

An abstract, high-contrast image featuring a vibrant red ink splash or smoke plume against a dark, almost black background. The red liquid is captured in mid-motion, creating intricate, swirling patterns and droplets that trail downwards. The overall effect is dynamic and visually striking.

Guía completa: Monta n8n en tu propio servidor (Clouding) con dominio y HTTPS

Objetivo: levantar n8n en 30–60 min en un servidor económico, con dominio propio y SSL, y terminar con un webhook de producción funcional. Esta guía está pensada para principiantes (sin programación) pero con buena precisión técnica.

Antes de empezar (qué es n8n y por qué en servidor propio)

n8n es una herramienta de automatización visual (workflows con nodos) tipo "Zapier/Make", pero auto-alojada y ampliable.

Servidor propio = costes predecibles y bajos (≈ 6–7 €/mes) y sin límites duros de "tareas/operaciones".

¿Qué vas a montar?

- Una VM en Clouding (Ubuntu 24.04)
- Docker para desplegar n8n
- Caddy como proxy inverso (emite certificados Let's Encrypt automáticamente)
- Tu dominio (subdominio) apuntando a la VM
- Un webhook de prueba en producción



Crea la cuenta en Clouding + saldo promocional (5€)

01

Regístrate en Clouding

Recarga 5 € (mínimo)

02

Contacta con soporte

Escribe a soporte@clouding.io indicando que vienes de Víctor Senovilla (Torr)

03

Recibe el bono

Cuando verifiquen la recarga, te añaden +5 € extra

 **Nota:** Clouding bloquea SMTP saliente por defecto (no afecta a esta guía; para emails usa APIs de Gmail/MailerLite/AC).

(Nuevo paso) Configura el DNS de tu dominio (registro tipo A)

Hazlo antes de instalar nada para que, cuando arranque el proxy, el certificado SSL se emita al instante.

1

Elige un subdominio para n8n

Ejemplo: `n8n.tudominio.com`

2

Crea el registro DNS

En tu proveedor DNS (donde tengas el dominio), crea:

- **Tipo:** A
- **Nombre/Host:** n8n
- **Contenido/IP:**
IP_PÚBLICA_DE_TU_SERVIDOR
(la verás al crear la VM; ej.
200.234.228.71)
- **TTL:** 300s o por defecto

3

Espera la propagación

Espera 1–10 min. (propaga rápido; si tarda más, es normal)

Por qué ahora: Caddy validará el dominio con Let's Encrypt mediante http-01. Si el DNS ya apunta, el SSL se emite sin fricción.

Crea el servidor en Clouding

Especificaciones recomendadas:

- Ubuntu 24.04 (64-bit)
- 1 vCore • 2 GB RAM • 15 GB SSD
- **Coste aproximado:** 0,0089 €/hora (~6,53 €/mes)

📌 **No abras 5678** si vas a usar dominio y proxy (no es necesario). 5678 solo se usa para acceso directo temporal.

Firewall en Clouding

(Mis Firewalls → default o uno nuevo):

Entrantes (INBOUND): abrir TCP 22 (SSH), TCP 80 (HTTP), TCP 443 (HTTPS).

Asigna ese perfil de firewall a tu servidor.



Conexión por SSH (Windows/Mac) y avisos de teclado

Windows

1. Instala Git for Windows → abre Git Bash
2. Conecta: `ssh root@IP_DEL_SERVIDOR`
3. La primera vez: escribe `yes` para aceptar la fingerprint
4. **Pegar contraseña:** `Shift + Insert` (no `Ctrl+V`)
5. Al escribir/pegar la contraseña no verás el cursor moverse (es normal)

macOS

1. Abre Terminal
2. Conecta: `ssh root@IP_DEL_SERVIDOR`
3. Acepta `yes` la fingerprint
4. **Pegar contraseña:** `Cmd + V` (el cursor tampoco se mueve; es normal)

Instala Docker + Docker Compose

Estás como root. Si no, antepone `sudo`.

```
apt-get update && apt-get upgrade -y  
apt-get install -y ca-certificates curl gnupg
```

```
install -m 0755 -d /etc/apt/keyrings  
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | gpg --dearmor -o /etc/apt/keyrings/docker.gpg  
chmod a+r /etc/apt/keyrings/docker.gpg  
echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg]  
https://download.docker.com/linux/ubuntu $(. /etc/os-release && echo $VERSION_CODENAME) stable" | tee  
/etc/apt/sources.list.d/docker.list > /dev/null
```

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

```
systemctl enable --now docker  
docker --version && docker compose version
```


Carpeta de proyecto y variables de entorno

Paso 1: Crear directorios

```
mkdir -p /opt/n8n/data /opt/caddy
cd /opt/n8n
```

Paso 2: Generar clave de cifrado

```
# Clave de cifrado para credenciales de n8n (guárdala):
openssl rand -hex 24
#ej: 8565914f707342a6566ec4b77e1275754cd1bfd22a0a2802
```

Paso 3: Crear */opt/n8n/.env*

(reemplaza por tus datos):

```
nano /opt/n8n/.env

# Dominio y protocolo (imprescindible para URLs correctas y cookies seguras)
N8N_HOST=n8n.tudominio.com
N8N_PROTOCOL=https
WEBHOOK_URL=https://n8n.tudominio.com/

# Zona horaria (afecta a cron, timestamps, etc.)
GENERIC_TIMEZONE=Europe/Madrid

# Seguridad: cifra credenciales guardadas por n8n
N8N_ENCRYPTION_KEY=PEGA_AQUI_LA_CLAVE_DEL_OPENSSL

# Telemetría/avisos (opcional)
N8N_DIAGNOSTICS_ENABLED=false
N8N_VERSION_NOTIFICATIONS_ENABLED=false

# Cookies seguras (true cuando hay HTTPS)
N8N_SECURE_COOKIE=true
```

 **Importante:** no dupliques variables (si hay dos N8N_HOST, n8n puede tomar la última). Cada clave una sola vez.

Paso 4: Proxy inverso (Caddy) y docker-compose

/opt/caddy/Caddyfile

```
mkdir -p /opt/n8n/caddy
cat > /opt/n8n/caddy/Caddyfile << 'EOF'
n8n.tudominio.com {
  encode zstd gzip
  reverse_proxy n8n:5678
}
EOF
```

```
ls -la /opt/n8n/caddy/Caddyfile
file /opt/n8n/caddy/Caddyfile
```

nano /opt/n8n/docker-compose.yml

```
services:
  n8n:
    image: n8nio/n8n:latest # fija una versión estable si quieres
    container_name: n8n
    restart: unless-stopped
    environment:
      N8N_HOST: ${N8N_HOST}
      N8N_PORT: 5678
      N8N_PROTOCOL: ${N8N_PROTOCOL}
      WEBHOOK_URL: ${WEBHOOK_URL}
      GENERIC_TIMEZONE: ${GENERIC_TIMEZONE}
      N8N_ENCRYPTION_KEY: ${N8N_ENCRYPTION_KEY}
      N8N_DIAGNOSTICS_ENABLED: ${N8N_DIAGNOSTICS_ENABLED}
      N8N_VERSION_NOTIFICATIONS_ENABLED: ${N8N_VERSION_NOTIFICATIONS_ENABLED}
      N8N_USER_FOLDER: /home/node/.n8n
      N8N_SECURE_COOKIE: ${N8N_SECURE_COOKIE}
    volumes:
      - ./data:/home/node/.n8n
    networks:
      - n8n-net

  caddy:
    image: caddy:latest
    container_name: caddy
    restart: unless-stopped
    ports:
      - "80:80"
      - "443:443"
    volumes:
      - ./caddy/Caddyfile:/etc/caddy/Caddyfile:ro
      - caddy-data:/data
    networks:
      - n8n-net

networks:
  n8n-net:

volumes:
  caddy-data:
```

Levanta el stack y prueba tu primer webhook

Levanta el stack y emite el certificado

```
cd /opt/n8n
docker compose pull
docker compose up -d
docker compose logs -f caddy
```

Espera el mensaje `certificate obtained successfully` para `n8n.tudominio.com`. Si tarda, revisa que el registro A existe y el firewall tiene 80/443 abiertos.

SI TE APARECE UN ERROR COMO EN EL VIDEO, mira la siguiente diapositiva y luego vuelve aquí

Primer acceso y alta del Owner

Entra en:

<https://n8n.tudominio.com/>

Verás `/setup` para crear el usuario Owner (email + password).

Si te muestra aviso de secure cookie es que entras por http o falta el proxy/SSL: vuelve a revisar los pasos 2, 6 y 7.

Truco: si el flujo de setup se atasca por caché, abre en ventana privada.

Smoke test: tu primer webhook de producción

Crea un workflow nuevo y añade Webhook:

- **HTTP Method:** GET
- **Path:** test-webhook (sin "/" inicial)
- **Respond:** Immediately

Publish el workflow (arriba a la derecha). Si aparece el botón "Publish", aún no está publicado. Pulsa y confirma.

Copia la Production URL que muestra el nodo (debe verse en https), por ejemplo:

```
https://n8n.tudominio.com/webhook/test-webhook
```

Prueba desde terminal o navegador:

```
curl -i -X GET
"https://n8n.tudominio.com/webhook/test-webhook"
```

Resultado esperado: HTTP/2 200 (o 204 si no devuelves cuerpo).

Ojo: `curl -I` (HEAD) puede devolver 404 si tu webhook espera GET. Usa `-X GET`.

¡Listo! Ya puedes construir flujos reales.

Error

Si te sale el mismo error que a mi en el video, lo que tienes que hacer para solucionarlo es lo siguiente:

Paso 1 – Parar n8n (para no pelear con el volumen)

```
docker compose stop n8n
```

Paso 2 – Arreglar permisos del directorio de datos

```
chown -R 1000:1000 /opt/n8n/data  
chmod -R u+rwX /opt/n8n/data
```

Paso 3 – Levantar n8n otra vez

```
docker compose up -d n8n
```

Paso 4 – Confirmar que ya no reinicia

```
docker compose ps  
docker compose logs --tail=40 n8n
```


Prompt de contexto

Con este prompt chatgpt comprenderá qué estás haciendo y solo deberás añadir tu problema.

Este Prompt solo sirve para el servidor. No para el uso de n8n. Sólo si tienes problemas en la parte específica de montar el servidor

Contexto técnico (léelo antes de enviar):

Estoy montando n8n (herramienta de automatización) en mi propio servidor para no depender de Zapier o Make.

Mi infraestructura es esta:

Un servidor VPS en Clouding con Ubuntu 24.04

Uso Docker y Docker Compose (no sé Docker, solo sigo pasos)

Tengo dos contenedores:

- n8n (la aplicación)

- Caddy (proxy inverso + HTTPS automático con Let's Encrypt)

n8n corre internamente en el puerto 5678

Accedo a n8n mediante un dominio (ej: n8n.midominio.com)

El dominio apunta por DNS (registro A) a la IP del servidor

Caddy recibe el tráfico en HTTPS (443) y lo redirige a n8n

n8n guarda datos en un volumen persistente (/opt/n8n/data)

Variables importantes que uso:

N8N_HOST = mi dominio

N8N_PROTOCOL = https

WEBHOOK_URL = https://mi-dominio/

N8N_SECURE_COOKIE = true

Lo que NO soy:

No soy programador

No quiero soluciones avanzadas o teóricas

Cómo quiero que me ayudes:

Explícame el problema en lenguaje simple

Dime qué comando exacto ejecutar o qué archivo tocar

Dime por qué pasa y cómo evitarlo en el futuro

Si es un error típico de permisos, redirecciones, puertos o DNS, indícalo claramente

Mi problema actual es este:

[AQUÍ PEGO EL ERROR, LOG O COMPORTAMIENTO]

Respóndeme paso a paso, como si estuvieras ayudando a alguien que está aprendiendo ingeniería de procesos, no sistemas.

Problemas frecuentes, seguridad y checklist final

Problemas frecuentes y soluciones en 1 minuto

No carga el dominio / no hay candado verde

Revisa DNS A (n8n.tudominio.com → IP) y que el firewall tenga 80/443 abiertos.

Reinicia: `docker compose down && docker compose up -d` y mira `docker compose logs -f caddy`.

Pantalla "secure cookie"

Estás en http. Asegura `N8N_PROTOCOL=https`, `WEBHOOK_URL=https://.../`, `N8N_SECURE_COOKIE=true` y que Caddy esté activo. Reinicia el stack.

404 en /webhook/...

El workflow no está Published o no copiaste la Production URL. El método no coincide (HEAD vs GET). Prueba `-X GET`.

Contenedor n8n reinicia en bucle

Permisos del volumen:

```
chown -R 1000:1000 /opt/n8n/data
chmod -R 700 /opt/n8n/data
docker compose up -d && docker compose logs -f n8n
```

Variables no se aplican

Comprueba qué ve el contenedor:

```
docker exec -it n8n env | egrep
'N8N_HOST|N8N_PROTOCOL|WEBHOOK_URL|N8N_SECURE_COOKIE'
```

Si ves valores antiguos, revisa `/opt/n8n/.env` (sin duplicados) y reinicia el stack.

Bloqueo SMTP

Normal en Clouding. Usa APIs (Gmail, MailerLite/ActiveCampaign) o un relay externo.

Seguridad y buenas prácticas (rápidas)

Basic Auth delante del editor (opcional, recomendable):

```
N8N_BASIC_AUTH_ACTIVE=true
N8N_BASIC_AUTH_USER=admin
N8N_BASIC_AUTH_PASSWORD=pon_una_contra_fuerte
```

Reinicia: `docker compose restart`.

Backups: Activa Snapshots en Clouding regularmente. La carpeta `/opt/n8n/data` contiene tu configuración y credenciales cifradas.

Actualizaciones: Evita `:latest` en producción. Cuando todo esté estable, fija una versión de n8n (ej.: `n8nio/n8n:2.0.2`) y actualiza con `docker compose pull && up -d`.

Costes orientativos

Clouding (1 vCPU / 2GB / 15GB): **~6,5 €/mes**. Alternativas SaaS (Zapier/Make/n8n Cloud) facturan por tareas/operaciones; con uso medio/alto, tu servidor sale más barato y con más control.

Checklist final (para que nada se te escape)

- Alta en Clouding y correo a soporte (promo +5€)
- DNS A: n8n.tudominio.com → IP
- Firewall con 22/80/443 y asignado a la VM
- Docker + Compose instalados
- `/opt/n8n/.env` sin duplicados y con https
- Caddyfile creado
- `docker-compose.yml` con n8n + caddy y red
- `docker compose up -d` + certificado emitido
- Owner creado en `/setup`
- Webhook Published y probado con GET en https

¿Quieres que lo convierta a PDF con portada, índice y capturas clave (login SSH, Caddy OK, `/setup`, Webhook Published, etc.)? Puedo prepararte un guion breve para tu vídeo (voice-over + on-screen tips) sobre esta misma guía.