

Introducción a Docker

UD 07. Caso práctico 05 (Extra) - IsardVDI, gestor de virtualización



Autor: Sergi García Barea

Actualizado Febrero 2025

Licencia





Reconocimiento – NoComercial - CompartirIgual (BY-NC-SA): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

 **Importante**

 **Atención**

 **Interesante**

1. Introducción	2
2. Instalando “IsardVDI”	3
3. Puesta en marcha de “IsardVDI”	3
4. Bibliografía	4

UD07. CASO PRÁCTICO 05 (EXTRA)

1. INTRODUCCIÓN

En este caso práctico vamos a poner en marcha la herramienta “**IsardVDI**”. Aunque esta herramienta no es específica de Docker, la especificamos como caso práctico, ya que además de su interés, esta se levanta con contenedores Docker.

IsardVDI es un gestor de cualquier tipo de virtualización soportada por “**KVM**” <https://www.linux-kvm.org> (https://es.wikipedia.org/wiki/Kernel-based_Virtual_Machine).

“**IsardVDI**” está desarrollado por gente de Barcelona. La web del oficial del proyecto es <https://www.isardvdi.com/>.

2. BUINSTALANDO “ISARDVDI”

Para instalar seguiremos los pasos de <https://isard.gitlab.io/isardvdi-docs/install/#quick-start>

En primer lugar, clonaremos el repositorio de IsarVDI con.

```
git clone https://gitlab.com/isard/isardvdi
```

Tras ello accedemos al directorio recién creado “isardvdi”

```
cd isardvdi
```

Y una vez allí copiamos la configuración de ejemplo para usarla como configuración de nuestro despliegue y ejecutamos “build.sh”.

```
cp isardvdi.cfg.example isardvdi.cfg
./build.sh
```

Tras ello, descargamos las imágenes asociadas.

```
docker compose pull
```

Y finalmente ponemos en marcha el sistema. Importante: aunque el comando haya finalizado, la **primera vez habrá que esperar dos minutos aproximadamente para que se pueble la base de datos.**

```
docker compose up -d
```

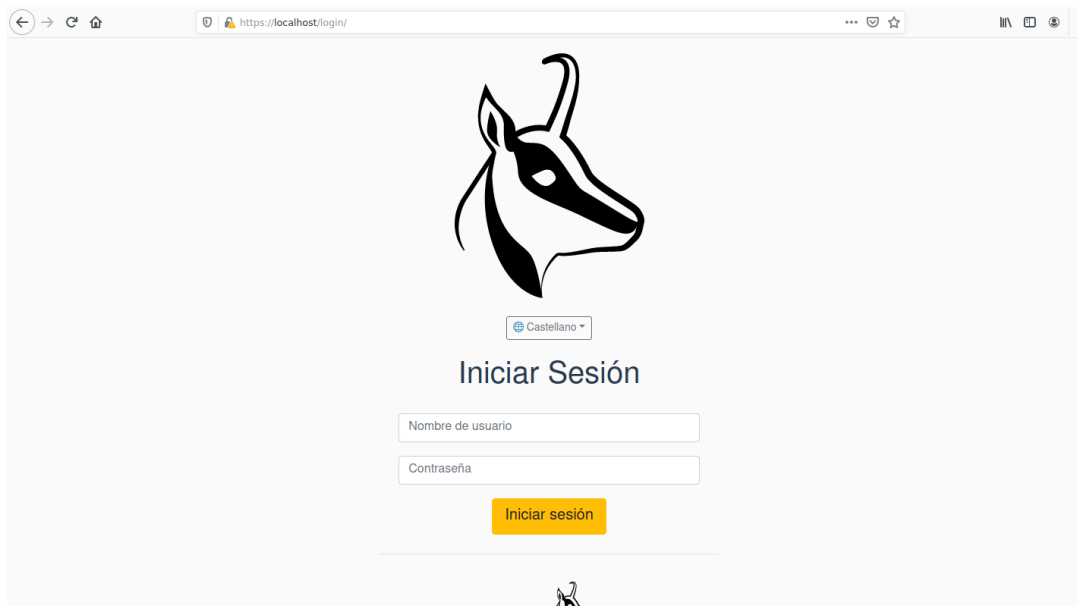
Al ponerlo en marcha, la persistencia del sistema se realizará con los directorios del anfitrión “/opt/isard” y “/opt/isard-local”. Tras ponerlo en marcha observaremos algo similar a:

```
Creating network "isard-network" with the default driver
Creating network "isard-vpn" with the default driver
Creating isard-redis    ... done
Creating isard-vpn      ... done
Creating isard-static   ... done
Creating isard-backend  ... done
Creating isard-db       ... done
Creating isard-hypervisor ... done
Creating isard-api      ... done
Creating isard-webapp   ... done
Creating isard-portal   ... done
Creating isard-squid    ... done
Creating isard-grafana  ... done
Creating isard-websockify ... done
Creating isard-engine   ... done
sergi@ubuntu:~$
```

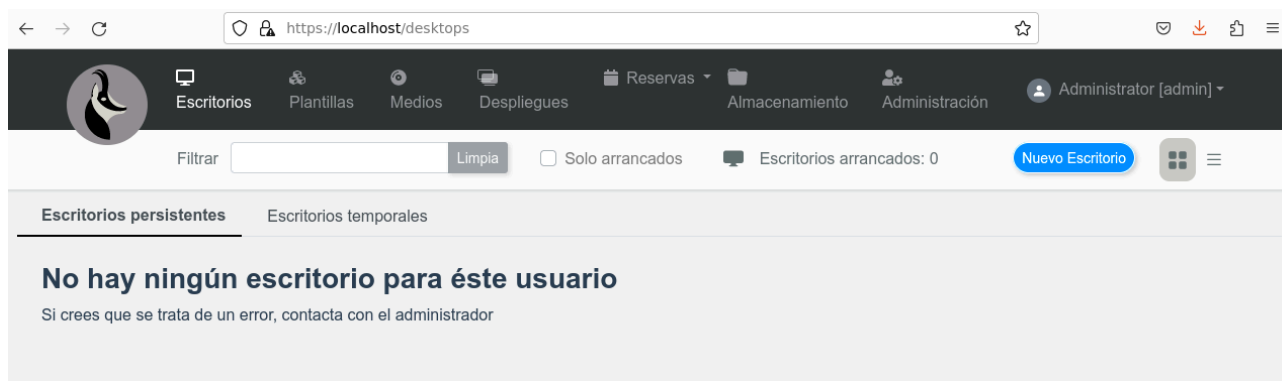
En el caso de que queramos desinstalar la aplicación o que hubiera algún problema y queramos reinstalar todo, además de eliminar los contenedores usando “**Docker Compose**”, deberemos borrar los directorios “/opt/isard” y “/opt/isard-local”.

3. PUESTA EN MARCHA DE “ISARDVDI”

Una vez pasados los dos minutos de espera, podremos acceder a la interfaz mediante la dirección <https://localhost>. El usuario por defecto es “**admin**” y la contraseña “**IsardVDI**”.



Si todo ha ido bien, veremos una ventana como la siguiente:



Una vez accedido, si quieres comprobar las bondades de IsardVDI, te recomendamos sigas su guía de primeros pasos <https://isard.gitlab.io/isardvdi-docs/install/first-steps/> donde te indicará como puedes bajarte escritorios, usarlos, etc. (Nota: para visualizar los escritorios, deberás permitir ventanas emergentes.)

4. BIBLIOGRAFÍA

- [1] Docker Docs <https://docs.docker.com/>
- [2] IsardVDI <https://www.isardvdi.com/> y <https://isard.gitlab.io/isardvdi-docs/>