Sistemas Informáticos

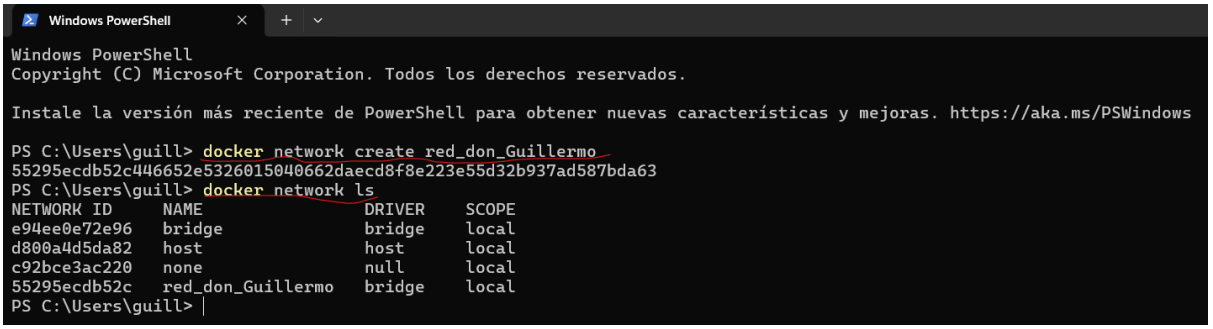
Desarrollo de Aplicaciones Web
Desarrollo de Aplicaciones Multiplataforma
CURSO 24-25

Bloque 4
Práctica 4: Gestión de Procesos y Servicios en Linux en un Contenedor Ubuntu (parte 2)

Guillermo Domínguez de la Fuente

Ejercicios

- **Creación de una Red Personalizada en Docker**
  - Crear una red personalizada en Docker para conectar los contenedores.
  - Verificar la creación de la red.



- **Creación del Contenedor con Nginx**
  - Crear un contenedor basado en Ubuntu e instala nginx.
    - Consideraciones para la creación del contenedor:
      - Debe ir asociado a la red que acabamos de crear.
      - Debe mapear el puerto 8080 de tu anfitrión.
      - Debes personalizar el hostname.
    - Instala nginx

    - Inicia el servicio nginx
    - Verifica que funciona.

```
root@don_Guillermo:/# service nginx status
 * nginx is not running
root@don_Guillermo:/# service nginx start
 * Starting nginx nginx
root@don_Guillermo:/# service nginx status
 * nginx is running
root@don_Guillermo:/#
```

- **Creación del Contenedor con MySQL**

docker run -it --name mysql-container-practica5 --hostname don_Guillermo_practica5-mysql --network red_don_Guillermo -e MYSQL_ROOT_PASSWORD=02041993 ubuntu:latest bash

```
PS C:\Users\guill> docker run -it --name mysql-container-practica5 --hostname don_Guillermo_practica5-mysql --network red_don_Guillermo -e MYSQL_ROOT_PASSWO
RD=02041993 ubuntu:latest bash
root@don_Guillermo_practica5-mysql:/#
```

  - Configurar MySQL para permitir conexiones remotas (desde el contenedor de Nginx:

```
root@don_Guillermo_practica5-mysql:/# apt-get update && apt-get install mysql-server -y
```

    - Accede con: mysql -u root -p
    - Crea una base de datos: CREATE DATABASE mi_basedatos;
    - Crea un Usuario y Otorga Privilegios:
      - CREATE USER 'miusuario'@'%' IDENTIFIED BY 'micontraseña';

- GRANT ALL PRIVILEGES ON mi_basedatos.* TO 'miusuario'@'%';

- Aplica los cambios: FLUSH PRIVILEGES;
- Salir de MySQL: EXIT;

```
root@don_Guillermo_practica5-mysql:/# service mysql status
 * MySQL is stopped.
root@don_Guillermo_practica5-mysql:/# service mysql start
 * Starting MySQL database server mysqld
su: warning: cannot change directory to /nonexistent: No such file or directory
                                                                                [ OK ]
root@don_Guillermo_practica5-mysql:/# service mysql status
 * /usr/bin/mysqladmin  Ver 8.0.41-0ubuntu0.24.04.1 for Linux on x86_64 ((Ubuntu))
Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Server version          8.0.41-0ubuntu0.24.04.1
Protocol version        10
Connection              Localhost via UNIX socket
UNIX socket             /var/run/mysqld/mysqld.sock
Uptime:                 25 sec

Threads: 2  Questions: 8  Slow queries: 0  Opens: 119  Flush tables: 3  Open tables: 38  Queries per second avg: 0.320
root@don_Guillermo_practica5-mysql:/# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.41-0ubuntu0.24.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql> CREATE DATABASE MI_BASEDEDATOS;
Query OK, 1 row affected (0.01 sec)
```

```
mysql> CREATE USER 'don_Guillermo'@'%' IDENTIFIED BY '02041993';
Query OK, 0 rows affected (0.05 sec)
```

(IMPORTANTE: Asegurarnos de que nuestro usuario tiene los privilegios necesarios. De no tenerlos, dárselos)

```
mysql> SELECT USER(), CURRENT_USER();
+----------------+----------------+
| USER()         | CURRENT_USER() |
+----------------+----------------+
| root@localhost | root@localhost |
+----------------+----------------+
1 row in set (0.00 sec)

mysql> SHOW GRANTS FOR 'don_Guillermo'@'%';
+-----------------------------------------+
| Grants for don_Guillermo@%              |
+-----------------------------------------+
| GRANT USAGE ON *.* TO `don_Guillermo`@`%` |
+-----------------------------------------+
1 row in set (0.00 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'don_Guillermo'@'%' WITH GRANT OPTION;
Query OK, 0 rows affected (0.04 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> EXIT
Bye
```

- **Conexión entre Contenedores**
  - Verifica la red creada con: docker network inspect my-network

```
PS C:\Users\guill> docker network inspect red_don_Guillermo
[
    {
        "Name": "red_don_Guillermo",
        "Id": "55295ecdb52c446652e5326015040662daecd8f8e223e55d32b937ad587bda63",
        "Created": "2025-03-14T18:30:18.715176082Z",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.18.0.0/16",
                    "Gateway": "172.18.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "Options": {},
        "Labels": {}
    }
]
PS C:\Users\guill>
```

- Desde el contenedor de Nginx, instalar el cliente MySQL.

```
root@don_Guillermo_practica5-mysql:/# apt-get update && apt-get install mysql-server -y
```

- Conectar a la base de datos MySQL desde el contenedor de Nginx.
  - Tendrás que verificar si el contenedor de mysql está configurado correctamente para recibir conexiones de cualquier ip. (netstat -tuln | grep 3306). Tienes que ver algo como esto:
  - ```
    tcp        0      0 0.0.0.0:3306            0.0.0.0:*               LISTEN
    ```

```
root@don_Guillermo_practica5-mysql:/# netstat -tuln | grep 3306
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:33060         0.0.0.0:*               LISTEN
root@don_Guillermo_practica5-mysql:/# nano /etc/mysql/mysql.conf.d/mysqld.cnf
```

IMPORTANTE: Asegurarse de configurar bind-address como 0.0.0.0

```
 GNU nano 7.2                                  /etc/mysql/mysql.conf.d/mysqld.cnf *
#
# The MySQL database server configuration file.
#
# One can use all long options that the program supports.
# Run program with --help to get a list of available options and with
# --print-defaults to see which it would actually understand and use.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html

# Here is entries for some specific programs
# The following values assume you have at least 32M ram

[mysqld]
#
# * Basic Settings
#
user            = mysql
# pid-file      = /var/run/mysqld/mysqld.pid
# socket        = /var/run/mysqld/mysqld.sock
# port          = 3306
# datadir       = /var/lib/mysql


# If MySQL is running as a replication slave, this should be
# changed. Ref https://dev.mysql.com/doc/refman/8.0/en/server-system-variables.html#sysvar_tmpdir
# tmpdir                = /tmp
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address            = 0.0.0.0
mysqlx-bind-address     = 127.0.0.1
#
# * Fine Tuning
#
key_buffer_size         = 16M
# max_allowed_packet    = 64M

^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo         M-A Set Mark     M-] To Bracket
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line    M-E Redo         M-6 Copy         ^Q Where Was
```

```
root@don_Guillermo_practica5-mysql:/# service mysql restart
 * Stopping MySQL database server mysqld
 * Starting MySQL database server mysqld
su: warning: cannot change directory to /nonexistent: No such file or directory

root@don_Guillermo_practica5-mysql:/# service mysql status
 * /usr/bin/mysqladmin  Ver 8.0.41-0ubuntu0.24.04.1 for Linux on x86_64 ((Ubuntu))
Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Server version          8.0.41-0ubuntu0.24.04.1
Protocol version        10
Connection              Localhost via UNIX socket
UNIX socket             /var/run/mysqld/mysqld.sock
Uptime:                 14 sec

Threads: 2  Questions: 8  Slow queries: 0  Opens: 119  Flush tables: 3  Open tables: 38  Queries per second avg: 0.571
root@don_Guillermo_practica5-mysql:/# netstat -tuln | grep 3306
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:33060         0.0.0.0:*               LISTEN
root@don_Guillermo_practica5-mysql:/#
```
```

- Verificar la conexión ejecutando una consulta simple.

```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| MI_BASEDEDATOS     |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.03 sec)
```

- **Gestión de Logs**
  - Verifica los logs de Nginx con: tail -f /var/log/nginx/access.log

```
root@don_Guillermo:/# tail -f /var/log/nginx/access.log
172.18.0.1 - - [14/Mar/2025:18:44:00 +0000] "GET / HTTP/1.1" 200 409 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) C
hrome/134.0.0.0 Safari/537.36"
172.18.0.1 - - [14/Mar/2025:18:44:00 +0000] "GET /favicon.ico HTTP/1.1" 404 196 "http://localhost:8080/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWeb
Kit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36"
exit
```

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.