Instalación y Configuración del Sistema Operativo

Documento 03 — Infraestructura de Servidores

Dozzier ERP — DozzierS.A.

Cliente: Dozzier ERP Proveedor: DozzierS.A. Versión: 1.0

Fecha: Febrero 2026

Clasificación: Confidencial

1. Introducción

El presente documento describe los procedimientos de instalación, configuración y aseguramiento del sistema operativo utilizado en la infraestructura de servidores de **Dozzier ERP**. La plataforma opera sobre **Ubuntu 24.04.3 LTS** (Noble Numbat) en modalidad headless, desplegada sobre instancias de cómputo en la nube de **DigitalOcean**.

Se detalla la configuración de cada uno de los tres servidores que componen el entorno: producción, contingencia y stage/QA. Cada servidor sigue lineamientos de seguridad, rendimiento y mantenibilidad acordes con las mejores prácticas para entornos empresariales de misión crítica.

2. Alcance

Este documento aplica a los tres servidores que conforman la infraestructura de Dozzier ERP:

Servidor	Propósito	Especificación
Producción	Operación principal del ERP	8 vCPU, 16 GB RAM
Contingencia	Réplica de producción (failover)	8 vCPU, 16 GB RAM
Stage/QA	Pruebas y validación previo a despliegue	4 vCPU, 8 GB RAM

Los tres servidores comparten la misma base de sistema operativo, paquetes esenciales y políticas de seguridad. Las diferencias radican en las especificaciones de recursos y las configuraciones específicas de cada rol.

3. Arquitectura General

3.1 Plataforma en la Nube

La infraestructura se aloja en **DigitalOcean**, utilizando Droplets (máquinas virtuales) como unidades de cómputo. DigitalOcean provee:

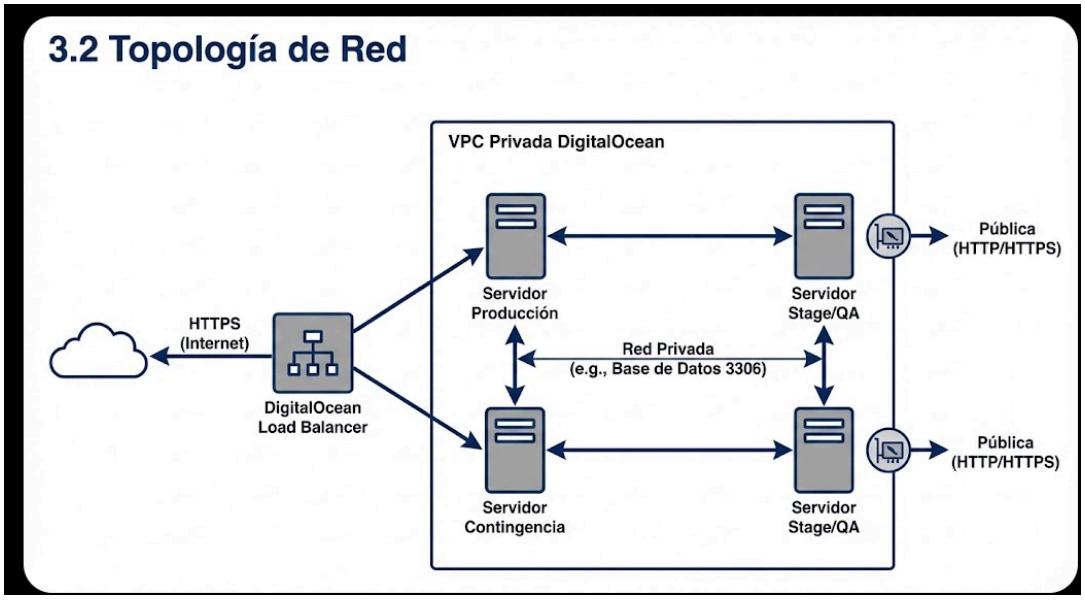
- **Alta disponibilidad** a nivel de centro de datos con SLA de 99.99% de uptime.
- **Escalabilidad vertical** para ajustar recursos de CPU y memoria según demanda.
- **Snapshots y backups** automatizados a nivel de plataforma.

- **Red privada virtual (VPC)** para comunicación segura entre servidores.

3.2 Topología de Red

Los tres servidores se comunican a través de una **VPC privada de DigitalOcean**, lo que garantiza que el tráfico interno no transita por la red pública. La topología es la siguiente:

- Cada Droplet posee una interfaz de red pública (para tráfico HTTP/HTTPS) y una interfaz privada (para comunicación interna).
- El tráfico de base de datos (puerto 3306) circula exclusivamente por la red privada.
- Un **DigitalOcean Load Balancer** puede distribuir el tráfico entrante hacia los servidores de producción y contingencia cuando se requiera balanceo de carga.



3.3 Sistema Operativo Base

Parámetro	Valor
Distribución	Ubuntu 24.04.3 LTS (Noble Numbat)
Kernel	6.8.0-71-generic
Arquitectura	x86_64 (amd64)
Shell predeterminado	Bash 5.2
Entorno gráfico	Ninguno (headless)
Ciclo de soporte	Hasta abril de 2029 (estándar), hasta 2036 (ESM)

La elección de Ubuntu LTS garantiza un ciclo de soporte prolongado con actualizaciones de seguridad regulares, amplia compatibilidad con paquetes empresariales y una extensa base de conocimiento para resolución de incidentes.

4. Servidor de Producción

4.1 Especificaciones de Hardware Virtual

El servidor de producción está dimensionado para atender aproximadamente **300 usuarios concurrentes** con la carga típica de un sistema ERP transaccional.

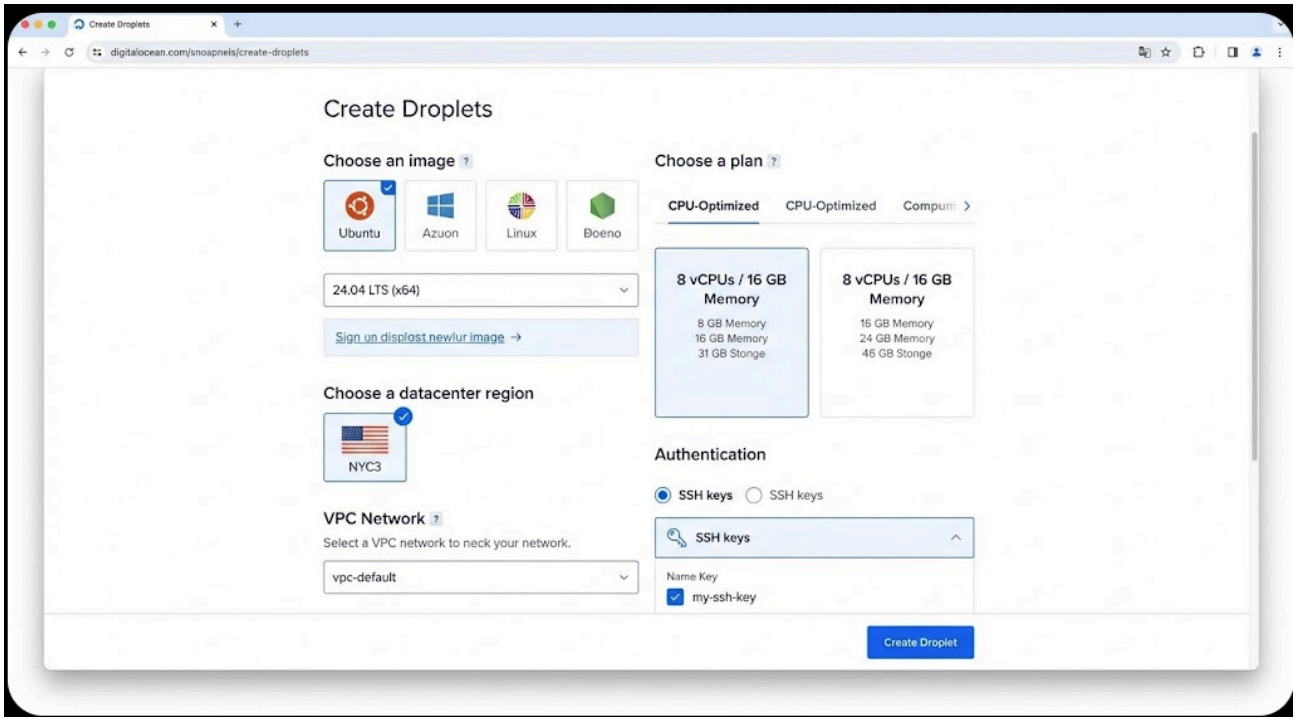
Recurso	Especificación
vCPU	8 núcleos
RAM	16 GB
Almacenamiento	SSD NVMe (tamaño según volumen de datos)
Ancho de banda	Según plan de DigitalOcean
Región	Seleccionada por proximidad geográfica al cliente

> **Nota:** Estas especificaciones pueden escalarse verticalmente (resize del Droplet) sin necesidad de reinstalar el sistema operativo. DigitalOcean permite aumentar CPU y RAM con una breve ventana de reinicio.

4.2 Instalación de Ubuntu 24.04 LTS

La creación del Droplet se realiza desde el panel de control de DigitalOcean o mediante su API. Los parámetros de creación son:

1. **Seleccionar imagen:** Ubuntu 24.04 (LTS) x64.
2. **Plan de Droplet:** Premium Intel/AMD — 8 vCPU, 16 GB RAM.
3. **Región:** Centro de datos más cercano al usuario final.
4. **VPC:** Asignar a la VPC privada del proyecto Dozzier.
5. **Autenticación:** Clave SSH (no contraseña).
6. **Nombre del host:** `dozzier-prod-01`.
7. **Tags** `dozzier`, `production`, `erp`.



4.2.1 Verificación Post-creación

Una vez creado el Droplet, se verifica la instalación conectando por SSH:

```
ssh root@<IP_PUBLICA_PRODUCCION>
```

Verificaciones iniciales:

```
# Versión del sistema operativo lsb_release -a
# Resultado esperado: Ubuntu 24.04.3 LTS

# Versión del kernel uname -r
# Resultado esperado: 6.8.0-71-generic

#Arquitectura
uname -m
# Resultado esperado: x86_64

# Shell
echo $BASH_VERSION
# Resultado esperado: 5.2.x
```

4.2.2 Actualización Inicial del Sistema

Antes de instalar paquetes adicionales, se actualiza el sistema por completo:

```
apt update && apt upgrade -y apt dist-
upgrade -y
apt autoremove -y
```

Se recomienda reiniciar el servidor tras la actualización del kernel:

```
reboot
```

4.3 Configuración de Red y Firewall (UFW)

4.3.1 Interfaces de Red

DigitalOcean asigna automáticamente las interfaces de red al crear el Droplet dentro de una VPC:

Interfaz	Tipo	Propósito
`eth0`	Pública	Tráfico HTTP/HTTPS entrante
`eth1`	Privada (VPC)	Comunicación entre servidores

Verificación:

```
ip addr show
```

4.3.2 Configuración de UFW (Uncomplicated Firewall)

UFW se utiliza como capa de firewall a nivel del sistema operativo. Las reglas se configuran de la siguiente manera:

```
# Habilitar UFW (si no está activo) ufw enable

# Política por defecto: denegar tráfico entrante, permitir saliente ufw default deny incoming
ufw default allow outgoing

# Permitir SSH (puerto 22) ufw allow
22/tcp

# Permitir HTTP (puerto 80) ufw allow
80/tcp

# Permitir HTTPS (puerto 443) ufw allow
443/tcp

# Permitir MySQL SOLO desde la red privada (VPC) ufw allow from
10.116.0.0/20 to any port 3306

# Verificar reglas activas ufw status
verbose
```

Puerto	Protocolo	Origen	Propósito
22	TCP	Cualquiera	Acceso SSH
80	TCP	Cualquiera	Tráfico HTTP (redirección a HTTPS)
443	TCP	Cualquiera	Tráfico HTTPS
3306	TCP	Solo red privada (VPC)	MySQL interno

Importante: El rango de red privada `10.116.0.0/20` corresponde al segmento VPC asignado por DigitalOcean. Este valor debe ajustarse al rango real configurado en el proyecto.

4.4 Instalación de Docker y Docker Compose

La plataforma Dozzier ERP opera en contenedores Docker. Se instala Docker Engine y Docker Compose desde los repositorios oficiales.

4.4.1 Instalación de Paquetes Prerequisito

```
apt install -y ca-certificates curl gnupg lsb-release
```

4.4.2 Agregar Repositorio Oficial de Docker

```
# Agregar clave GPG oficial
install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg \
    -o /etc/apt/keyrings/docker.asc chmod a+r
/etc/apt/keyrings/docker.asc

# Agregar repositorio echo \
    "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
    https://download.docker.com/linux/ubuntu \
    $(. /etc/os-release && echo "${VERSION_CODENAME}") stable" | \ tee
    /etc/apt/sources.list.d/docker.list > /dev/null

apt update
```

4.4.3 Instalación de Docker Engine y Compose

```
apt install -y docker-ce docker-ce-cli containerd.io \ docker-buildx-plugin
    docker-compose-plugin
```

Verificación:

```
docker --version
# Resultado esperado: Docker version 27.x.x

docker compose version
# Resultado esperado: Docker Compose version v2.x.x
```

4.4.4 Configuración del Servicio Docker

Se habilita Docker para inicio automático con el sistema:

```
systemctl enable docker
systemctl start docker
```

4.5 Instalación de Paquetes Adicionales

Además de Docker, se instalan las siguientes utilidades requeridas por la operación del sistema:

```
apt install -y git openssl curl wget htop \ unzip software-
    properties-common
```

Paquete	Propósito
`git`	Control de versiones para despliegues
`openssl`	Gestión de certificados y cifrado
`curl` / `wget`	Transferencia de datos y descargas
`htop`	Monitoreo de recursos en tiempo real
`unzip`	Descompresión de archivos

4.6 Configuración de Usuarios y Permisos

4.6.1 Crear Usuario de Despliegue

Se crea un usuario dedicado para operaciones de despliegue, evitando el uso directo de `root`:

```
# Crear usuario
adduser --disabled-password --gecos "Deploy User" deploy

# Agregar a grupo sudo usermod
-aG sudo deploy

# Agregar a grupo docker (para ejecutar contenedores sin sudo) usermod -aG docker
deploy
```

4.6.2 Configurar Acceso SSH para el Usuario

```
# Crear directorio SSH mkdir -p
/home/deploy/.ssh chmod 700
/home/deploy/.ssh

# Copiar clave pública autorizada
cp /root/.ssh/authorized_keys /home/deploy/.ssh/authorized_keys chmod 600
/home/deploy/.ssh/authorized_keys
chown -R deploy:deploy /home/deploy/.ssh
```

4.6.3 Deshabilitar Acceso Root por SSH

Una vez confirmado el acceso con el usuario `deploy`, se deshabilita el login de root:

```
# Editar configuración SSH
sed -i 's/^PermitRootLogin yes/PermitRootLogin no/' /etc/ssh/sshd_config

# Reiniciar servicio SSH
systemctl restart sshd
```

5. Servidor de Contingencia

5.1 Propósito

El servidor de contingencia actúa como réplica del entorno de producción, permitiendo una transición rápida (failover) en caso de fallo del servidor principal. Su configuración es un espejo exacto del servidor de producción.

5.2 Especificaciones

Recurso	Especificación
vCPU	8 núcleos
RAM	16 GB
Almacenamiento	SSD NVMe (idéntico a producción)
Nombre del host	`dozzier-contingency-01`
Tags	`dozzier`, `contingency`, `erp`

5.3 Configuración

El procedimiento de configuración es **idéntico** al del servidor de producción (secciones 4.2 a 4.6). Se aplican las mismas versiones de paquetes, reglas de firewall, usuarios y permisos.

Adicionalmente, se configura:

- **Sincronización de datos:** Replicación de la base de datos desde producción mediante mecanismos de replicación MySQL o snapshots periódicos de DigitalOcean.
- **Monitoreo de disponibilidad:** Scripts de health-check que verifican el estado del servidor de producción y alertan al equipo de operaciones ante indisponibilidad.
- **DNS y failover:** Configuración de registros DNS con TTL bajo para facilitar la conmutación de tráfico hacia el servidor de contingencia.

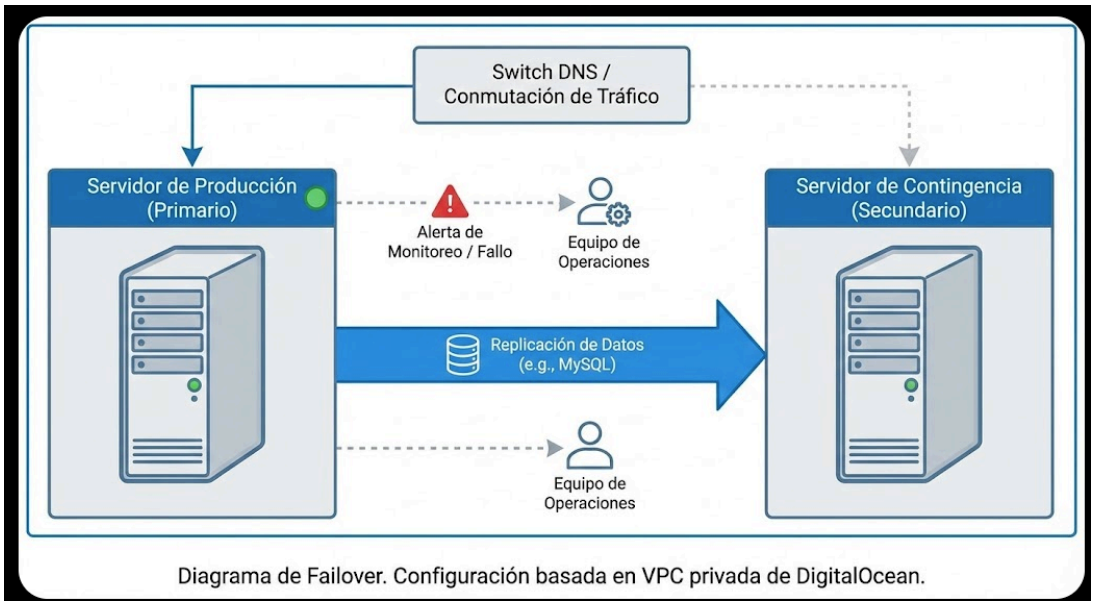


Diagrama de Failover. Configuración basada en VPC privada de DigitalOcean.

6. Servidor de Stage/QA

6.1 Propósito

El servidor de Stage/QA proporciona un entorno aislado para pruebas de calidad, validación de nuevas funcionalidades y ensayo de procedimientos de despliegue antes de su aplicación en producción.

6.2 Especificaciones

Recurso	Especificación
vCPU	4 núcleos
RAM	8 GB
Almacenamiento	SSD NVMe (menor que producción)
Nombre del host	`dozzier-stage-01`
Tags	`dozzier, stage, qa`

6.3 Configuración

El procedimiento base es el mismo que el del servidor de producción (secciones 4.2 a 4.6), con las siguientes diferencias:

- **Recursos reducidos:** 4 vCPU y 8 GB RAM, suficientes para pruebas funcionales con carga limitada.
- **Datos de prueba:** La base de datos contiene datos anonimizados o de prueba, nunca datos reales de producción.
- **Acceso restringido:** Solo el equipo de desarrollo y QA tiene acceso SSH. Se pueden agregar restricciones adicionales de IP en las reglas de UFW.

```
# Ejemplo: restringir SSH solo a IPs del equipo de desarrollo ufw delete allow 22/tcp
ufw allow from <IP_EQUIPO_1> to any port 22 ufw allow
from <IP_EQUIPO_2> to any port 22
```

7. Seguridad del Sistema Operativo

7.1 Acceso SSH Exclusivo por Clave

La autenticación por contraseña se deshabilita en todos los servidores. Solo se permite acceso mediante claves SSH:

```
# Verificar/establecer en /etc/ssh/sshd_config PasswordAuthentication no
PubkeyAuthentication yes
PermitRootLogin no
MaxAuthTries 3
```

Reiniciar el servicio después de los cambios:

```
systemctl restart sshd
```

7.2 Actualizaciones Automáticas de Seguridad

Se configura `unattended-upgrades` para aplicar parches de seguridad automáticamente:

```
apt install -y unattended-upgrades
dpkg-reconfigure -plow unattended-upgrades
```

Verificar la configuración en `/etc/apt/apt.conf.d/50unattended-upgrades`:

```
Unattended-Upgrade::Allowed-Origins {
    "${distro_id}:${distro_codename}-security";
    "${distro_id}ESMApps:${distro_codename}-apps-security";
};
Unattended-Upgrade::AutoFixInterruptedDpkg "true";
Unattended-Upgrade::Remove-Unused-Dependencies "true";
```

7.3 Fail2ban

Fail2ban protege contra ataques de fuerza bruta al servicio SSH y otros servicios expuestos:

```
apt install -y fail2ban
```

Crear configuración local en `/etc/fail2ban/jail.local`:

```
[DEFAULT]
bantime  = 3600
findtime  = 600
maxretry  = 3

[sshd]
enabled = true port
        = 22
filter  = sshd
logpath = /var/log/auth.log maxretry =
3
```

Activar el servicio:

```
systemctl enable fail2ban systemctl
start fail2ban
```

Verificar estado:

```
fail2ban-client status sshd
```

7.4 Resumen de Reglas de Firewall

La siguiente tabla consolida las reglas de firewall aplicadas a todos los servidores:

Puerto	Protocolo	Dirección	Origen	Descripción
22	TCP	Entrante	Cualquiera (o IPs restringidas en Stage)	Acceso SSH
80	TCP	Entrante	Cualquiera	HTTP (redirección a HTTPS)
443	TCP	Entrante	Cualquiera	HTTPS (tráfico cifrado)
3306	TCP	Entrante	Solo VPC privada	MySQL (comunicación interna)
Todos	TCP/UDP	Saliente	Cualquiera	Tráfico saliente permitido

7.5 Endurecimiento Adicional

Se aplican las siguientes medidas complementarias de seguridad:

- **Deshabilitar servicios innecesarios:** Ningún entorno de escritorio ni servicio gráfico se instala. Solo se ejecutan los servicios mínimos requeridos.
- **Limitar el uso de sudo:** Solo los usuarios del grupo `sudo` pueden elevar privilegios. Se registra toda actividad de sudo en los logs del sistema.

- **Protección del kernel:** Se configuran parámetros de sysctl para mitigar ataques comunes:

```
# /etc/sysctl.d/99-security.conf

# Deshabilitar respuestas a broadcast ICMP net.ipv4.icmp_echo_ignore_broadcasts = 1

# Protección contra SYN flood
net.ipv4.tcp_syncookies = 1

# Deshabilitar redirecciones ICMP net.ipv4.conf.all.accept_redirects
= 0
net.ipv4.conf.default.accept_redirects = 0

# Deshabilitar IP source routing net.ipv4.conf.all.accept_source_route = 0
net.ipv4.conf.default.accept_source_route = 0

# Habilitar protección contra IP spoofing net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1
```

Aplicar los cambios:

```
sysctl --system
```

8. Certificados TLS

El cifrado del tráfico HTTPS se gestiona mediante una de las siguientes opciones:

8.1 Opción A: DigitalOcean Load Balancer

Cuando se utiliza el balanceador de carga de DigitalOcean, la terminación TLS ocurre en el Load Balancer:

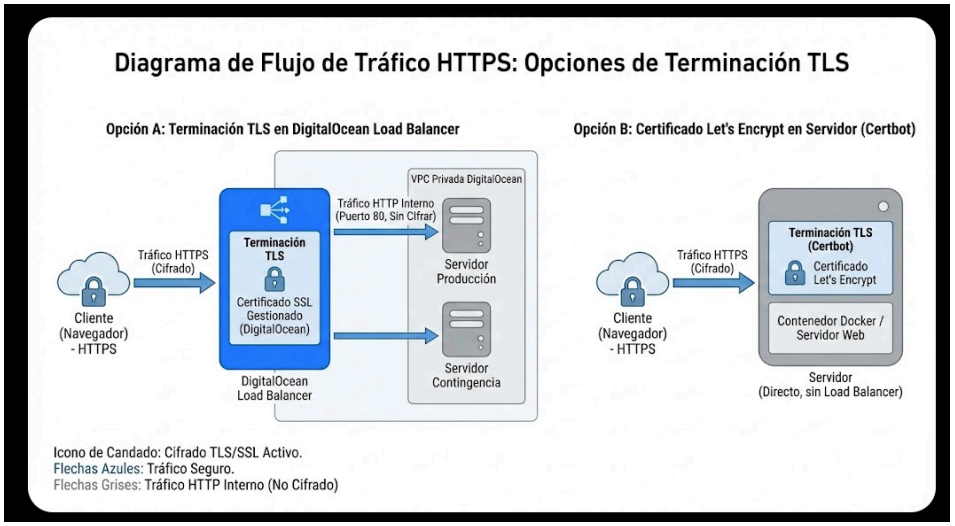
- El certificado SSL se gestiona directamente desde el panel de DigitalOcean.
- El Load Balancer distribuye tráfico hacia los servidores de producción y contingencia.
- Los servidores reciben tráfico HTTP internamente (puerto 80), simplificando la configuración.

8.2 Opción B: Let's Encrypt (Certbot)

Si no se utiliza Load Balancer, se puede instalar Certbot para obtener certificados gratuitos de Let's Encrypt:

```
apt install -y certbot
```

La gestión del certificado se coordina con la configuración del servidor web dentro de los contenedores Docker (documentado en el documento de despliegue correspondiente).



9. Balanceo de Carga

Para un estimado de **300 usuarios concurrentes**, se contempla la utilización del **DigitalOcean Load Balancer** como mecanismo de distribución de tráfico:

- **Algoritmo:** Round-robin o menor número de conexiones.
- **Health checks:** Verificación periódica del estado de los servidores backend (HTTP 200 en endpoint de salud).
- **Servidores backend:** Producción y contingencia registrados como nodos del balanceador.
- **Sticky sessions:** Configurable según los requerimientos de la aplicación.

Esta capa de balanceo permite:

- Distribuir la carga entre los servidores disponibles.
- Redirigir automáticamente el tráfico si un servidor no responde a los health checks. Escalar
- horizontalmente agregando nuevos servidores backend sin cambios en DNS.

Nota: La implementación detallada del Load Balancer se documenta en el entregable correspondiente de arquitectura de red.

10. Conclusiones

La infraestructura de sistema operativo de Dozzier ERP se fundamenta en:

- **Ubuntu 24.04 LTS** como base estable y con soporte a largo plazo.
- **Tres servidores** con roles diferenciados: producción, contingencia y stage/QA.
- **Contenedorización con Docker** para aislar servicios y facilitar despliegues reproducibles.
- **Seguridad multicapa** mediante UFW, SSH por claves, fail2ban, actualizaciones automáticas y endurecimiento del kernel.
- **Red privada (VPC)** para aislar la comunicación interna entre servidores.
- **Cifrado TLS** gestionado externamente para proteger todo el tráfico de los usuarios.

Esta configuración proporciona un entorno robusto, seguro y escalable, alineado con las necesidades operativas del sistema ERP y las mejores prácticas de la industria para infraestructura en la nube.

Este documento es propiedad confidencial de DozzierS.A. Su distribución, copia o divulgación sin autorización expresa está prohibida.