# TUTORIAL DE MULTIPASS EN UBUNTU

Multipass es una herramienta poderosa y flexible que puede usarse para muchos propósitos. En su forma más simple, se puede usar para crear y destruir rápidamente máquinas virtuales (instancias) de Ubuntu en cualquier máquina host. Utilizado en mayor medida, Multipass es una mininube local en su computadora portátil, lo que permite la prueba y el desarrollo de aplicaciones en la nube basadas en contenedores o de instancias múltiples.

Este tutorial le permitirá comprender cómo funciona Multipass y las habilidades que necesita para usar sus funciones principales.

Contenido:

Instalar Multipass

Crear y usar una instancia básica

Crear una instancia personalizada

Administrar instancias

Ponga sus instancias en uso

Ejecute un servidor web simple

Lanzar desde un Blueprint para ejecutar contenedores Docker

Próximos pasos

## Instalar Multipass

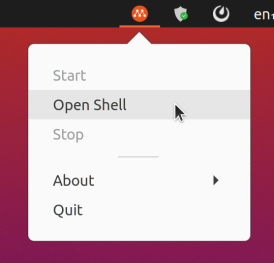
Multipass está disponible para Linux, macOs o Windows. Para instalarlo en el sistema operativo de su elección, siga las instrucciones que se proporcionan aquí. Nota: este tutorial demuestra el uso en Linux, específicamente en Ubuntu, pero la experiencia en cualquier sistema operativo debería ser similar.

## Crear y usar una instancia básica

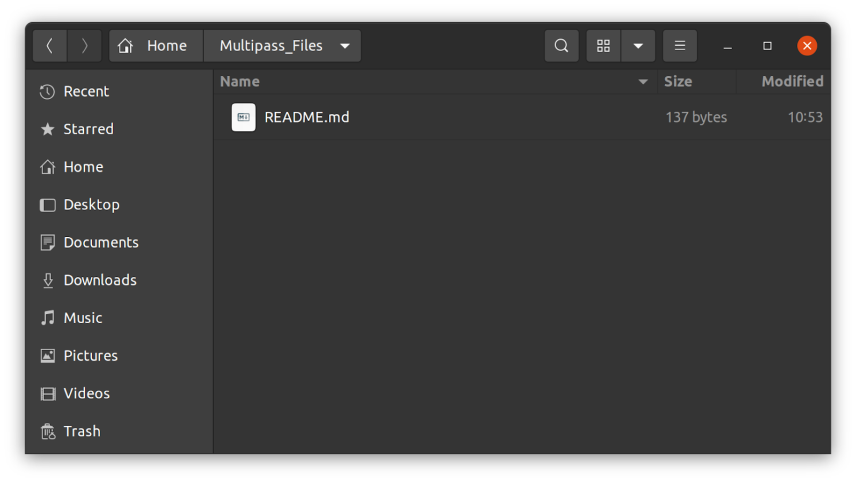
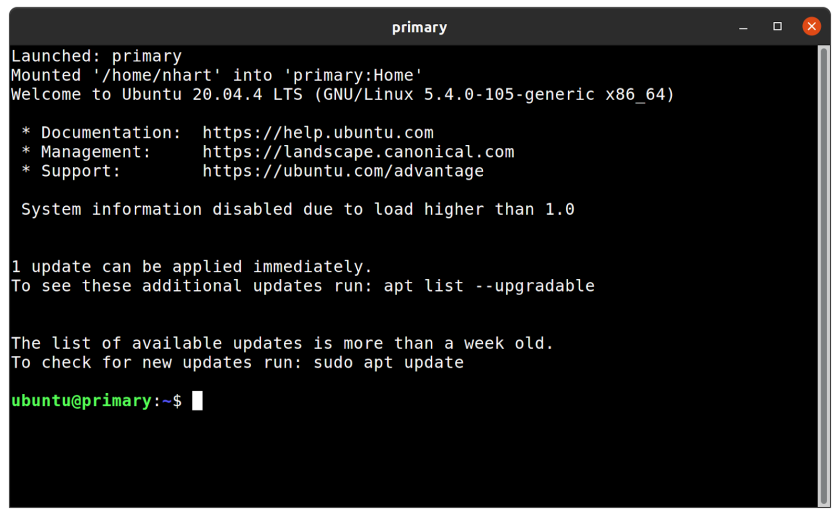
Desde el lanzador de aplicaciones, comencemos Multipass. En Ubuntu, presione la tecla súper y escriba Multipass, o busque Multipass en el panel Aplicaciones en la parte inferior izquierda del escritorio.

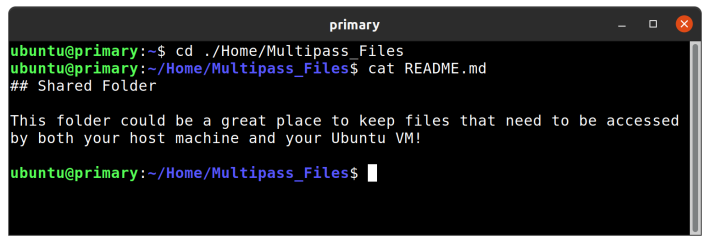
Una vez que hayamos lanzado la aplicación, deberíamos ver el icono de la bandeja Multipass en la sección superior derecha de la pantalla:

Hagamos clic en el icono, luego en "Open Shell".https://lh5.googleusercontent.com/oUYTLsURnmTkQOdA8XK7j7vQsPTnrQ0FgywygLZodhE7ZObA8LIsXJDsZImx1SRTH-0XTgskjzfRoU3-359jYrXGyi-D5xUy5ZQI6ibs9L88bOJGs5DXljYVJUGdEPk2avLmdaGGyZhXYYOqww



Hacer clic en este botón ha ce muchas cosas en segundo plano: crea una nueva máquina virtual (instance), denominada principal (primary) , con 1 GB de RAM, 5 GB de disco y 1 CPU; instala la versión más reciente de Ubuntu LTS en esa instancia; monta nuestro directorio $HOME en la instancia; y abre un shell a la instancia, anunciado por el símbolo del sistema ubuntu@primary. Puede ver elementos de esto en la captura de pantalla a continuación.

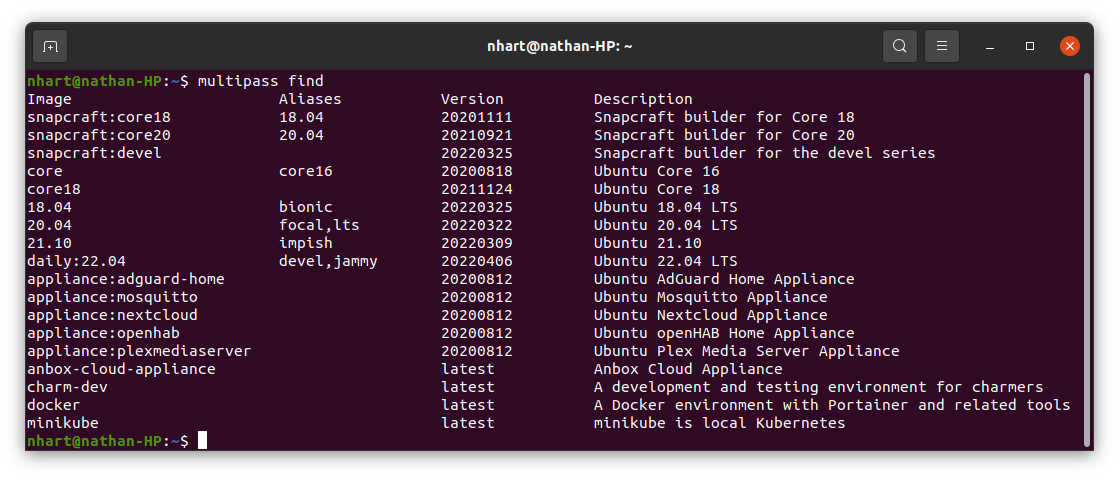
¡Vamos a probarlo! Como acabamos de aprender, el paso anterior montó automáticamente nuestro directorio $HOME en la instancia. Usemos esto para compartir datos con nuestra instancia. Más concretamente, creemos una nueva carpeta en nuestro directorio $HOME llamada Multipass\_Files:

Como puede ver, he agregado un archivo Léame en esta carpeta compartida. Busquemos la carpeta y leamos el archivo de nuestra nueva instancia:

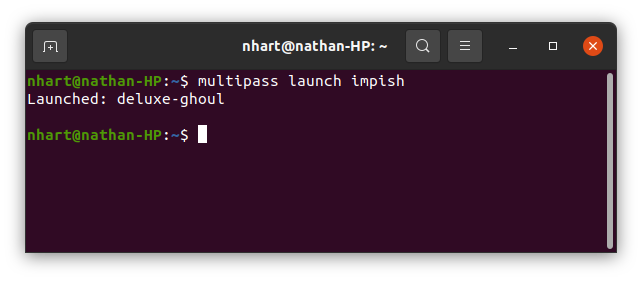
Ejercicio 1:

Cuando hizo clic en Open Shell hace un momento, lo que sucedió en segundo plano fue el equivalente a los comandos CLI multipass launch: nombre principal seguido de multipass shell. Abra una terminal y pruebe el shell multipass (si no siguió los pasos anteriores, primero deberá ejecutar el comando de inicio).

## Crear una instancia personalizada

Multipass tiene una excelente función para ayudarnos a comenzar a crear instancias personalizadas. Abramos una terminal y ejecutemos el comando multipass find. Esto nos muestra una lista de todas las imágenes que podemos lanzar a través de Multipass actualmente.

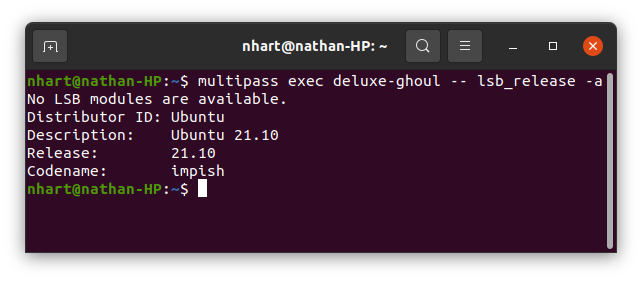
Iniciemos una instancia que ejecute Ubuntu 21.10 ("Impish Indri") escribiendo el comando multipass launch impish

Ahora tenemos una instancia en ejecución que ha sido nombrada aleatoriamente por Multipass, en mi caso se llama deluxe-ghoul.

Podemos verificar información básica sobre nuestra nueva instancia ejecutando lo siguiente:

multipass exec deluxe-ghoul -- lsb\_release -a

Esto le dice a multipass que ejecute el comando lsb\_release -a en la instancia "deluxe-ghoul".

Quizás después de usar esta instancia por un tiempo, decidamos que lo que realmente necesitamos es la última versión LTS de Ubuntu, con un nombre más informativo y un poco más de memoria y disco. Podemos eliminar la instancia deluxe-ghoul ejecutando:

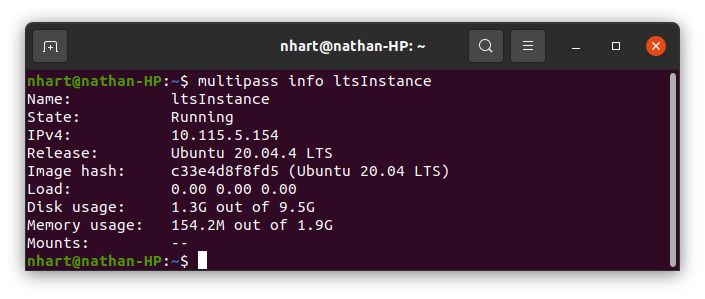
multipass delete deluxe-ghoul

Ahora lancemos el tipo de instancia que estamos buscando ejecutando esto:

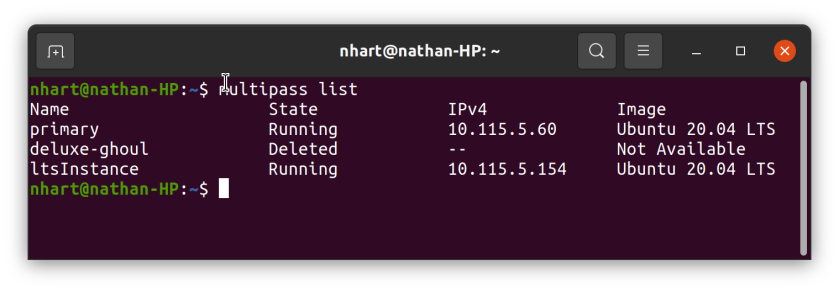
multipass launch lts --name ltsInstance --mem 2G --disk 10G --cpus 2

## Administrar instancias

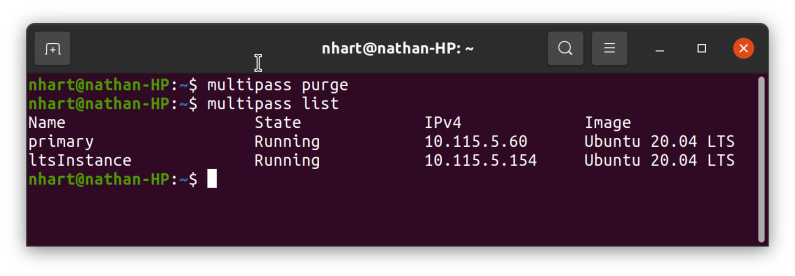
Ahora confirmemos que esta nueva instancia tiene las especificaciones que estamos buscando ejecutando la información de varias pasadas ltsInstance



Hemos creado y eliminado bastantes instancias ahora. Ejecutemos multipass list para ver qué instancias tenemos actualmente.



Tenemos dos instancias actualmente en ejecución, la instancia principal y nuestra máquina LTS con especificaciones personalizadas. Nuestra instancia deluxe-ghoul todavía está en la lista, pero su estado es "Eliminado". Podemos recuperar esta instancia ejecutando multipass recovery deluxe-ghoul, pero por ahora eliminemos la instancia de forma permanente ejecutando multipass purge. Ejecutar la multipass list nuevamente confirma que la instancia ahora se eliminó de forma permanente:



Ahora hemos visto algunas formas de crear, personalizar y eliminar una instancia. ¡Ahora pongamos esas instancias a trabajar!

## Ponga sus instancias en uso

### Ejecute un servidor web simple

Volvamos a esa instancia LTS personalizada que creamos. Tome nota de su dirección IP revelada por la lista de múltiples pasos en el paso anterior, luego ejecute el shell de múltiples pasos ltsInstance para abrir un shell en la instancia.

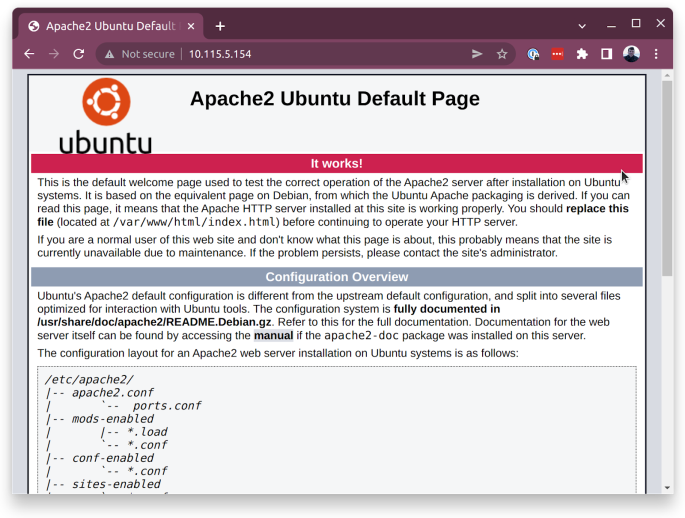
Desde el shell, ahora podemos ejecutar:

sudo apt **update**

sudo apt **install** apache2

Ahora, abramos un navegador y escribamos la dirección IP de la instancia en la barra de direcciones. Ahora deberíamos ver la página de inicio predeterminada de Apache.

¡Así de simple, tenemos un servidor web ejecutándose en una instancia de Multipass!



Podemos usar este servidor web localmente para cualquier tipo de desarrollo local o prueba que deseemos. Sin embargo, si queremos acceder a este servidor web desde Internet (por ejemplo, desde una computadora diferente), necesitamos una instancia que esté expuesta a la red externa.

## Lanzar desde un Blueprint para ejecutar contenedores Docker

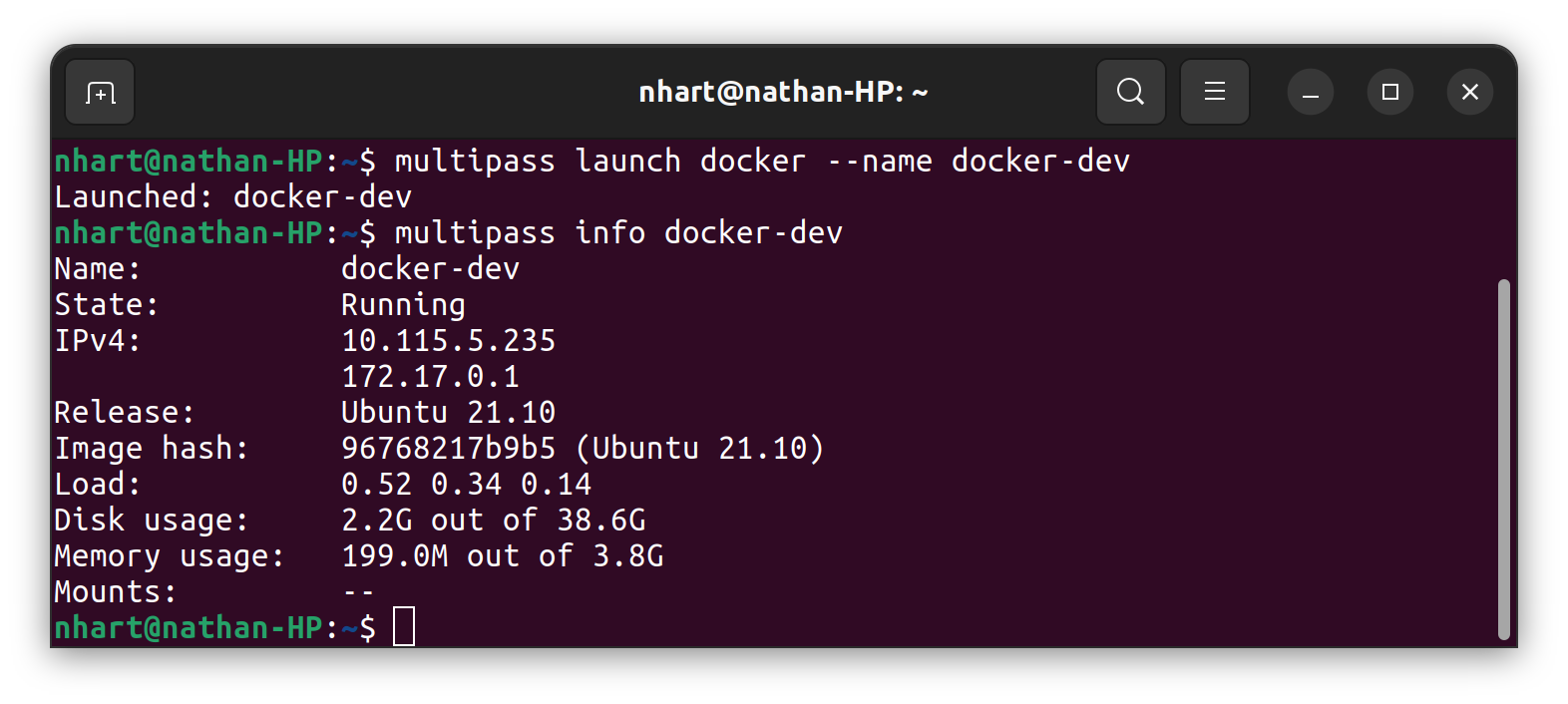
Algunos entornos requieren mucha configuración y preparación. Los blueprints multipass son instancias con un alto nivel de personalización. Docker Blueprint, por ejemplo, es un entorno Docker preconfigurado con un contenedor Portainer que ya se está ejecutando. Podemos lanzar una instancia usando Docker Blueprint ejecutando

multipass launch docker --name docker-dev

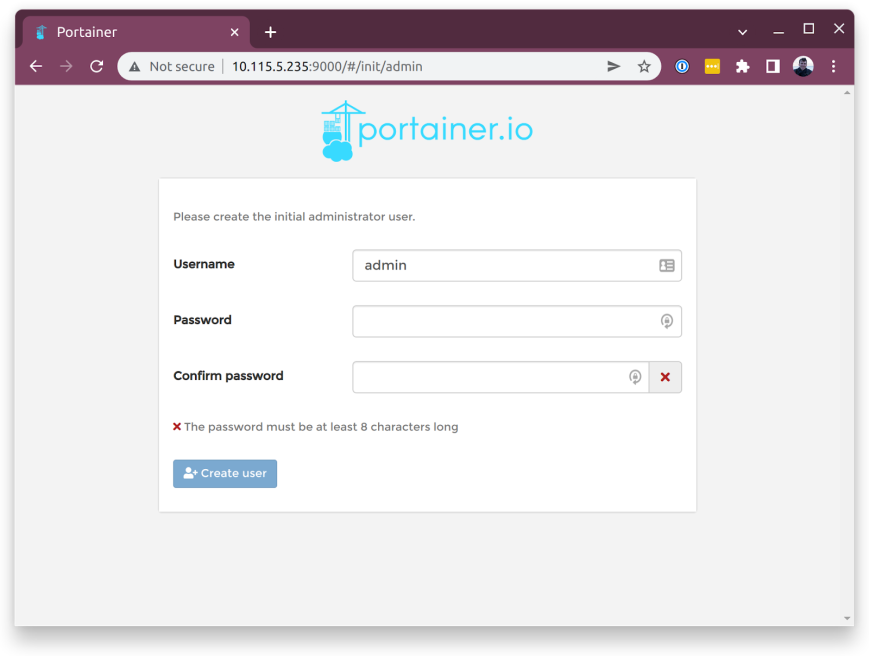
Una vez que haya terminado, ejecutemos

multipass info docker-dev

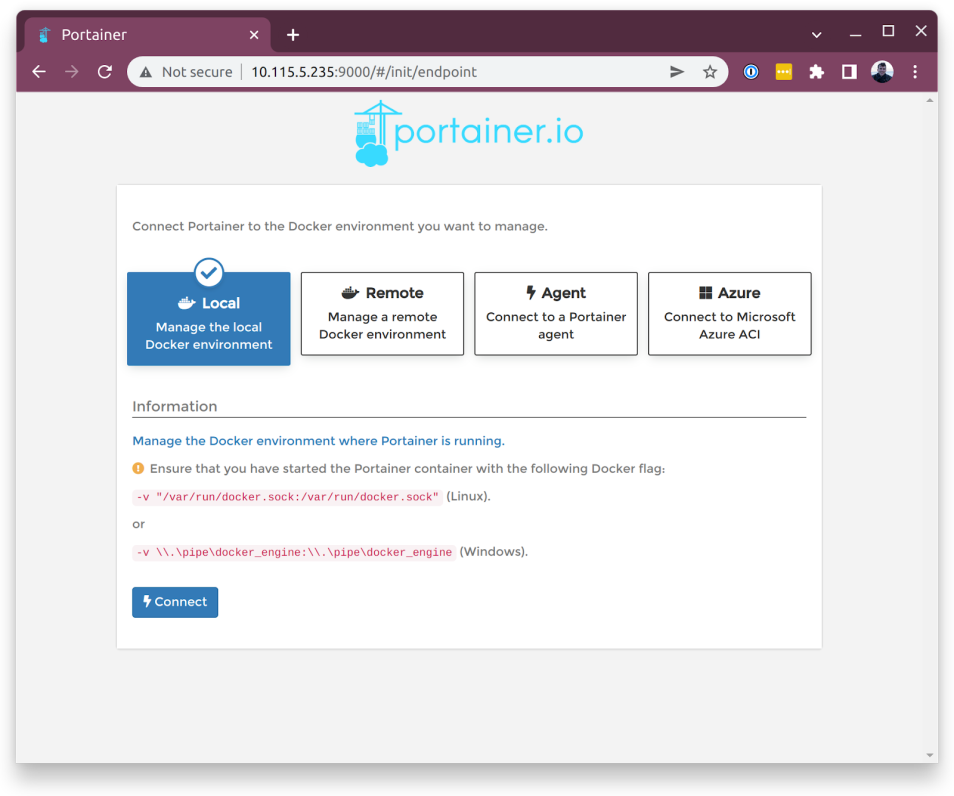
para anotar la IP de la nueva instancia.

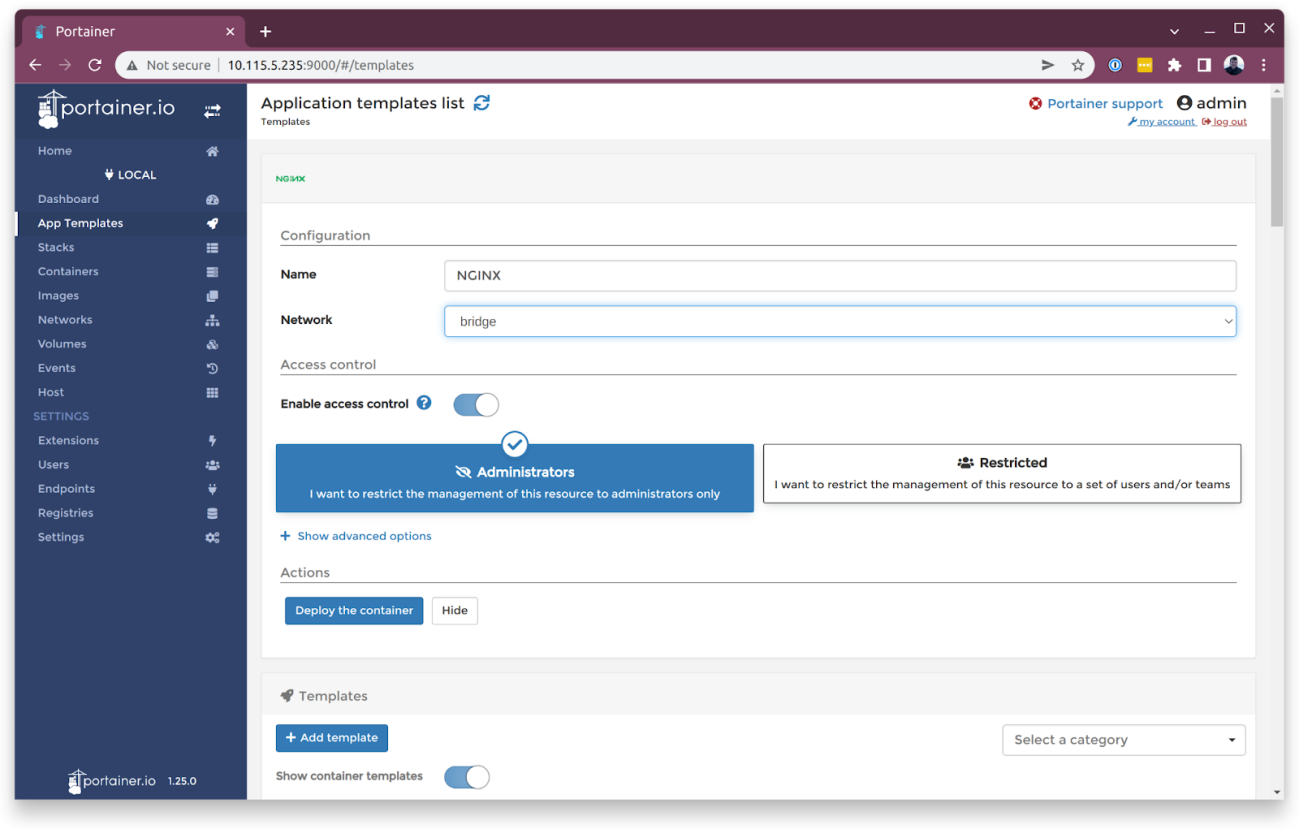


Tomemos la dirección IP que comienza con "10" y la peguemos en nuestro navegador, luego agreguemos dos puntos y el puerto predeterminado del portainer, 9000, así: 10.115.5.235:9000. Esto nos llevará a la página de inicio de sesión de Portainer, donde podemos establecer un nombre de usuario y una contraseña.



A partir de ahí, seleccionemos un entorno docker local.

Desde allí, podemos hacer clic en el botón al final de la ventana acoplable "local" recién creado, navegar a la página de plantillas de la aplicación y seleccionar NGINX.

Desde el panel de control de Portainer, podemos ver los puertos que nginx tiene disponibles. ¡Podemos navegar a la dirección IP de nuestra instancia seguida de ese número de puerto para ver que efectivamente tenemos nginx ejecutándose en un contenedor docker dentro de Multipass!

