EBENEZER IT · GUÍAS PASO A PASO

GUÍA 06 · EPISODIOS 11 Y 12

Tu propia VPN con WireGuard, y el servidor cerrado detrás de ella

Una VPN WireGuard con panel web (wg-easy) en Docker, tus dispositivos conectados, y después lo más importante: SSH y los paneles de administración accesibles solo a través de la VPN, sin quedarte nunca fuera.

~2–3 horas
la primera vez, sin prisa

Requisito
Guías 01, 02 y 03 terminadas

Ubuntu 24.04 LTS
WireGuard · wg-easy · UFW

www.ebenezer.org.es · Alejandro Quevedo · Versión pública 1.0 · noviembre 2026

Índice

Antes de empezar

0 Qué vas a construir y valores de esta guía

Parte A · La VPN

Paso 01 Qué es WireGuard y cómo encaja

Paso 02 Abrir el puerto de la VPN

Paso 03 Subdominio y certificado del panel

Paso 04 Carpeta, versión y configuración inicial

Paso 05 El compose.yaml

Paso 06 Arrancar y comprobar

Paso 07 Publicar el panel con Nginx

Paso 08 Primer acceso y limpieza de la contraseña

Paso 09 Conectar tu PC y tu móvil

Paso 10 Comprobar que el tráfico va por la VPN

Parte B · El servidor, solo por la VPN

Paso 11 Desde dónde llegas cuando usas la VPN

Paso 12 El plan de rescate, antes de cerrar nada

Paso 13 SSH solo por la VPN

Paso 14 fail2ban: no castigar a la VPN

Paso 15 Cerrar el 22 en IONOS

Paso 16 Paneles solo por la VPN

Paso 17 Comprobación final e instantánea

Anexos

A Problemas frecuentes y cómo volver a entrar

B Puertos al terminar esta guía

C Chuleta de la VPN

Cómo usar esta guía

Tienes aquí **todos los pasos y comandos**, en orden. La explicación de cada paso (el porqué de cada línea) está en el **vídeo del episodio**.

Para copiar los comandos sin errores, usa la versión web: **https://www.ebenezer.org.es/guia-06-vpn.html**. Al copiar desde un PDF se pueden romper los saltos de línea y las comillas.

ANTES DE EMPEZAR

Qué vas a construir

Dos cosas, una detrás de otra. Primero, **tu propia VPN**: un túnel cifrado entre tus dispositivos y tu servidor, que se maneja desde un panel web en `https://vpn.TU_DOMINIO`. Después, usarla para **cerrar el servidor**: que SSH y los paneles de administración solo se puedan abrir desde dentro de la VPN. Para un robot de Internet, tu puerto SSH dejará de existir.

Pieza	Qué hace	Dónde vive
WireGuard	El protocolo de VPN: rápido, moderno y con muy poco código (menos fallos posibles)	Dentro del contenedor de wg-easy
wg-easy	Un panel web para crear y borrar dispositivos de la VPN sin escribir configuraciones a mano, con código QR para el móvil	Contenedor; panel en `127.0.0.1:51821` detrás de Nginx
Puerto 51820/UDP	Por donde entran los túneles. Es el único puerto nuevo que se abre a Internet	Público, a propósito (Guía 03, paso 7)
UFW e IONOS	En la Parte B dejan pasar SSH solo si viene de la VPN	El servidor y el panel de IONOS

El recorrido de una conexión

1. Activas la VPN en tu PC. El programa WireGuard cifra todo tu tráfico y lo envía al puerto 51820/UDP de tu servidor.
2. El contenedor de wg-easy lo recibe, lo descifra y lo envía a su destino: a Internet o al propio servidor.
3. Para el resto de Internet, tu tráfico sale con la IP del servidor. Y para el servidor, tú llegas desde dentro, desde la red de la VPN.
4. Esa diferencia es la que aprovecha la Parte B: «si vienes de la red de la VPN, pasas; si no, no».

La Parte B puede dejarte fuera de tu servidor

Cerrar SSH es el cambio más delicado del curso. Por eso el paso 12 prepara una forma de volver a entrar **antes** de cerrar nada, y cada cierre se prueba con dos ventanas abiertas. Sigue el orden de los pasos al pie de la letra.

Valores de esta guía

Valor	Qué es
`TU_DOMINIO`	Tu dominio, como en las guías anteriores
`TU_IP`	La IP del VPS
`TU_USUARIO`	Tu usuario del servidor

Valor	Qué es
`TU_VERSION_WGEASY`	La versión exacta de wg-easy (paso 4)
`TU_USUARIO_VPN`	El usuario administrador del panel de la VPN
`TU_CLAVE_VPN`	Su contraseña (paso 4)

Recordatorio

- Haz una instantánea en IONOS antes de empezar.
- Bloques **verdes** en el servidor (`ssh lab`), **azules** en PowerShell de tu PC, **morados** son contenido de archivos, **ocres** son clics en un panel web.
- `sudo nginx -t` siempre antes de recargar Nginx.

PARTE A

La VPN

Montamos wg-easy en Docker, configurado sin dejar nunca un asistente abierto a Internet, publicamos su panel con HTTPS y conectamos tus dispositivos. Corresponde al **episodio 11**.

Paso 1 Episodio 11

Qué es WireGuard y cómo encaja

Por qué: antes de abrir puertos, conviene tener claro qué es cada pieza y por qué la VPN no puede ir detrás de Nginx como el resto de aplicaciones.

Concepto	Qué es
Túnel	Una conexión cifrada entre un dispositivo tuyo (un *peer*, «par» o cliente) y el servidor. Todo lo que viaja dentro va cifrado aunque estés en la wifi de un bar
Claves	Cada dispositivo tiene su par de claves, como las de SSH (Guía 01). El servidor solo acepta dispositivos cuya clave pública conoce. No hay contraseña que adivinar
Red de la VPN	Cada dispositivo conectado recibe una IP privada propia: `10.8.0.2`, `10.8.0.3`… El servidor de la VPN es el `10.8.0.1`
UDP 51820	WireGuard usa UDP, no TCP, y no es HTTP. Nginx no puede ponerse delante: por eso este puerto se publica directamente, de forma consciente (la excepción que anunciaba la Guía 03)
Silencio	Si alguien escanea el 51820 sin una clave válida, WireGuard **no responde nada**. El puerto parece cerrado. Es una de sus mejores propiedades de seguridad

Puerto	Para qué	Abierto a
51820/UDP	Los túneles de la VPN	Internet (protegido por las claves)
51821/TCP	El panel web de wg-easy	Solo `127.0.0.1`; se llega por Nginx con HTTPS

Comprueba antes de seguir

Sabes explicar por qué el 51820 se publica a Internet y el 51821 no.

Paso 2 Episodio 11

Abrir el puerto de la VPN

Por qué: el 51820/UDP tiene que pasar las dos capas de cortafuegos de la Guía 01: la de IONOS y UFW.

2.1 · En IONOS

En el panel de IONOS: **Red** → **Políticas de firewall** → la política de tu servidor → **Añadir regla**: protocolo **UDP**, puerto **51820**, origen **cualquiera**, descripción `WireGuard`. Guarda.

2.2 · En UFW

EN EL SERVIDOR

```
sudo ufw allow 51820/udp comment 'WireGuard'
sudo ufw status numbered
```

Comprueba antes de seguir

En IONOS aparece la regla UDP 51820 y `ufw status` muestra `51820/udp ALLOW IN Anywhere`.

Paso 3 Episodio 11

Subdominio y certificado del panel

Por qué: el panel tendrá su subdominio con HTTPS, con el mismo procedimiento de las guías anteriores.

En el DNS de IONOS, un registro **A**: nombre de host `vpn`, apunta a `TU_IP`. Comprueba con `Resolve-DnsName vpn.TU_DOMINIO -Server 1.1.1.1` desde tu PC. Después, el sitio HTTP mínimo y el certificado:

EN EL SERVIDOR

```
sudo tee /etc/nginx/sites-available/vpn.TU_DOMINIO > /dev/null <<'EOF'
server {
    listen 80;
    listen [::]:80;
    server_name vpn.TU_DOMINIO;

    location ^~ /.well-known/acme-challenge/ {
        root /var/www/letsencrypt;
    }

    location / {
        return 301 https://$host$request_uri;
    }
}
EOF
sudo ln -s /etc/nginx/sites-available/vpn.TU_DOMINIO /etc/nginx/sites-enabled/
sudo nginx -t && sudo systemctl reload nginx
sudo certbot certonly --webroot -w /var/www/letsencrypt -d vpn.TU_DOMINIO --dry-run
sudo certbot certonly --webroot -w /var/www/letsencrypt -d vpn.TU_DOMINIO --cert-name
vpn.TU_DOMINIO
```

Comprueba antes de seguir

`sudo certbot certificates` muestra el certificado `vpn.TU_DOMINIO`.

Paso 4 Episodio 11

Carpeta, versión y configuración inicial

Por qué: wg-easy trae un asistente web para crear el administrador, pero lo completa **el primero que llegue**, como el de WordPress.

4.1 · La versión

wg-easy no se publica en Docker Hub sino en GitHub (`ghcr.io`). Busca la última versión en `github.com/wg-easy/wg-easy/releases`: tendrá el formato `15.x.y`. Apúntala como **TU_VERSION_WGEASY**. Esta guía está escrita para la serie 15; si la versión más reciente empieza por otro número, lee antes sus notas de cambios.

4.2 · La contraseña del panel

EN EL SERVIDOR

```
openssl rand -base64 24
```

Genera una contraseña larga. Guárdala en tu gestor como **TU_CLAVE_VPN**, junto con el usuario que elijas (**TU_USUARIO_VPN**, que no sea `admin`).

4.3 · La carpeta y el .env

EN EL SERVIDOR

```
mkdir -p /opt/apps/vpn/datos
umask 077
cat > /opt/apps/vpn/.env <<'EOF'
INIT_ENABLED=true
INIT_USERNAME=TU_USUARIO_VPN
INIT_PASSWORD=TU_CLAVE_VPN
INIT_HOST=vpn.TU_DOMINIO
INIT_PORT=51820
INIT_DNS=1.1.1.1,9.9.9.9
EOF
umask 022
ls -la /opt/apps/vpn
```

Las variables INIT_ solo se leen una vez

Se usan **solo en el primer arranque**, con la carpeta `datos` vacía. Después, wg-easy guarda todo en su base de datos (dentro de `datos`) y las ignora. Por eso, en el paso 8, borraremos la contraseña del `.env`: ya no hace falta y no tiene sentido dejarla escrita.

Comprueba antes de seguir

`ls -la /opt/apps/vpn` muestra `.env` con `-rw-------` y la carpeta `datos` vacía.

Paso 5 Episodio 11

El compose.yaml

Por qué: parte del ejemplo oficial de wg-easy, con tres cambios para nuestro esquema: el panel solo en `127.0.0.1`, sin IPv6 y con una red de Docker de dirección fija, que en la Parte B será la llave para entrar al servidor.

Créalo con `nano /opt/apps/vpn/compose.yaml`:

CONTENIDO DEL ARCHIVO · /OPT/APPS/VPN/COMPOSE.YAML

```
services:
  wg-easy:
    image: ghcr.io/wg-easy/wg-easy:TU_VERSION_WGEASY
    container_name: wg-easy
    restart: unless-stopped
    env_file: .env
    environment:
      - PORT=51821
      - HOST=0.0.0.0
      - INSECURE=false
      - DISABLE_IPV6=true
    volumes:
      - ./datos:/etc/wireguard
    ports:
      - "51820:51820/udp"
      - "127.0.0.1:51821:51821/tcp"
    cap_add:
      - NET_ADMIN
      - SYS_MODULE
    sysctls:
      - net.ipv4.ip_forward=1
      - net.ipv4.conf.all.src_valid_mark=1
    networks:
      vpn:
        ipv4_address: 172.30.0.2

networks:
  vpn:
    ipam:
      config:
        - subnet: 172.30.0.0/24
```

Comprueba antes de seguir

`cd /opt/apps/vpn && docker compose config --quiet && echo correcto` responde `correcto`.

Paso 6 Episodio 11

Arrancar y comprobar

Por qué: arrancamos, miramos el registro para confirmar que la configuración inicial se ha aplicado y comprobamos que WireGuard escucha.

EN EL SERVIDOR

```
cd /opt/apps/vpn
docker compose pull
docker compose up -d
docker compose ps
docker compose logs --tail 30
```

En el registro deben aparecer líneas indicando que se ha creado el usuario y que la interfaz `wg0` está levantada. Después:

EN EL SERVIDOR

```
sudo ss -ulpn | grep 51820
sudo ss -tlpn | grep 51821
docker exec wg-easy wg show
```

SALIDA ESPERADA

```
UNCONN 0 0   0.0.0.0:51820   0.0.0.0:*   users:(("docker-proxy",...))
LISTEN 0 4096 127.0.0.1:51821 0.0.0.0:*  users:(("docker-proxy",...))
interface: wg0
  public key: AbC...=
  private key: (hidden)
  listening port: 51820
```

Comprueba antes de seguir

- `docker compose ps` muestra `wg-easy` en `Up`.
- 51820/UDP en `0.0.0.0` y 51821/TCP en `127.0.0.1`.
- `wg show` muestra la interfaz `wg0`.

Paso 7 Episodio 11

Publicar el panel con Nginx

Por qué: el sitio completo, con el mismo patrón de las guías anteriores.

EN EL SERVIDOR

```
sudo tee /etc/nginx/sites-available/vpn.TU_DOMINIO > /dev/null <<'EOF'
# HTTP: Let's Encrypt y redirección a HTTPS
server {
    listen 80;
    listen [::]:80;
    server_name vpn.TU_DOMINIO;

    location ^~ /.well-known/acme-challenge/ {
        root /var/www/letsencrypt;
    }

    location / {
        return 301 https://$host$request_uri;
    }
}

# HTTPS: panel de wg-easy (contenedor en 127.0.0.1:51821)
server {
    listen 443 ssl http2;
    listen [::]:443 ssl http2;
    server_name vpn.TU_DOMINIO;

    ssl_certificate     /etc/letsencrypt/live/vpn.TU_DOMINIO/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/vpn.TU_DOMINIO/privkey.pem;
    include snippets/tls.conf;

    add_header Strict-Transport-Security "max-age=86400" always;
    add_header X-Content-Type-Options "nosniff" always;
    add_header Referrer-Policy "strict-origin-when-cross-origin" always;
    add_header X-Frame-Options "DENY" always;

    location / {
        proxy_pass http://127.0.0.1:51821;
        include snippets/proxy.conf;
    }

    access_log /var/log/nginx/vpn.access.log;
    error_log  /var/log/nginx/vpn.error.log;
}
EOF
sudo nginx -t && sudo systemctl reload nginx
```

Comprueba antes de seguir

Desde tu PC, `curl.exe -I https://vpn.TU_DOMINIO` responde `200` (o una redirección a la página de entrada).

Paso 8 Episodio 11

Primer acceso y limpieza de la contraseña

Por qué: entramos al panel con el usuario de la configuración desatendida y, ya comprobado, borramos la contraseña del `.env`.

1. Abre `https://vpn.TU_DOMINIO`. Debe salir la página de inicio de sesión, **no** el asistente de instalación. Si sale el asistente, revisa el Anexo A antes de seguir.
2. Entra con `TU_USUARIO_VPN` y `TU_CLAVE_VPN`.
3. En los ajustes del panel (**Admin Panel** o el icono del engranaje, según la versión), comprueba que el host es `vpn.TU_DOMINIO` y el puerto `51820`, y pon el idioma en español si está disponible.

Ahora, en el servidor, quita la contraseña del `.env` y desactiva la configuración inicial:

```
EN EL SERVIDOR
cd /opt/apps/vpn
sed -i '/^INIT_PASSWORD=/d; s/^INIT_ENABLED=true/INIT_ENABLED=false/' .env
cat .env
docker compose up -d
```

Comprueba antes de seguir

- `cat .env` ya no muestra la contraseña.
- Después del `up -d`, sigues pudiendo entrar al panel.

Paso 9 Episodio 11

Conectar tu PC y tu móvil

Por qué: cada dispositivo es un cliente con sus propias claves.

9.1 · Crear los clientes en el panel

1. En el panel, **+ Nuevo cliente** (*New Client*).
2. Nombre: uno claro por dispositivo, por ejemplo `pc-casa` y `movil`. Nunca compartas un cliente entre dos dispositivos.
3. Créalos. Cada uno aparece en la lista con su IP (`10.8.0.2`, `10.8.0.3`...).

9.2 · El PC con Windows

1. Descarga WireGuard para Windows desde la web oficial, `wireguard.com/install`, e instálalo.
2. En el panel, en el cliente `pc-casa`, pulsa el botón de **descarga**: baja un archivo `.conf`.
3. En WireGuard: **Importar túnel(es) desde archivo** → elige el `.conf` → **Activar**.
4. Borra el `.conf` de la carpeta de descargas: contiene la clave privada de ese dispositivo.

9.3 · El móvil

1. Instala la app oficial **WireGuard** (de WireGuard Development Team) desde Play Store o App Store.
2. En la app, **+** → **Escanear desde código QR**.
3. En el panel, en el cliente `movil`, pulsa el icono del **código QR** y escanéalo. Ponle un nombre y actívalo.

El archivo y el QR son la llave

Quien tenga ese `.conf` o haga una foto del QR, entra en tu VPN. No los mandes por correo ni por chat, y no los enseñes en un vídeo. Si crees que se ha filtrado uno, bórralo en el panel y crea otro.

Comprueba antes de seguir

En el panel, los dos clientes muestran un punto verde o la hora de su último contacto (*handshake*) al tener la VPN activada.

Paso 10 Episodio 11

Comprobar que el tráfico va por la VPN

Por qué: conectar no es lo mismo que funcionar.

Con la VPN **activada** en el PC:

EN TU PC · POWERSHELL

```
curl.exe -4 ifconfig.me
ping 10.8.0.1
```

En el servidor verás el túnel con datos:

EN EL SERVIDOR

```
docker exec wg-easy wg show
```

SALIDA ESPERADA

```
peer: XyZ...=
  endpoint: IP_DE_TU_CASA:54321
  allowed ips: 10.8.0.2/32
  latest handshake: 12 seconds ago
  transfer: 1.20 MiB received, 4.85 MiB sent
```

Desactiva la VPN y repite `curl.exe -4 ifconfig.me`: ahora debe salir la IP de tu casa. Prueba también en el móvil con los datos móviles, abriendo `ifconfig.me` en el navegador.

Comprueba antes de seguir

- Con VPN: `ifconfig.me` da TU_IP y `10.8.0.1` responde.
- Sin VPN: vuelve a salir la IP de tu casa.
- `wg show` muestra *latest handshake* reciente.

PARTE B

El servidor, solo por la VPN

Ahora usamos la VPN para cerrar el servidor: SSH y los paneles de administración solo aceptarán conexiones que vengan de dentro. Antes de cerrar nada, preparamos cómo volver a entrar si algo sale mal. Corresponde al **episodio 12**.

Paso 11 Episodio 12

Desde dónde llegas cuando usas la VPN

Por qué: para decir «solo desde la VPN» hay que saber con qué IP llegas al servidor cuando la usas.

Con la VPN **activada**, entra por SSH como siempre y pregunta al servidor desde dónde le hablas:

EN TU PC · POWERSHELL

```
ssh lab
echo $SSH_CLIENT
```

SALIDA ESPERADA

```
172.30.0.2 51234 22
```

Por qué

1. Tu PC envía la conexión SSH por el túnel. Llega al contenedor de wg-easy.
2. El contenedor la reenvía hacia el servidor, pero cambiando la dirección de origen por la suya propia (`172.30.0.2`). Eso se llama **NAT** o enmascaramiento, y wg-easy lo hace siempre para que las respuestas vuelvan por el mismo camino.
3. El servidor ve llegar la conexión desde la red `172.30.0.0/24`, por dentro, sin pasar por el firewall de IONOS.

Desactiva la VPN, vuelve a entrar y repite `echo $SSH_CLIENT`: ahora verás la IP de tu casa. Esa diferencia es todo lo que necesitamos: **la red `172.30.0.0/24` significa «viene de la VPN»**.

Comprueba antes de seguir

Con VPN, `$SSH_CLIENT` empieza por `172.30.0.2`; sin VPN, por la IP de tu casa.

Paso 12 Episodio 12

El plan de rescate, antes de cerrar nada

Por qué: a partir del paso 13, si la VPN deja de funcionar (el contenedor no arranca, una actualización, un error tuyo), no podrás entrar por SSH.

12.1 · Instantánea

En IONOS, crea una instantánea: `06-antes-de-cerrar-ssh`. Si todo falla, vuelves a este punto exacto.

12.2 · La consola remota de IONOS

IONOS te deja usar el servidor como si tuvieras una pantalla y un teclado conectados, desde el navegador, sin pasar por la red. No depende ni de SSH ni de los cortafuegos. Pruébala ahora:

1. En el panel de IONOS, entra en tu servidor y busca **Acciones → Consola remota** (o «KVM»).
2. Se abre una pantalla negra con `login:`. Escribe tu usuario (TU_USUARIO) y **la contraseña de tu usuario**, la que pusiste en la Guía 01. Aquí sí se entra con contraseña: la regla de «solo claves» es solo para SSH.
3. Si entras, escribe `exit` y cierra la consola.

> **Si la consola remota no te deja entrar, no sigas**
> Sin una puerta de atrás probada, el paso 13 es demasiado arriesgado. Lo más habitual es no recordar la contraseña del usuario: cámbiala ahora, por SSH, con `passwd`, guárdala en tu gestor y repite la prueba. El teclado de la consola suele estar en inglés: si la contraseña tiene símbolos, pueden salir en otra tecla.

12.3 · Cómo se vuelve a abrir SSH desde la consola

Apúntalo junto a la contraseña del usuario. Si un día te quedas fuera, desde la consola remota:

```
EN EL SERVIDOR
sudo ufw allow OpenSSH comment 'SSH temporal'
sudo ufw status
```

Y en IONOS, vuelve a abrir el 22 en la política de firewall. Cuando lo arregles, deshaces los dos cambios.

> **Comprueba antes de seguir**
> - Tienes la instantánea `06-antes-de-cerrar-ssh`.
> - Has entrado al menos una vez por la consola remota con tu usuario y contraseña.

Paso 13 Episodio 12

SSH solo por la VPN

Por qué: añadimos la regla nueva (SSH solo desde la red de la VPN), comprobamos que funciona con una ventana abierta de reserva, y **solo entonces** quitamos la regla antigua (SSH desde cualquier sitio).

13.1 · Añadir la regla nueva

Con la VPN activada y una sesión SSH abierta (esta será tu ventana de reserva: **no la cierres hasta el final del paso**):

```
EN EL SERVIDOR
sudo ufw allow from 172.30.0.0/24 to any port 22 proto tcp comment 'SSH por la VPN'
sudo ufw status numbered
```

SALIDA ESPERADA

```
Status: active
     To                         Action      From
     --                         ------      ----
[ 1] OpenSSH                    ALLOW IN    Anywhere                   # SSH
[ 2] 80/tcp                     ALLOW IN    Anywhere                   # HTTP
[ 3] 443/tcp                    ALLOW IN    Anywhere                   # HTTPS
[ 4] 51820/udp                  ALLOW IN    Anywhere                   # WireGuard
[ 5] 22/tcp                     ALLOW IN    172.30.0.0/24              # SSH por la VPN
[ 6] OpenSSH (v6)               ALLOW IN    Anywhere (v6)              # SSH
...
```

13.2 · Probarla con la VPN

Abre una **segunda** ventana de PowerShell, con la VPN activada, y entra:

EN TU PC · POWERSHELL

```
ssh lab
echo $SSH_CLIENT
```

Debe entrar y mostrar `172.30.0.2`. Esta conexión ya entra por la regla nueva. Sal con `exit`.

13.3 · Quitar la regla antigua

En la ventana de reserva, borra las reglas `OpenSSH` (la de IPv4 y la de IPv6). **Usa los números de tu salida**, empezando por el más alto: al borrar una regla, las siguientes cambian de número.

EN EL SERVIDOR

```
sudo ufw delete 6
sudo ufw delete 1
sudo ufw status numbered
```

13.4 · La prueba de verdad

Prueba (en una ventana nueva)	Resultado correcto
Con la VPN activada: `ssh lab`	Entra
Con la VPN **desactivada**: `ssh lab`	Se queda esperando y falla con `Connection timed out`

Solo cuando las dos pruebas salgan bien, cierra la ventana de reserva.

> **Las sesiones abiertas no se cortan**
> Quitar una regla de UFW no corta las conexiones que ya están abiertas: por eso la ventana de reserva sigue funcionando, y por eso hay que probar siempre con una ventana **nueva**.

> **Comprueba antes de seguir**
> Por SSH solo se entra con la VPN activada, y la ventana de reserva sigue abierta hasta comprobarlo.

Paso 14 Episodio 12

fail2ban: no castigar a la VPN

Por qué: ahora todas tus conexiones SSH llegan desde la misma IP, la del contenedor (`172.30.0.2`).

EN EL SERVIDOR

```
sudo sed -i 's|^ignoreip = .*|& 172.30.0.0/24|' /etc/fail2ban/jail.local
grep ignoreip /etc/fail2ban/jail.local
sudo systemctl restart fail2ban
sudo fail2ban-client get sshd ignoreip
```

¿No es esto un riesgo?

Para llegar a `172.30.0.0/24` hay que estar dentro de tu VPN, es decir, tener una de las claves de tus dispositivos. Y aun así, SSH sigue pidiendo tu clave SSH. Es un riesgo mucho menor que dejar la VPN entera a merced de un bloqueo por error.

Comprueba antes de seguir

`fail2ban-client get sshd ignoreip` incluye `172.30.0.0/24` .

Paso 15 Episodio 12

Cerrar el 22 en IONOS

Por qué: la segunda capa.

1. En IONOS: **Red** → **Políticas de firewall** → tu política.
2. Borra la regla del puerto **22** (TCP). Deja 80, 443 y 51820/UDP.
3. Guarda.

Comprueba desde tu PC, con la VPN **desactivada**:

EN TU PC · POWERSHELL

```
Test-NetConnection TU_IP -Port 22
```

SALIDA ESPERADA

```
WARNING: TCP connect to TU_IP:22 failed
TcpTestSucceeded : False
```

Y con la VPN **activada**, `ssh lab` sigue entrando.

Comprueba antes de seguir

- Sin VPN, el 22 no responde ni siquiera a un escaneo.
- Con VPN, entras con normalidad.
- Tienes apuntado cómo volver a abrirlo (paso 12.3).

Paso 16 Episodio 12

Paneles solo por la VPN

Por qué: lo mismo para los paneles web de administración, empezando por el de la propia VPN.

16.1 · El snippet

EN EL SERVIDOR

```
sudo tee /etc/nginx/snippets/solo-vpn.conf > /dev/null <<'EOF'
# Ebenezer: solo se puede acceder desde la VPN
allow 172.30.0.0/24;
deny all;
EOF
```

16.2 · Aplicarlo al panel de la VPN

EN EL SERVIDOR

```
sudo sed -i 's|^    include snippets/tls.conf;|&\n    include snippets/solo-vpn.conf;|'
/etc/nginx/sites-available/vpn.TU_DOMINIO
grep -n "include" /etc/nginx/sites-available/vpn.TU_DOMINIO
sudo nginx -t && sudo systemctl reload nginx
```

Prueba	Resultado correcto
Con la VPN activada: `https://vpn.TU_DOMINIO`	Abre el panel
Con la VPN desactivada (o desde el móvil con datos, sin VPN)	**403 Forbidden**

¿Y si se cae la VPN? ¿Cómo llego al panel?
Puede pasar que wg-easy funcione pero no tengas ningún dispositivo con la VPN configurada (por ejemplo, si pierdes el móvil y el PC a la vez). Para crear uno nuevo necesitas el panel, y el panel solo se abre desde la VPN. La salida: entra por la consola remota de IONOS (paso 12), pon un `#` delante de la línea `include snippets/solo-vpn.conf;` del sitio, recarga Nginx, entra al panel desde tu PC, crea el cliente nuevo, y vuelve a quitar el `#`. Apúntalo con el resto del plan de rescate.

16.3 · Otros paneles

A partir de ahora, cada panel de administración que montes (Uptime Kuma, Grafana, AdGuard...) llevará esta misma línea en su sitio. Para paneles dentro de una web pública, se pone solo en la ruta del panel. Por ejemplo, para que el acceso de WordPress (Guía 05) solo funcione con la VPN, en `location = /wp-login.php` y en un `location ^~ /wp-admin/` nuevo:

CONTENIDO DEL ARCHIVO · FRAGMENTO DE /ETC/NGINX/SITES-AVAILABLE/BLOG.TU_DOMINIO

```
    location = /wp-login.php {
        include snippets/solo-vpn.conf;
        limit_req zone=wplogin burst=5 nodelay;
        proxy_pass http://127.0.0.1:8082;
        include snippets/proxy.conf;
    }
    location ^~ /wp-admin/ {
        include snippets/solo-vpn.conf;
        proxy_pass http://127.0.0.1:8082;
        include snippets/proxy.conf;
    }
```

Excepción de WordPress: admin-ajax.php

Algunos temas y plugins usan `/wp-admin/admin-ajax.php` desde la parte pública de la web (formularios, carritos). Si al cerrar `/wp-admin/` algo de la web pública deja de funcionar, añade antes un `location = /wp-admin/admin-ajax.php` sin el snippet. Si tu web es un blog sencillo, no lo necesitarás.

Comprueba antes de seguir

El panel de la VPN da 403 sin VPN y se abre con ella.

Paso 17 Episodio 12

Comprobación final e instantánea

Por qué: repaso completo con y sin VPN, reinicio para ver que todo vuelve solo, y la instantánea.

EN EL SERVIDOR

```
sudo ufw status verbose
sudo fail2ban-client get sshd ignoreip
docker compose -f /opt/apps/vpn/compose.yaml ps
docker exec wg-easy wg show | grep -E "peer|handshake"
sudo nginx -t
```

Comprobación	Resultado correcto
UFW	`22/tcp` **solo** desde `172.30.0.0/24`; 80, 443 y 51820/udp desde cualquier sitio; nada de `OpenSSH` general
IONOS	Abiertos 80, 443 y 51820/UDP. El 22, cerrado
fail2ban	`ignoreip` incluye `172.30.0.0/24`
wg-easy	`Up`; tus dispositivos con *handshake* reciente
Sin VPN	`ssh lab` falla por tiempo; `https://vpn.TU_DOMINIO` da 403; tu web y tu blog cargan con normalidad
Con VPN	`ssh lab` entra; el panel se abre; `ifconfig.me` muestra `TU_IP`

Reinicia el servidor (con la VPN activada, entra por SSH y escribe `sudo reboot`). Al minuto, la VPN debe reconectar sola y `ssh lab` debe volver a entrar. Después, en IONOS, crea la instantánea `06-vpn`.

Copia de la VPN

Añade `/opt/apps/vpn` (el `.env`, el `compose.yaml` y la carpeta `datos`) a tus copias de seguridad. Con eso, si rehaces el servidor, tus dispositivos vuelven a conectar sin reconfigurar nada.

Comprueba antes de seguir

Tu servidor ya no tiene SSH abierto a Internet, y sus paneles solo se ven desde tu VPN. En la siguiente guía instalamos AdGuard Home y lo usamos como DNS de la VPN: todos tus dispositivos conectados navegarán sin anuncios ni rastreadores.

ANEXOS

Para cuando algo no sale

Si no puedes entrar al servidor, empieza por el final de esta página: «Me he quedado fuera».

Anexo A · Problemas frecuentes y cómo volver a entrar

La VPN

Lo que ves	Qué pasa	Qué hacer
Al abrir el panel sale el asistente de instalación, no el inicio de sesión	No se aplicó la configuración desatendida: falta alguna variable `INIT_` o la carpeta `datos` no estaba vacía	**No lo rellenes desde fuera.** `docker compose down`, revisa el `.env` (las cuatro: USERNAME, PASSWORD, HOST y PORT, más `INIT_ENABLED=true`), vacía `datos` (`sudo rm -rf datos/*`, solo en una instalación nueva) y `docker compose up -d`
No puedo iniciar sesión: se recarga la página o dice que no es seguro	wg-easy no detecta el HTTPS	Comprueba que el sitio incluye `snippets/proxy.conf` (`X-Forwarded-Proto`). Como último recurso, `INSECURE=true` es aceptable **solo** porque el panel está en 127.0.0.1 y Nginx pone el HTTPS
El cliente dice «activo» pero no hay *handshake*	Los paquetes no llegan al 51820/UDP	Regla UDP 51820 en IONOS (paso 2); `sudo ss -ulpn | grep 51820`; el host del cliente debe ser `vpn.TU_DOMINIO` y resolver a TU_IP
Hay *handshake* pero no navega	Falta el reenvío de paquetes o el DNS no responde	`sysctls` del compose (paso 5); prueba `ping 1.1.1.1` con la VPN: si responde, el problema es el DNS (`INIT_DNS` o los ajustes del panel)
`Pool overlaps with other one on this address space` al arrancar	La red `172.30.0.0/24` ya existe en tu servidor	Elige otro rango (paso 5) y úsalo en **toda** la guía: UFW, fail2ban y el snippet
El móvil conecta en wifi pero no con datos (o al revés)	Algunas redes bloquean UDP o puertos poco comunes	Prueba en otra red. Es raro, pero existen redes de empresas u hoteles que solo dejan salir web

SSH y paneles

Lo que ves	Qué pasa	Qué hacer
Con la VPN activada, `ssh lab` da *timed out*	La regla de UFW no coincide con la IP desde la que llegas	Desde la consola remota: `sudo ufw status numbered` y compara con lo que dio `$SSH_CLIENT` en el paso 11
El panel da 403 incluso con la VPN	El navegador no está usando la VPN, o el rango del snippet no coincide	Comprueba `ifconfig.me` con la VPN; revisa `/etc/nginx/snippets/solo-vpn.conf` y `vpn.access.log` (verás la IP desde la que llegas)
fail2ban ha bloqueado la VPN	Falta el `ignoreip` del paso 14	Consola remota: `sudo fail2ban-client set sshd unbanip 172.30.0.2`, y haz el paso 14

Me he quedado fuera

1. Entra por la **consola remota de IONOS** con tu usuario y contraseña (paso 12).
2. Mira si la VPN funciona: `docker compose -f /opt/apps/vpn/compose.yaml ps` y `docker compose -f /opt/apps/vpn/compose.yaml logs --tail 30`.
3. Si necesitas entrar por SSH mientras lo arreglas: `sudo ufw allow OpenSSH comment 'SSH temporal'` y abre el 22 en IONOS.
4. Cuando todo funcione, quita la regla temporal (`sudo ufw status numbered` y `sudo ufw delete N`) y vuelve a cerrar el 22 en IONOS.
5. Si nada de esto funciona, restaura la instantánea `06-antes-de-cerrar-ssh`.

Anexo B · Puertos al terminar esta guía

Puerto	Servicio	Firewall de IONOS	UFW	Qué lo protege
22/TCP	SSH	**Cerrado**	Solo desde `172.30.0.0/24`	VPN + clave SSH + fail2ban
80/TCP	HTTP (Nginx)	Abierto	Abierto	Solo Let's Encrypt y redirección
443/TCP	HTTPS (Nginx)	Abierto	Abierto	TLS; paneles con `solo-vpn.conf`
51820/UDP	WireGuard	Abierto	Abierto	Claves de cada dispositivo; no responde sin clave válida
127.0.0.1:51821	Panel de wg-easy	—	—	Solo el propio servidor; por Nginx, solo desde la VPN

Redes de esta guía

Red	Qué es
`10.8.0.0/24`	La red de la VPN: el servidor es `10.8.0.1` y tus dispositivos, `10.8.0.2` en adelante
`172.30.0.0/24`	La red de Docker del contenedor de wg-easy (`172.30.0.2`). Es la IP con la que llegas al servidor cuando usas la VPN: la que autorizan UFW, fail2ban y Nginx

Anexo C · Chuleta de la VPN

Para...	Comando
Estado del contenedor	`docker compose -f /opt/apps/vpn/compose.yaml ps`
Túneles activos y último contacto	`docker exec wg-easy wg show`
Registros	`docker compose -f /opt/apps/vpn/compose.yaml logs --tail 50`
¿Desde dónde me conecto?	`echo $SSH_CLIENT` (en el servidor) · `curl.exe -4 ifconfig.me` (en el PC)
Reglas del cortafuegos con número	`sudo ufw status numbered`
Abrir SSH temporalmente (rescate)	`sudo ufw allow OpenSSH comment 'SSH temporal'`
Probar un puerto desde Windows	`Test-NetConnection TU_IP -Port 22`
Añadir o quitar un dispositivo	En el panel: **+ Nuevo cliente** o la papelera del cliente
Actualizar wg-easy	Lee las notas de la versión, haz copia de `/opt/apps/vpn` , cambia la versión en el compose y `docker compose pull && docker compose up -d` . Hazlo **con la consola remota a mano**

Si te atascas

Pega: la salida de `sudo ufw status numbered`, `docker exec wg-easy wg show` (la clave privada sale oculta), las últimas líneas del registro de wg-easy y lo que te devuelve `echo $SSH_CLIENT`. Nunca pegues un `.conf` de cliente ni una captura del QR.