idioma Español (México)



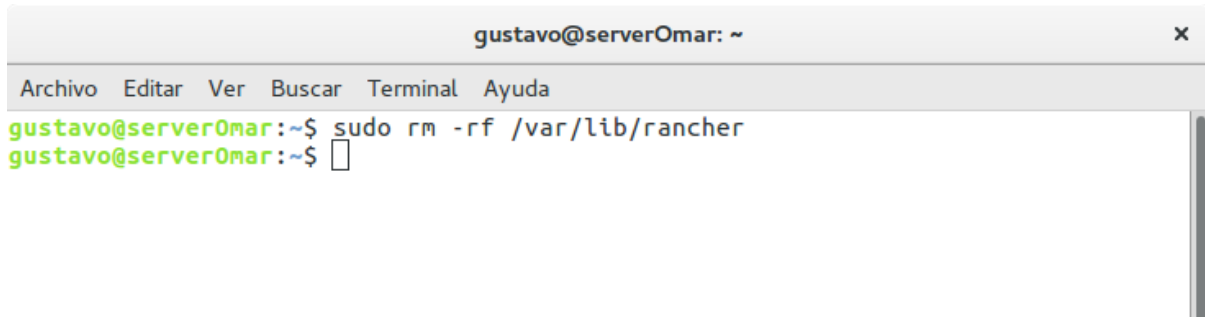
2.5. Una vez instalado cambiar la fuente de entrada a Español (latinoamericano)



CAMBIAR NOMBRE DE USUARIO A LA COMPUTADORA

1. Borrar la carpeta /var/lib/rancher

sudo rm -rf /var/lib/rancher



```
gustavo@serverOmar: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
gustavo@serverOmar:~$ sudo rm -rf /var/lib/rancher  
gustavo@serverOmar:~$
```

2. Cambiar el nombre del host

Ejecutar los comandos:

sudo nano /etc/hosts

sudo nano /etc/hostname



```
gustavo@serverOmar: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
f5051184bc6c  
c423f5deb6ab  
331b53995b45  
35d9f689282c  
b271be583e9b  
gustavo@serverOmar:~$ docker rm $(sudo docker ps -aq)  
3f4e2a39958c  
2ff79114fb7a  
9f2410186d22  
b047d5b8481a  
cf4ead344532  
dcc5c66f7d39  
f5051184bc6c  
c423f5deb6ab  
331b53995b45  
35d9f689282c  
b271be583e9b  
gustavo@serverOmar:~$  
gustavo@serverOmar:~$ sudo nano /etc/hosts  
gustavo@serverOmar:~$  
gustavo@serverOmar:~$ sudo nano /etc/hostname
```

Modificar en el archivo

/etc/hosts de ubuntu a serverNOMBRE:

```
gustavo@serverOmar: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.5.3 Archivo: /etc/hosts

127.0.0.1    localhost
127.0.1.1    serverOmar

# The following lines are desirable for IPv6 capable hosts
::1        localhost ip6-localhost ip6-loopback
ff02::1    ip6-allnodes
ff02::2    ip6-allrouters

[ 7 líneas leídas ]
^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar Tex ^J Justificar ^C Posición
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar txt ^T Ortografía ^_ Ir a línea
```

Modificar en el archivo

/etc/hostname de ubuntu a serverNOMBRE:

```
gustavo@serverOmar: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.5.3 Archivo: /etc/hostname

serverOmar

[ 1 línea leída ]
^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar Tex ^J Justificar ^C Posición
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar txt ^T Ortografía ^_ Ir a línea
```


3. Reiniciar para que se apliquen los cambios

sudo reboot

```
gustavo@server0mar: ~
Archivo Editar Ver Buscar Terminal Ayuda
331b53995b45
35d9f689282c
b271be583e9b
gustavo@server0mar:~$ docker rm $(sudo docker ps -aq)
3f4e2a39958c
2ff79114fb7a
9f2410186d22
b047d5b8481a
cf4ead344532
dcc5c66f7d39
f5051184bc6c
c423f5deb6ab
331b53995b45
35d9f689282c
b271be583e9b
gustavo@server0mar:~$
gustavo@server0mar:~$ sudo nano /etc/hosts
gustavo@server0mar:~$
gustavo@server0mar:~$ sudo nano /etc/hosts
gustavo@server0mar:~$ sudo nano /etc/hostname
gustavo@server0mar:~$ sudo reboot
```

CREAR Y CORRER EL AGENTE DE RANCHER

Con el comando dado por el Host Rancher:

```
sudo docker run --rm --privileged -v
/var/run/docker.sock:/var/run/docker.sock -v
/var/lib/rancher:/var/lib/rancher rancher/agent:v1.2.10
http://192.168.1.138:8080/v1/scripts/33F78729505C4BF6D4A3:1514678400000:Yu
zIZpCL7sWJJzdXBTMZkaVg6Q
```

```
gustavo@serverOmar: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
gustavo@serverOmar:~$ sudo docker run --rm --privileged -v /var/run/docker.sock:/var/run/docker.sock -v /var/lib/rancher:/var/lib/rancher rancher/agent:v1.2.10 http://192.168.1.138:8080/v1/scripts/33F78729505C4BF6D4A3:1514678400000:YuzIZpCL7sWJJzdXBTMZkaVg6Q
Unable to find image 'rancher/agent:v1.2.10' locally
v1.2.10: Pulling from rancher/agent
b3e1c725a85f: Already exists
6a710864a9fc: Already exists
d0ac3b234321: Already exists
87f567b5cf58: Already exists
063e24b217c4: Already exists
d0a3f58caef0: Already exists
16914729cfd3: Already exists
4956f3e025a2: Pull complete
64292afc18d7: Pull complete
Digest: sha256:5618f36cdb6c6fe25baa3e72977f3a226d43ffd1d74dda6a8860e32b73a1e171
Status: Downloaded newer image for rancher/agent:v1.2.10

INFO: Running Agent Registration Process, CATTLE_URL=http://192.168.1.138:8080/v1
INFO: Attempting to connect to: http://192.168.1.138:8080/v1
INFO: http://192.168.1.138:8080/v1 is accessible
INFO: Configured Host Registration URL info: CATTLE_URL=http://192.168.1.138:8080/v1 ENV_URL=http://192.168.1.138:8080/v1
```

INFORMACIÓN ADICIONAL

Comandos útiles:

Para detener y borrar todos los dockers

```
docker stop $(sudo docker ps -aq)
```

```
docker rm $(sudo docker ps -aq)
```