

Plan Estratégico del Proyecto Final de Infraestructura

Fecha de creación: 2025-11-10

Resumen del Proyecto

Implementación de un entorno virtualizado con **Docker** y **Podman**, desplegando contenedores **Apache**, **MySQL** y **Nginx** sobre volúmenes persistentes gestionados con **RAID1** y **LVM**, documentando todo el proceso en un repositorio **Git**.

Objetivo General

Configurar un entorno funcional con tres contenedores (Apache, MySQL, Nginx) utilizando almacenamiento persistente sobre RAID y LVM, validando la persistencia de datos, el despliegue con Docker y Podman, y la documentación completa del proceso.

Fases del Proyecto y Porcentaje de Avance

Fase	Descripción	Porcentaje
0	Preparación inicial: estructura de carpetas, revisión de hardware, inicialización de repositorio Git.	0% → 5%
1	Configuración de 3 RAID1 con mdadm (uno por servicio).	5% → 25%
2	Creación de LVM sobre cada RAID, montaje en puntos dedicados y validación.	25% → 45%
3	Preparación de volúmenes para contenedores (permisos, estructura, pruebas de escritura).	45% → 55%
4	Creación de imágenes personalizadas (Dockerfile y Containerfile) y pruebas en Docker y Podman.	55% → 75%
5	Ejecución de contenedores vinculando volúmenes LVM y validación de persistencia.	75% → 90%
6	Documentación final, bitácora, subida al repositorio Git y preparación de la sustentación.	90% → 100%

Paso a Paso Detallado

Fase 0 — Preparación (0% → 5%)

- Crear estructura base del proyecto y repositorio Git.
- Verificar discos disponibles con `lsblk`.
- Estructura de carpetas:

```
proyecto/
├─ docs/
├─ dockerfiles/{apache,mysql,nginx}/
├─ containerfiles/{apache,mysql,nginx}/
└─ scripts/
```

Fase 1 — RAID (5% → 25%)

- Crear 3 RAID1 con `mdadm`:

```
sudo mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/sdb
/dev/sdc
```

- Verificar con `cat /proc/mdstat` y `mdadm --detail`.
- Registrar configuración en `/etc/mdadm/mdadm.conf`.

Fase 2 — LVM (25% → 45%)

- Crear PV, VG, LV sobre cada RAID.

```
pvccreate /dev/md0
vgcreate vg_web /dev/md0
lvcreate -n lv_web -L 20G vg_web
mkfs.ext4 /dev/vg_web/lv_web
```

- Montar volúmenes y registrar en `/etc/fstab`.

Fase 3 — Volúmenes y permisos (45% → 55%)

- Crear rutas de persistencia: `/mnt/lv_web/www`, `/mnt/lv_db/mysql-data`, `/mnt/lv_nginx/conf-or-logs`.
- Asignar permisos y probar escritura.

Fase 4 — Imágenes personalizadas (55% → 75%)

- Crear `Dockerfile` y `Containerfile` para cada servicio.
- Construir con Docker y Podman:

```
docker build -t proyecto_apache:1.0 ./dockerfiles/apache
podman build -t proyecto_apache:1.0 -f
containerfiles/apache/Containerfile .
```

Fase 5 — Despliegue y pruebas (75% → 90%)

- Ejecutar contenedores vinculando volúmenes:

```
docker run -d --name apache_proy -p 8080:80 -v /mnt/lv_web/www:/usr/local/apache2/htdocs proyecto_apache:1.0
docker run -d --name mysql_proy -p 3306:3306 -v /mnt/lv_db/mysql-data:/var/lib/mysql proyecto_mysql:1.0
docker run -d --name nginx_proy -p 8081:80 -v /mnt/lv_nginx/conf-or-logs:/etc/nginx/conf.d proyecto_nginx:1.0
```

- Validar acceso web y persistencia tras reinicios.

Fase 6 — Documentación y entrega (90% → 100%)

- Documentar todo en `docs/bitacora.md` con capturas, comandos y resultados.
- Subir repositorio con:
 - o Dockerfile/Containerfile
 - o Scripts
 - o Bitácora y README
 - o Presentación del proyecto

📊 Métricas de Control y Evaluación

Indicador	Meta	Evidencia
RAID configurado correctamente	100% funcionalidad RAID	<code>cat /proc/mdstat</code> muestra sincronización completa
LVM sobre RAID montado	3 volúmenes activos	<code>lvs</code> y <code>mount</code>
Imágenes personalizadas construidas	3 imágenes válidas	<code>docker images</code> y <code>podman images</code>
Contenedores funcionales	3 contenedores corriendo	<code>docker ps</code> / <code>podman ps</code>
Persistencia validada	Archivos permanecen tras reinicio	Pruebas prácticas
Repositorio Git completo	Documentación + código + capturas	Revisión final

⚠ Riesgos y Medidas de Mitigación

Riesgo	Mitigación
Falta de discos para RAID	Simular discos con archivos loopback
Problemas SELinux con volúmenes	Usar <code>:Z</code> o <code>chcon</code> para ajustar contexto
Errores de permisos MySQL	Ajustar UID/GID y permisos antes de iniciar contenedor
Incompatibilidad Docker–Podman	Probar ambos y registrar diferencias en bitácora

Entregables Finales

- `docs/bitacora.md` con capturas, comandos y pruebas.
- `dockerfiles/` y `containerfiles/` (Apache, MySQL, Nginx).
- `scripts/` automatizando RAID, LVM y despliegue.

- `README.md` con guía de ejecución.
- `presentacion.pdf` o `.pptx` para sustentación.

Proyecto: Infraestructura con Docker, Podman, RAID y LVM

Porcentaje total al finalizar: 100% completado