

# Informe Personal del Proyecto

## Blog en Contenedores LXC

Tobías Paz Posse  
DNI: 43562927

31 de julio de 2025

### 1. Introducción

Como parte de la consigna del Trabajo Práctico Final, se me solicitó montar un blog personal en contenedores LXC dentro de Proxmox. El objetivo era implementar una arquitectura de dos contenedores separados: uno para la parte web y otro para la base de datos. Esto permitía un sistema más organizado, fácil de mantener y optimizado en cuanto a recursos, ya que cada contenedor fue configurado con 128MB de RAM y 1 CPU.

Para la implementación elegí **Debian 12** para ambos contenedores, **Apache2** con **PHP** para el servidor web y **MariaDB** para la base de datos.

### 2. Creación y Configuración de los Contenedores

La creación de los contenedores se realizó a través de la interfaz visual de Proxmox. A cada contenedor se le asignó una IP estática dentro de la red local.

El contenedor web, destinado a alojar el blog, se configuró con las siguientes especificaciones:

Parámetro	Valor
Nombre	43562927A
IP	172.16.90.114
Memoria	128MB
CPU	1 núcleo

El contenedor de base de datos se configuró con:

Parámetro	Valor
Nombre	43562927DB
IP	172.16.90.117
Memoria	128MB
CPU	1 núcleo

### 3. Configuración del Contenedor Web

Una vez creados los contenedores, accedí a la consola del contenedor web a través de la interfaz gráfica de Proxmox para instalar y configurar los servicios.

Primero actualicé el sistema operativo y luego instalé **Apache2** y **PHP** junto con los módulos requeridos para la conexión con **MariaDB**:

```
1 apt update && apt upgrade -y
2 apt install -y apache2 php libapache2-mod-php php-mysql
```

A continuación, habilité los módulos `php8.2` y `rewrite` de Apache para asegurar que el servidor pudiera interpretar los scripts PHP y manejar las URLs adecuadamente:

```
1 a2enmod php8.2
2 a2enmod rewrite
```

Luego, para mantener un directorio web limpio y dedicado, eliminé todo el contenido predefinido:

```
1 rm -rf /var/www/html/*
```

Posteriormente, creé un archivo de configuración de VirtualHost para apuntar el servidor web al directorio `/var/www/html` y lo almacené en la ruta `/etc/apache2/sites-available/blog.conf`. Este paso me permitió deshabilitar la configuración por defecto y habilitar la del blog, reiniciando Apache2 para aplicar los cambios.

## 4. Configuración del Contenedor de Base de Datos

De manera similar, accedí a la consola del contenedor de base de datos para realizar la instalación y configuración de MariaDB.

Comencé con la actualización del sistema e instalación de los paquetes del servidor y cliente de MariaDB:

```
1 apt update && apt upgrade -y
2 apt install -y mariadb-server mariadb-client
```

Dado que el contenedor web se conecta de forma remota a la base de datos, modifiqué la configuración de MariaDB para que aceptara conexiones desde cualquier dirección IP:

```
1 sed -i 's/bind-address.*/bind-address = 0.0.0.0/' /etc/mysql/mariadb.conf.d/50-server.cnf
2 systemctl restart mariadb
```

## Estructura de la Base de Datos

La base de datos `blog_db` fue diseñada con dos tablas principales:

- **posts:** Almacena las entradas del blog. Sus campos son `id` (clave primaria de auto-incremento), `title` (título de la entrada), `content` (contenido del texto) y `created_at` (fecha y hora de creación).
- **personal\_info:** Contiene mis datos personales. Sus campos incluyen `id`, `name` (nombre completo), `email`, `bio` (una breve biografía), `image_path` (URL de la foto de perfil) y `pdf_url` (URL del informe en PDF).

Para la configuración final, creé un usuario llamado `blog_user` con su respectiva contraseña y le otorgué permisos completos sobre la base de datos `blog_db`.

## 5. Desarrollo y Despliegue del Código PHP

Para finalizar la implementación, volví al contenedor web para crear y subir el archivo `index.php`, que contiene toda la lógica del blog.

Este script PHP se encarga de:

- Establecer la conexión remota a la base de datos a través de la IP del contenedor de MariaDB.

- Implementar la funcionalidad para agregar y eliminar entradas del blog.
- Realizar consultas a la tabla `personal_info` para mostrar mis datos, incluyendo la imagen y el enlace al PDF alojados en un repositorio público de GitHub.
- Generar una interfaz de usuario sencilla y responsiva utilizando HTML y CSS embebidos.

Una vez subido el archivo, me aseguré de que el servidor web tuviera los permisos necesarios para acceder y ejecutar el código, ajustando la propiedad y los permisos del directorio `/var/www/html`:

```
1 chown -R www-data:www-data /var/www/html
2 chmod -R 755 /var/www/html
```

## 6. Pruebas y Validaciones

Para validar el correcto funcionamiento del sistema, realicé varias pruebas:

Verifiqué que **Apache2** estuviera activo y escuchando en el puerto 80 del contenedor web:

```
1 systemctl status apache2
2 netstat -tlnp | grep :80
```

Desde la consola del contenedor web, confirmé la conexión con el servidor de base de datos remoto:

```
1 mysql -h 172.16.90.117 -u blog_user -pblog_pass123 blog_db -e "SELECT name,
email FROM personal_info;"
```

El acceso externo al blog se habilitó posteriormente a la configuración interna, a través de una URL proporcionada por el profesor, que se encargó de configurar el enrutamiento necesario hacia la IP de mi contenedor web.

## 7. Capturas

Aquí se presentan algunas capturas de pantalla que documentan el estado de la implementación.



Type	Description	Disk usage %	Memory usage...	CPU usage
 lxc	114 (43562927A)	9.3 %	31.0 %	0.0% of 1 CPU
 lxc	117 (43562927DB)	12.9 %	60.1 %	0.0% of 1 CPU

Figura 1: Contenedores activos en Proxmox

```

root@43562927A:/# systemctl status apache2
* apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-07-31 18:55:37 UTC; 2h 8min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 18397 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=0/SUCCESS)
    Main PID: 17889 (apache2)
       Tasks: 7 (limit: 9332)
      Memory: 16.1M
         CPU: 402ms
    CGroup: /system.slice/apache2.service
            |-17889 /usr/sbin/apache2 -k start
            |-18401 /usr/sbin/apache2 -k start
            |-18402 /usr/sbin/apache2 -k start
            |-18403 /usr/sbin/apache2 -k start
            |-18404 /usr/sbin/apache2 -k start
            |-18405 /usr/sbin/apache2 -k start
            ~-18463 /usr/sbin/apache2 -k start

Jul 31 18:55:36 43562927A systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jul 31 18:55:37 43562927A systemd[1]: Started apache2.service - The Apache HTTP Server.
Jul 31 20:19:16 43562927A systemd[1]: Reloading apache2.service - The Apache HTTP Server...
Jul 31 20:19:16 43562927A systemd[1]: Reloaded apache2.service - The Apache HTTP Server.
root@43562927A:/#

```

Figura 2: Estado del servicio Apache en el contenedor web

The screenshot displays a web application interface for a personal blog. On the left, a user profile for Tobias Paz Posse is shown, including a circular profile picture, name, email (tobias.pazposse@alu.frt.utn.edu.ar), and affiliation (Estudiante de Sistemas - UTN FRT). Below the profile is a green button labeled 'Descargar Informe del Proyecto (PDF)'. To the right, there is a 'Nueva Entrada' (New Entry) form with fields for 'Título' (Title) and 'Contenido' (Content), and a blue 'Publicar' (Publish) button. Below the form, a section titled 'Entradas del Blog' (Blog Entries) shows a single entry titled 'Primera clase de virtualizacion' (First class of virtualization) dated 31/07/2025 22:30. The entry text mentions a first class on virtualization. A red 'Eliminar' (Delete) button is located at the bottom of the entry. The total number of entries is indicated as 1.

Figura 3: Vista del blog personal desde navegador

## 8. Conclusión

Este proyecto me permitió adquirir experiencia práctica en la gestión de contenedores LXC, la configuración de servicios web y de bases de datos, y la integración de todos estos componentes en una aplicación funcional. La separación de la lógica del blog y la base de datos en contenedores distintos demostró ser una solución eficiente, cumpliendo con los requisitos de la consigna y proporcionando un sistema optimizado y escalable.

*Tobías Paz Posse – 31 de julio de 2025*