

Configuración de Docker para Ubuntu

Este README explica cómo configurar y ejecutar un contenedor de Ubuntu utilizando Docker. El contenedor establecerá una contraseña para el usuario root y proporcionará utilidades básicas de Linux.

Archivos

1. `.env`

Este archivo contiene las variables de entorno que se usarán en la configuración del contenedor. Específicamente, define la contraseña `ROOT_PASSWORD` para el usuario root en el contenedor.

```
ROOT_PASSWORD=abcd1234
```

2. `Dockerfile`

El Dockerfile especifica la imagen base (Ubuntu) e instala algunos paquetes básicos (curl, wget, vim, net-tools, etc.). También incluye el script `entrypoint.sh` que establece la contraseña de root usando el valor definido en el archivo `.env`.

```
FROM ubuntu:latest

# Instalar paquetes básicos
RUN apt-get update && apt-get install -y \
    sudo \
    curl \
    wget \
    vim \
    net-tools \
    iputils-ping

# Establecer variables de entorno
ENV LANG C.UTF-8
ENV LC_ALL C.UTF-8

# Establecer el directorio de trabajo
WORKDIR /root

# Copiar el script de entrada y hacerlo ejecutable
COPY entrypoint.sh /entrypoint.sh
RUN chmod +x /entrypoint.sh

# Ejecutar el script de entrada
ENTRYPOINT ["/entrypoint.sh"]

# Comando predeterminado
CMD ["/bin/bash"]
```

3. entrypoint.sh

Este script se ejecuta cuando el contenedor inicia. Establece la contraseña de root usando la variable de entorno `ROOT_PASSWORD`.

```
#!/bin/bash

# Verificar si ROOT_PASSWORD está establecido
if [ -z "$ROOT_PASSWORD" ]; entonces
    echo "Error: ROOT_PASSWORD no está establecido!"
    exit 1
fi

# Establecer la contraseña de root
echo "root:$ROOT_PASSWORD" | chpasswd

# Continuar con el comando original
exec "$@"
```

4. docker-compose.yml

Este archivo define cómo construir y ejecutar el contenedor utilizando Docker Compose. Carga las variables de entorno del archivo `.env`, construye el contenedor desde el Dockerfile y expone el puerto `8080`.

```
version: '3.8'
services:
  linux:
    container_name: ubuntu-linux
    build:
      context: .
      dockerfile: Dockerfile
    restart: always
    ports:
      - "8080:80"
    networks:
      - network_local_server
    volumes:
      - linux_data:/var/lib/data
      - "/home/ubuntu:/root"
    environment:
      - ROOT_PASSWORD=abcd1234
    healthcheck:
      test: ["CMD-SHELL", "ping -c 1 google.com || exit 1"]
      interval: 30s
      timeout: 10s
      retries: 5
    labels:
      - com.corhuila.group=linux-environment
```

```
volumes:
  linux_data:
    driver: local

networks:
  network_local_server:
    external: true
```

Cómo Usar

Paso 1: Construir y Ejecutar el Contenedor

Asegúrate de tener Docker y Docker Compose instalados en tu máquina. Luego, ejecuta el siguiente comando para construir e iniciar el contenedor:

```
docker-compose up --build
```

Paso 2: Acceder al Contenedor

Para acceder al contenedor con la contraseña de root definida, ejecuta el siguiente comando:

```
docker run -it --env ROOT_PASSWORD=abcd1234 ubuntu-linux:latest /bin/bash
```

Esto te dará acceso a la shell bash del contenedor de Ubuntu en ejecución.

Paso 3: Probar la Contraseña de Root

Puedes verificar si la contraseña de root ha sido establecida intentando cambiar al usuario root dentro del contenedor:

```
su root
```

Cuando se te solicite la contraseña, ingresa la que se especifica en el archivo `.env` (por ejemplo, `abcd1234`).

Solución de Problemas

- Asegúrate de que el archivo `.env` esté correctamente formateado y sea accesible para Docker Compose.
- Asegúrate de que Docker esté instalado y en funcionamiento en tu máquina.
- Si el contenedor se reinicia continuamente, revisa los registros con `docker-compose logs` para ver si hay errores.

Resolución del problema: "REMOTE HOST IDENTIFICATION HAS CHANGED"

Si recibes el siguiente mensaje al intentar conectarte vía SSH:

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!    @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE QUE ALGUIEN ESTÉ HACIENDO ALGO MALICIOSO!
...
Host key verification failed.
```

Este error ocurre porque la clave del host ha cambiado o no coincide con la registrada en [known_hosts](#). Sigue estos pasos para resolverlo:

Pasos para resolver el problema:

1. Ejecuta el siguiente comando para eliminar la clave ofensiva:

```
ssh-keygen -R [localhost]:2222
```

2. Intenta reconectarte con el siguiente comando:

```
ssh root@localhost -p 2222
```

Advertencia sobre el uso de Docker

Si utilizas el siguiente comando Docker:

```
docker run -it --env ROOT_PASSWORD=abcd1234 ubuntu-linux:latest /bin/bash
```

Los cambios que realices dentro del contenedor **no se guardarán** cuando lo detengas. Esta es una desventaja importante si necesitas persistir las modificaciones.

Configuración de redes en Dockerfile

La configuración de red debe incluir el siguiente bloque para definir el driver y el nombre de la red:

```
networks:
  network_local_server:
    driver: bridge
    name: network_local_server
```

