

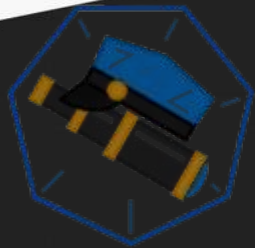
# Podman, Buildah & Skopeo



podman



buildah

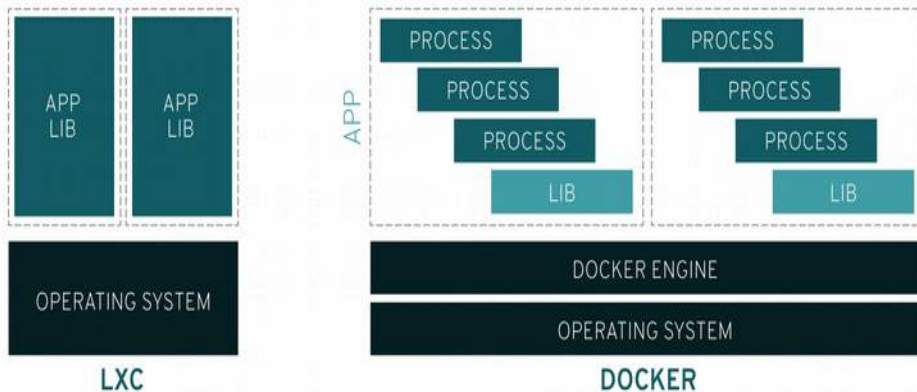


skopeo

# Docker

Docker usa el kernel de Linux y las funciones de este, como Cgroups y namespaces, para agregar los procesos, de modo que puedan ejecutarse de manera independiente similar a una máquina virtual

Traditional Linux containers vs. Docker



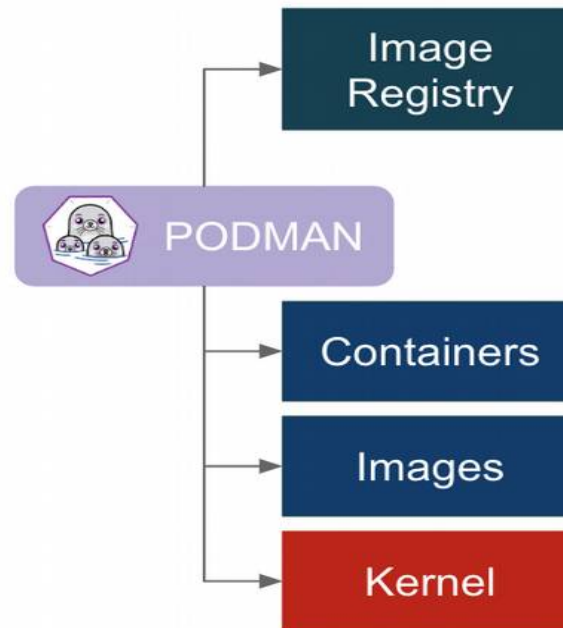
Despliegue e integración continua

- Despliegue rápido
- Control de versiones
- Eficiencia

# ¿Qué es Podman y Buildah?

Podman es un motor de contenedores desarrollado por RedHat similar a Docker y Buildah es el encargado de construir imágenes como Docker.

- Daemon-less
- Root-less
- Fácil de usar



# ¿Qué es Skopeo?

Skopeo es una herramienta diseñada para gestionar las imágenes en diferentes registros y de forma local

- Copiar imágenes entre registros
- Inspeccionar imágenes en registros remoto
  - Firmar y verificar firmas
- Borrar imágenes de registros remotos

# Diferencias entre Podman y Builda con Docker

En mi experiencia hay varias diferencias entre ambas tecnologías.

- Podman levanta menos servicios por contenedor
- Buildah no tiene cache de creación de imágenes
- Se pueden hacer inspect de imagenes en registros remotos
  - SELinux y los volumenes persistentes
  - Carece de DNS entre contenedores

# Como instalar

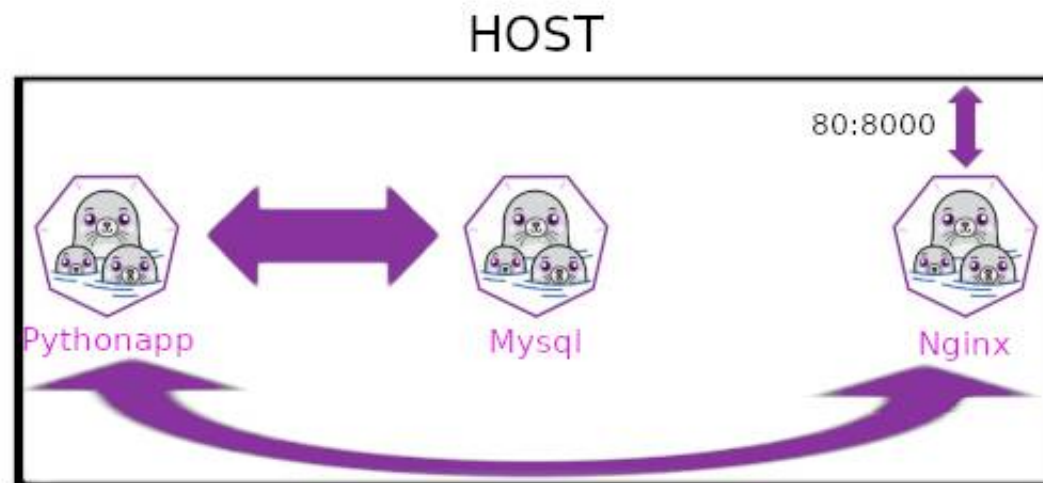
Tanto Podman como Buildah como Skopeo vienen instalados en la nueva versión CentOS RHEL 8 pero si queremos instalarlo en otra distribución como debian por ejemplo.

```
echo 'deb https://download.opensuse.org/repositories/devel:/kubic:/libcontainers:/stable/Debian_10/ /' >
/etc/apt/sources.list.d/devel:kubic:libcontainers:stable.list
curl -L https://download.opensuse.org/repositories/devel:/kubic:/libcontainers:/stable/Debian_10/Release.key |
sudo apt-key add -
sudo apt-get update -qq
sudo apt-get -qq -y install podman buildah skopeo
```

# ¿Qué vamos a hacer?

El escenario que vamos a probar a continuación son 3 contenedores.

Uno contendrá una aplicación Python y la servirá, otro contendrá la base de datos y el tercero hará de proxy inverso



Comencemos...