Nombre: José de Jesús Vázquez Medina

**SE CREA EL MICROSERVICIO QUE CALCULA EL NÚMERO DE DÍGITOS QUE TIENE EL FACTORIAL DE UN NÚMERO**

1. Se clona el proyecto **vertx-sample** del sitio de Git Hub.
2. Se modifica el archivo **MyController.java** para crear un servicio que calcule el factorial de un número y regrese el total de dígitos que tuvo ese factorial.
3. Se crea un archivo **Dockerfile** y un archivo **entrypoint.sh**, los cuales se depositan en el directorio **/home/gustavo/curso-dgp/examen2/vertx-sample/target**, y se ponen dentro de ellos los comandos necesarios para crear la imagen del Docker que realiza el cálculo del factorial y sus dígitos.
4. Con el comando **“docker build -t cursomicro05/factorial-vertx2 . ”**  se generó la imagen del docker.
5. Se probó la imagen **cursomicro05/factorial-vertx2** creando con ella un servicio por el puerto 8081 y funcionó correctamente. Comando para generar el contenedor:

**docker run -d -p 8081:8080 cursomicro05/factorial-vertx2.**

1. Se probó el servicio y funcionó correctamente. Los ejemplos de llamadas al servicio son:

<http://192.168.1.76:8081/api/calcula?numero=5>

*Resultado:*

{

"Factorial" : "120",

"Número" : "5",

"Current Node IP" : "172.17.0.2",

"Num. Digitos del factorial" : "3",

"variable\_entorno PBA " : null,

"nombre" : "Jesús"

}

<http://192.168.1.76:8081/api/calcula?numero=20>

*Resultado:*

{

"Factorial" : "2432902008176640000",

"Número" : "20",

"Current Node IP" : "172.17.0.2",

"Num. Digitos del factorial" : "19",

"variable\_entorno PBA " : null,

"nombre" : "Jesús"

}

**SE SUBE AL SITIO DE DOCKERHUB LA IMAGEN DEL DOCKER**

1. Se inició sesión al sitio de Docker Hub con el comando**docker login.**
2. Se subió al sitio de Docker Hub la imagen **cursomicro05/factorial-vertx2** haciendo uso del comando **docker push cursomicro05/factorial-vertx.**

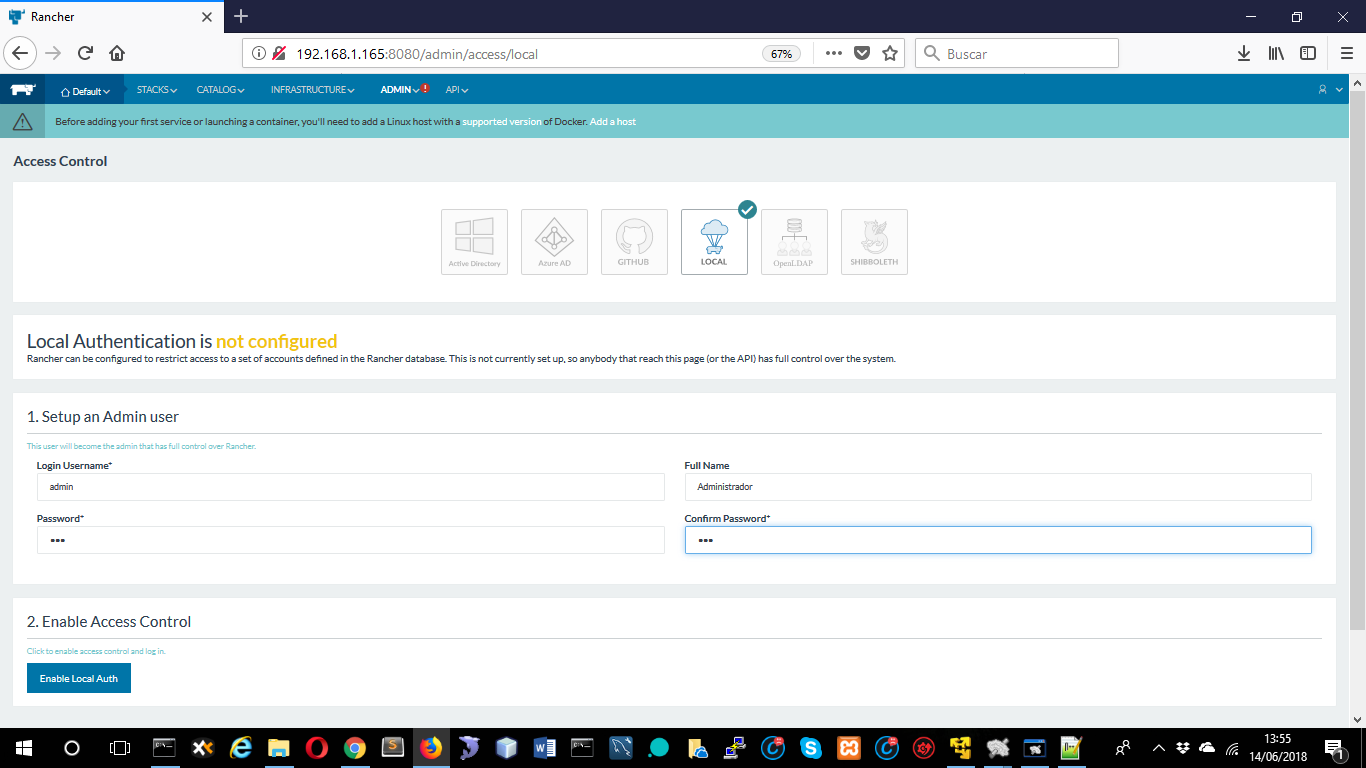
**PUBLICACIÓN DE UN MICROSERVICIO EN RANCHER**

1. Comando para levantar el rancher server:

**sudo docker run -d --restart=unless-stopped -p 8080 rancher/server:stable**

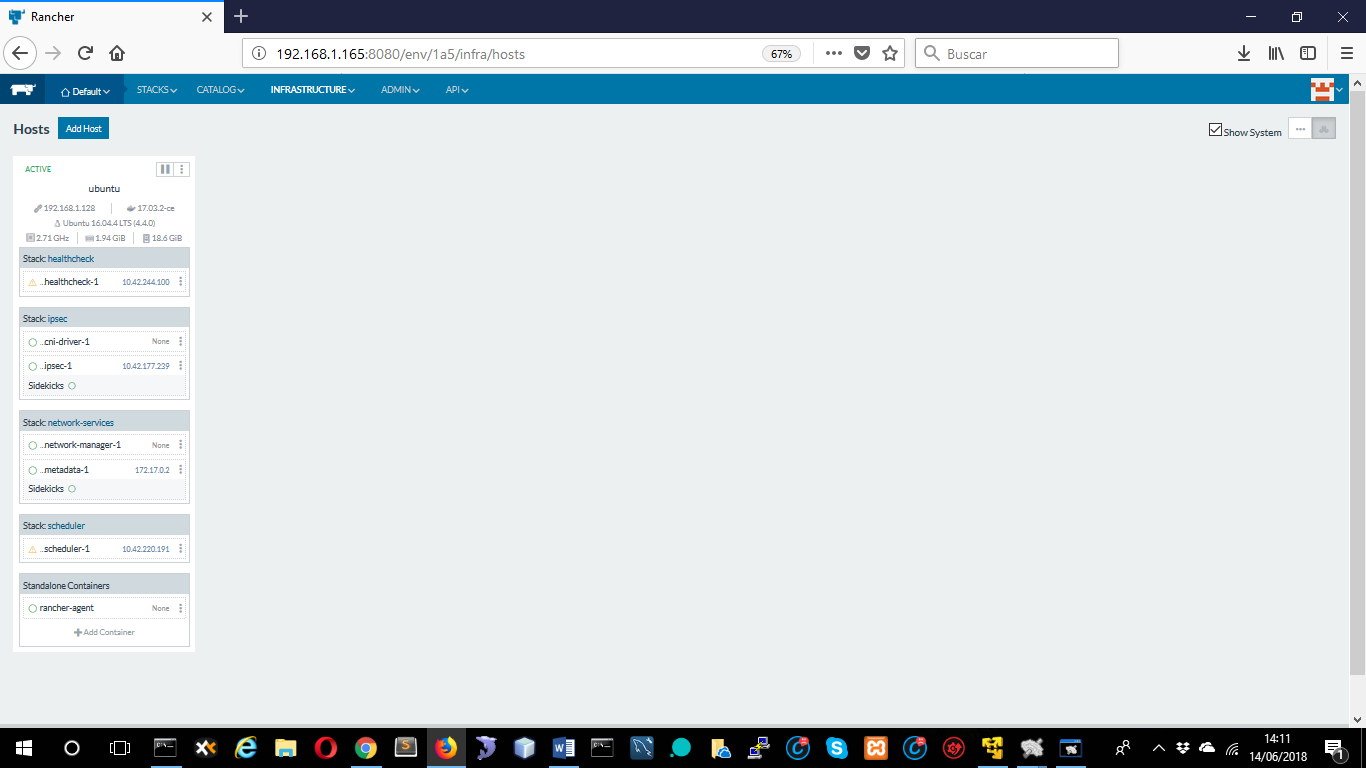
1. Ir a la dirección <http://192.168.1.165:8080> para mostrar en el navegador la ventana-interfaz del rancher server.
2. Una vez dentro de rancher:

**CREAR UNA CUENTA DE USUARIO LOCAL**

1. Entrar a la opción **ADMIN/Access control.**
2. Seleccionar “**LOCAL**”.
3. Capturar en los campos “**Login Username**”: admin, **contraseña**: la\_*contraseña*.
4. Dar clic en “Enable Local Auth”.

**DAR DE ALTA LOS HOSTS O AGENTES**

1. Entrar a la opción **INFRAESTRUCTURE/Hosts**.
2. Dar clic en el botón **Add Host**.
3. Dar clic en el botón **Save**.
4. Copiar el comando para dar de alta los hosts al server, y enviarlo a los responsables de los hosts.
5. Un responsable de host ejecuta el comando y aparece el primer host en la pantalla.



**ALTA DE UN SERVICIO**

1. Entrar a la opción **Default/Default.**
2. Clic en el botón **Define a Service.**
3. Capturar los datos en los campos

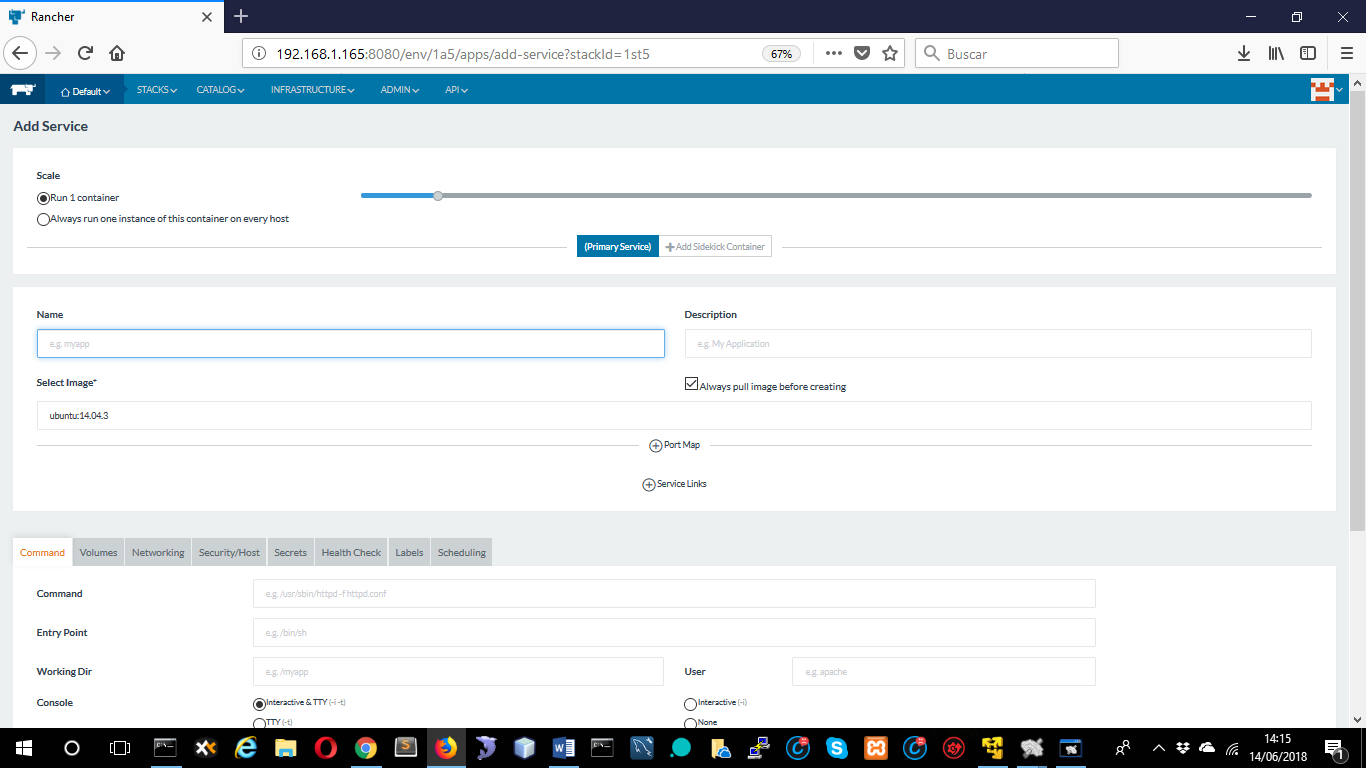
**Name:** Aquí se pone el nombre que queremos que tenga el servicio.

**Description:** Aquí se pone alguna descripción del servicio.

**Select Image:** Aquí se pone el nombre de la imagen de docker que contiene el

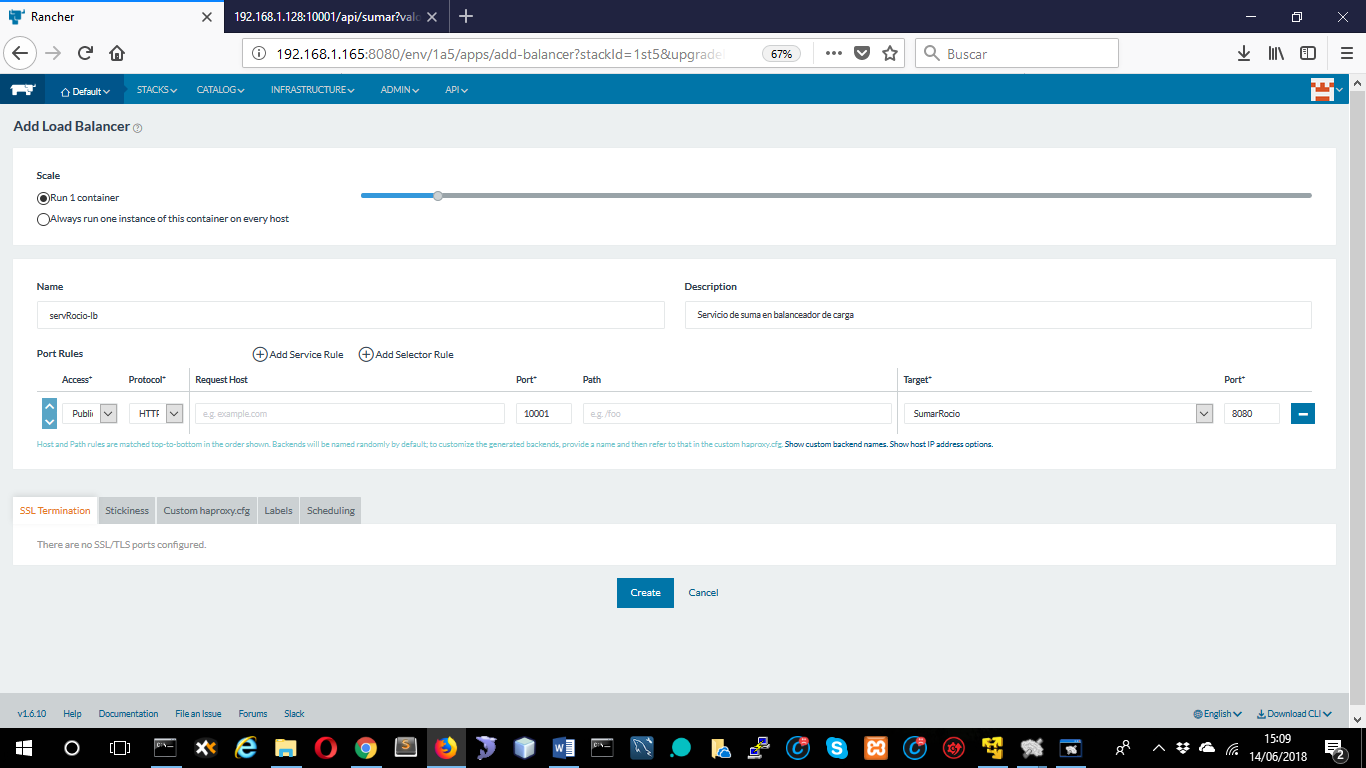
servicio que queremos publicar.

1. En pestaña **Command** , Seleccionar **None.**
2. En la pestaña **Security/Host** , en **Log Driver** seleccionar **json-file.**
3. Clic en el botón **Create.** // Para crear el servicio.

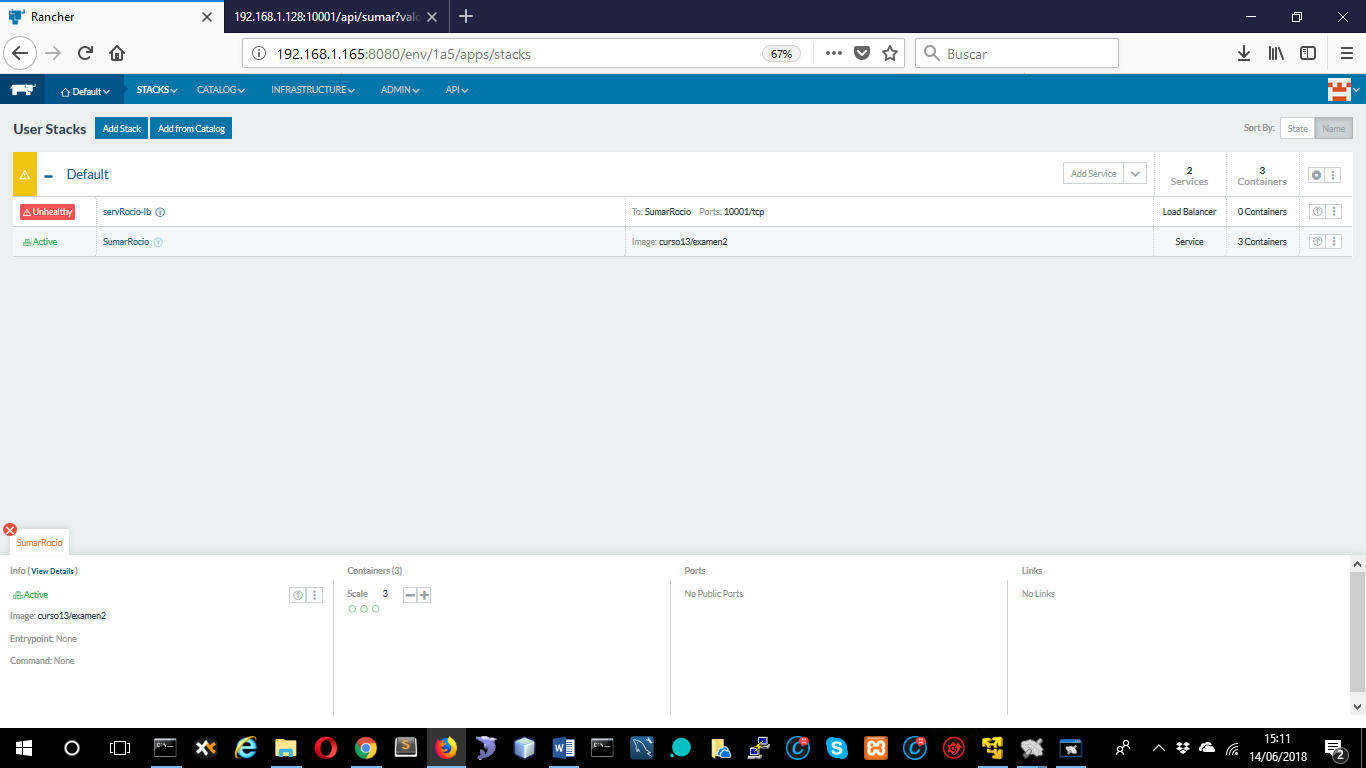


**CREAR UN BALANCEADOR DE CARGA**

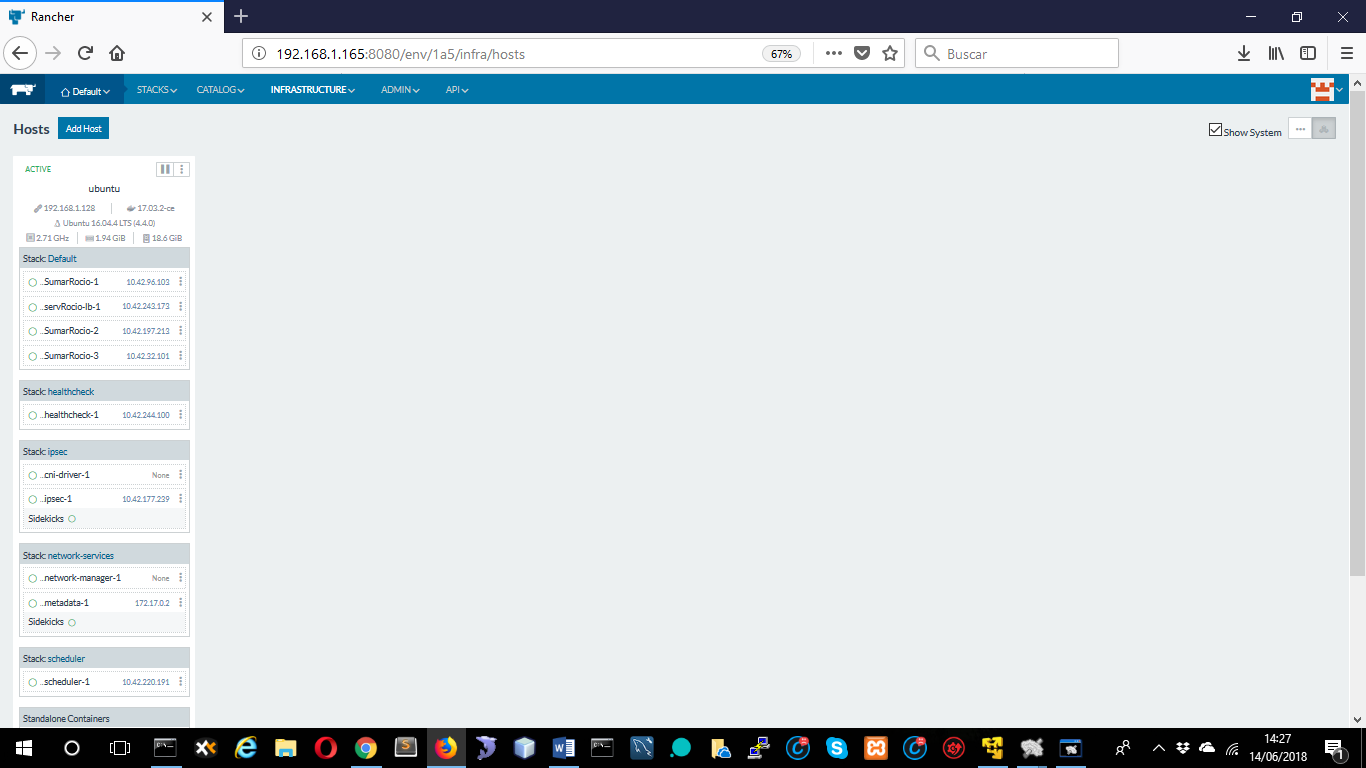
1. Ir a la opción **Default/Default.**
2. Seleccionar **Add Load Balancer.**
3. Capturar **Name:** *balanRocio***, Description:** *Balanceador para el servicio de Rocio***, Puerto-externo:** 10001**, Target:** SumarRocio **y Puerto-interno:** 8080
4. Clic en botón **Crear**.



**DAR DE ALTA 3 INSTANCIAS DEL SERVICIO**

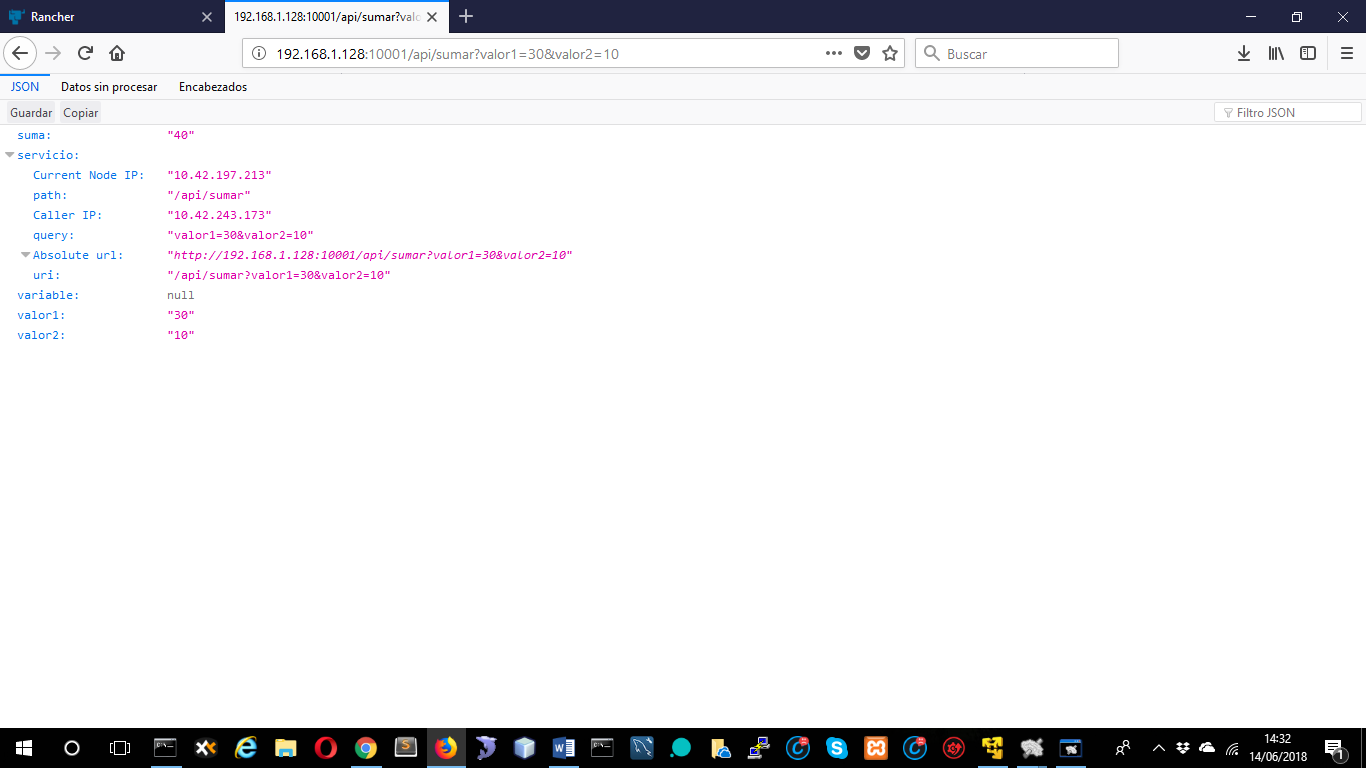
1. Clic en la letra “ i” del servicio
2. En Conteiners llevar a 3

**SE VISUALIZA EL BALANCEADOR Y LAS 3 INSTANCIAS DEL SERVICIO EN EL HOST**



**PROBAR LOS SERVICIOS CON EL BALANCEADOR**

1. Abrir un browser e ingresar a la dirección-puerto del balanceador de carga, indicando el servicio a ejecutar.

<http://192.168.1.128:10001/api/sumar?valor=20&valor2=10>

1. Al ejecutar repetidamente la url anterior se observó que en efecto en el dato **Current Node IP** la ip cambiaba repetidamente.