EBENEZER IT · GUÍAS PASO A PASO

GUÍA 07 · EPISODIO 13

Adiós a los anuncios: AdGuard Home como DNS de tu VPN

Un filtro de DNS en tu propio servidor que bloquea anuncios, rastreadores y webs maliciosas en todos los dispositivos conectados a tu VPN, sin instalar nada en ellos. Con DNS cifrado hacia Internet y su panel accesible solo desde la VPN.

~1–2 horas
la primera vez, sin prisa

Requisito
Guía 06 terminada

Ubuntu 24.04 LTS
AdGuard Home · wg-easy

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 · Preparación

Paso 01 Qué hace AdGuard Home y dónde encaja

Paso 02 Subdominio, certificado y sitio solo por la VPN

Parte B · Instalar

Paso 03 Carpeta y versión

Paso 04 El compose.yaml

Paso 05 Arrancar y comprobar

Paso 06 El asistente de configuración

Parte C · Configurar

Paso 07 DNS cifrado hacia Internet

Paso 08 Listas de bloqueo

Paso 09 Ajustes recomendados

Parte D · Usarlo en la VPN

Paso 10 Cambiar el DNS de la VPN

Paso 11 Actualizar tus dispositivos

Paso 12 Comprobar el bloqueo

Paso 13 Cuando se bloquea algo que necesitas

Paso 14 Comprobación final e instantánea

Anexos

A Problemas frecuentes

B Puertos y redes al terminar esta guía

C Chuleta de AdGuard Home

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-07-adguard.html**. Al copiar desde un PDF se pueden romper los saltos de línea y las comillas.

ANTES DE EMPEZAR

Qué vas a construir

Cada vez que abres una web o una app, tu dispositivo pregunta a un servidor **DNS** qué IP tiene cada dirección: la de la web, pero también la de los anuncios, los rastreadores y las estadísticas que lleva incrustados. **AdGuard Home** es un servidor DNS propio que, cuando le preguntan por un dominio de anuncios o malicioso, responde «no existe». El anuncio no llega a cargarse.

Lo pondremos en la red de la VPN, así que todos tus dispositivos conectados quedarán protegidos sin instalar nada en ellos: el móvil, el PC y, más adelante, los de tu familia o tus clientes.

Pieza	Qué hace	Dónde vive
AdGuard Home	Recibe las preguntas DNS de tus dispositivos, bloquea las de la lista negra y reenvía el resto a Internet, cifradas	Contenedor, en la red de la VPN con IP fija `172.30.0.3`
wg-easy	Le dice a cada dispositivo que use `172.30.0.3` como DNS mientras esté conectado	El de la Guía 06
Panel de AdGuard	Estadísticas, registro de consultas, listas de bloqueo y excepciones	`https://adguard.TU_DOMINIO`, solo desde la VPN

El recorrido de una consulta

1. Con la VPN activada, abres una web. El móvil pregunta: «¿qué IP tiene `anuncios.ejemplo.com`?». La pregunta viaja por el túnel.
2. wg-easy la entrega a AdGuard (`172.30.0.3`), por la red interna de Docker.
3. AdGuard mira sus listas. Si es un dominio de anuncios, responde `0.0.0.0` (no existe) y el anuncio no se carga. Si no, pregunta a Internet por un canal cifrado y devuelve la respuesta.
4. Todo queda en el registro de consultas y en las estadísticas del panel.

AdGuard no es accesible desde Internet

AdGuard **no publica el puerto 53** (el de DNS) en el servidor. Solo se le puede preguntar desde la red interna de la VPN. Así no se convierte en un «DNS abierto», que es lo que los atacantes buscan para usar en sus ataques, y no choca con el DNS del propio Ubuntu.

Valores de esta guía

Valor	Qué es
`TU_DOMINIO`	Tu dominio
`TU_IP`	La IP del VPS
`TU_VERSION_ADGUARD`	La versión exacta de AdGuard Home (paso 3)

Valor	Qué es
`TU_USUARIO_ADGUARD`	El usuario del panel de AdGuard (paso 6)

Recordatorio

- Haz una instantánea en IONOS antes de empezar.
- Esta guía da por hecha la Guía 06: wg-easy funcionando en la red `172.30.0.0/24` y el snippet `solo-vpn.conf` creado. Si usaste otro rango, cámbialo también aquí.
- Bloques **verdes** en el servidor, **azules** en PowerShell de tu PC, **morados** son contenido de archivos, **ocres** son clics en un panel web.

PARTE A

Preparación

Entendemos dónde se coloca AdGuard y preparamos su panel con HTTPS y cerrado a la VPN desde el primer momento. Corresponde al **episodio 13**.

Paso 1 Episodio 13

Qué hace AdGuard Home y dónde encaja

Por qué: AdGuard filtra por **nombre de dominio**, no por contenido.

Bloquea bien	No puede bloquear
Anuncios y rastreadores que vienen de dominios propios (`doubleclick.net` , `ads.ejemplo.com` ...)	Anuncios servidos desde el mismo dominio que el contenido, como los de YouTube: el vídeo y el anuncio vienen del mismo sitio
Estadísticas y telemetría de apps y del sistema	Lo que va dentro de una conexión ya abierta
Dominios de malware, estafas y *phishing* conocidos	Lo que no esté en sus listas (por eso se actualizan solas)

¿Por qué en la red de la VPN?

En la Guía 06 creamos para wg-easy una red de Docker con rango fijo (`172.30.0.0/24`). Si AdGuard entra en esa misma red con una IP fija (`172.30.0.3`), wg-easy puede entregarle las consultas de tus dispositivos directamente, sin publicar ningún puerto DNS en el servidor.

Dirección	Qué es
`172.30.0.2`	wg-easy (Guía 06)
`172.30.0.3`	AdGuard Home (esta guía)
`10.8.0.x`	Tus dispositivos dentro del túnel

Comprueba antes de seguir

Sabes explicar por qué AdGuard no bloquea los anuncios de YouTube y por qué no hace falta abrir el puerto 53.

Paso 2 Episodio 13

Subdominio, certificado y sitio solo por la VPN

Por qué: el asistente de instalación de AdGuard, como el de WordPress, lo completa el primero que llegue.

2.1 · DNS y certificado

En el DNS de IONOS, un registro **A**: `adguard` → `TU_IP`. El certificado se pide por el puerto 80, que sigue abierto a todo el mundo; solo el panel (el 443) quedará cerrado.

EN EL SERVIDOR

```
sudo tee /etc/nginx/sites-available/adguard.TU_DOMINIO > /dev/null <<'EOF'
server {
    listen 80;
    listen [::]:80;
    server_name adguard.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/adguard.TU_DOMINIO /etc/nginx/sites-enabled/
sudo nginx -t && sudo systemctl reload nginx
sudo certbot certonly --webroot -w /var/www/letsencrypt -d adguard.TU_DOMINIO --dry-run
sudo certbot certonly --webroot -w /var/www/letsencrypt -d adguard.TU_DOMINIO --cert-name
adguard.TU_DOMINIO
```

2.2 · El sitio completo

Ya con el certificado, el sitio definitivo. Todavía no recargues Nginx: el contenedor aún no existe (lo harás en el paso 5).

EN EL SERVIDOR

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

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

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

# HTTPS: panel de AdGuard Home (contenedor en 127.0.0.1:8083), solo desde la VPN
server {
    listen 443 ssl http2;
    listen [::]:443 ssl http2;
    server_name adguard.TU_DOMINIO;

    ssl_certificate     /etc/letsencrypt/live/adguard.TU_DOMINIO/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/adguard.TU_DOMINIO/privkey.pem;
    include snippets/tls.conf;
    include snippets/solo-vpn.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://127.0.0.1:8083;
        include snippets/proxy.conf;
    }

    access_log /var/log/nginx/adguard.access.log;
    error_log  /var/log/nginx/adguard.error.log;
}
EOF
```

Comprueba antes de seguir

`sudo certbot certificates` muestra `adguard.TU_DOMINIO` y el archivo del sitio tiene la línea `include snippets/solo-vpn.conf;`.

PARTE B

Instalar

AdGuard en su propio compose, unido a la red de la VPN con una IP fija, y su asistente de configuración completado desde dentro de la VPN.

Paso 3 Episodio 13

Carpeta y versión

Por qué: como siempre, una carpeta por aplicación y una versión fija.

EN EL SERVIDOR

```
mkdir -p /opt/apps/adguard/datos/work /opt/apps/adguard/datos/conf
cd /opt/apps/adguard
docker network ls | grep vpn
```

SALIDA ESPERADA

```
a1b2c3d4e5f6   vpn_vpn   bridge   local
```

Para la versión, en `hub.docker.com/r/adguard/adguardhome` → **Tags**: busca la más reciente con formato `v0.107.x` (con la `v` delante, sin `beta` ni `edge`). Apúntala como TU_VERSION_ADGUARD.

Comprueba antes de seguir

Existe la red `vpn_vpn` y tienes apuntada la versión.

Paso 4 Episodio 13

El compose.yaml

Por qué: corto, pero con una pieza nueva: en vez de crear su propia red, AdGuard se une a una red que ya existe, la de wg-easy.

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

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

```
services:
  adguard:
    image: adguard/adguardhome:TU_VERSION_ADGUARD
    container_name: adguard
    restart: unless-stopped
    volumes:
      - ./datos/work:/opt/adguardhome/work
      - ./datos/conf:/opt/adguardhome/conf
    ports:
      - "127.0.0.1:8083:3000"
    networks:
      vpn:
        ipv4_address: 172.30.0.3

networks:
  vpn:
    name: vpn_vpn
    external: true
```

El orden de arranque importa

Como la red pertenece al proyecto de wg-easy, tiene que existir antes de arrancar AdGuard. Tras un reinicio del servidor, Docker levanta los dos solos y en orden (la red ya existe). Pero si un día haces `docker compose down` en `/opt/apps/vpn`, la red se borra y AdGuard se queda sin ella: vuelve a levantar primero la VPN y después AdGuard.

Comprueba antes de seguir

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

Paso 5 Episodio 13

Arrancar y comprobar

Por qué: arrancamos, activamos el sitio de Nginx y comprobamos que el panel solo se abre desde la VPN.

EN EL SERVIDOR

```
cd /opt/apps/adguard
docker compose pull
docker compose up -d
docker compose ps
docker network inspect vpn_vpn --format '{{range .Containers}}{{.Name}} {{.IPv4Address}}
{{"\n"}}{{end}}'
sudo nginx -t && sudo systemctl reload nginx
```

SALIDA ESPERADA

```
NAME      IMAGE                                    STATUS        PORTS
adguard   adguard/adguardhome:TU_VERSION_ADGUARD   Up 5 seconds  53/tcp, 53/udp, ...,
127.0.0.1:8083→3000/tcp

wg-easy 172.30.0.2/24
adguard 172.30.0.3/24
```

Prueba	Resultado correcto
Con la VPN activada: `https://adguard.TU_DOMINIO`	El asistente de AdGuard («Bienvenido a AdGuard Home»)
Sin la VPN	**403 Forbidden**

Comprueba antes de seguir

Los dos contenedores están en `vpn_vpn` con sus IPs, y el panel solo se abre con la VPN.

Paso 6 Episodio 13

El asistente de configuración

Por qué: cinco pantallas.

Con la VPN activada, en `https://adguard.TU_DOMINIO`:

1. **Bienvenido** → Empezar.
2. **Interfaz web de administración:** interfaz «Todas las interfaces», **puerto 3000**. **Servidor DNS:** «Todas las interfaces», **puerto 53**. Es importante que el panel siga en el 3000: es el que publicamos en el compose. Si eliges el 80, Nginx dejará de encontrarlo.
3. **Autenticación:** usuario TU_USUARIO_ADGUARD (no `admin`) y una contraseña larga, generada con tu gestor. Guárdalas.
4. **Configura tus dispositivos:** esta pantalla explica cómo poner AdGuard como DNS en un router o un móvil. No hace falta: lo hará wg-easy (paso 10). Siguiente.
5. **Todo listo** → Abrir el panel. Entra con tu usuario.

¿«Todas las interfaces» no es peligroso?

Dentro del contenedor, «todas las interfaces» solo significa sus direcciones de Docker. Quién puede llegar desde fuera lo decide el compose: el panel solo por `127.0.0.1:8083` (y Nginx con `solo-vpn`) y el DNS solo por la red `vpn_vpn`.

Comprueba antes de seguir

Entras al panel de AdGuard con tu usuario, a través de `https://adguard.TU_DOMINIO`.

PARTE C

Configurar

Tres ajustes que marcan la diferencia: preguntar a Internet de forma cifrada, elegir buenas listas de bloqueo y dejar el registro con un tamaño razonable.

Paso 7 Episodio 13

DNS cifrado hacia Internet

Por qué: AdGuard reenvía a Internet las consultas que no bloquea.

En el panel: **Configuración → Configuración DNS**. En **Servidores DNS de subida**, borra lo que haya y escribe:

PANEL DE IONOS

```
https://dns.quad9.net/dns-query
https://cloudflare-dns.com/dns-query
```

- Debajo, en el modo, elige **Solicitudes paralelas**: pregunta a los dos a la vez y usa la respuesta que llegue antes.
- Pulsa **Probar servidores DNS de subida**: deben salir los dos en verde.
- En **Servidores DNS de arranque** (*bootstrap*) deja los que vienen: solo sirven para averiguar la IP de los dos anteriores.
- Guarda (**Aplicar**).

Comprueba antes de seguir

La prueba de servidores de subida sale correcta para los dos.

Paso 8 Episodio 13

Listas de bloqueo

Por qué: AdGuard trae activada su propia lista.

1. **Filtros → Listas de bloqueo DNS**.
2. Comprueba que **AdGuard DNS filter** está activada.
3. **Añadir lista de bloqueo → Elegir de la lista**.
4. Marca **OISD Blocklist Big** (en el apartado «General») y guarda.
5. Pulsa **Buscar actualizaciones**. Arriba se verá el número de reglas de cada lista.

Lista	Qué bloquea
AdGuard DNS filter	Anuncios y rastreadores, mantenida por AdGuard
OISD Blocklist Big	Una recopilación muy amplia de anuncios, rastreo, malware y estafas, revisada para no romper webs

Si quieres más protección

En **Configuración** → **Configuración general** puedes activar además la **Navegación segura** (webs de malware y *phishing*) y, si lo usa tu familia, el **Control parental** y la **Búsqueda segura**. Pruébalas de una en una.

Comprueba antes de seguir

Hay dos listas activas y cada una muestra su número de reglas y la fecha de actualización.

Paso 9 Episodio 13

Ajustes recomendados

Por qué: tres ajustes de **Configuración** pensados para un servidor pequeño y para tu privacidad.

Dónde	Qué poner	Por qué
Configuración general → **Registro de consultas**	Conservar **7 días**	El registro guarda cada web consultada por cada dispositivo. Es útil para resolver problemas, pero son datos personales: no hace falta guardarlos más
Configuración general → **Estadísticas**	Conservar **30 días**	Solo números (cuántas consultas, cuántas bloqueadas): ocupan poco
Configuración DNS → **Configuración de caché**	Activada, tamaño **4 MB** (4194304 bytes)	Las respuestas se guardan un tiempo: la segunda vez que alguien visita una web, la respuesta es instantánea
Configuración DNS → **DNSSEC**	Activado	Comprueba que las respuestas que vienen de Internet están firmadas por su dueño, cuando el dominio lo admite

Todos tus dispositivos salen con la misma IP

En el registro verás todas las consultas como si vinieran de `172.30.0.2`, la IP de wg-easy, sea del PC o del móvil. Es el mismo enmascaramiento (NAT) que viste en la Guía 06 con `$SSH_CLIENT`. Por eso, en este montaje, no se pueden poner reglas distintas por dispositivo. Para lo que buscamos (bloquear en todos por igual) no importa.

Comprueba antes de seguir

Registro a 7 días, estadísticas a 30, caché activada y DNSSEC activado.

PARTE D

Usarlo en la VPN

Le decimos a wg-easy que entregue AdGuard como DNS a tus dispositivos, actualizamos los que ya tenías y comprobamos que bloquea.

Paso 10 Episodio 13

Cambiar el DNS de la VPN

Por qué: el DNS que usa cada dispositivo va escrito en su configuración de la VPN.

En el panel de wg-easy (`https://vpn.TU_DOMINIO`, con la VPN activada):

1. En el **panel de administración**, en la configuración que se aplica a los clientes nuevos, cambia **DNS** a `172.30.0.3` (quita `1.1.1.1` y `9.9.9.9`). Guarda.
2. Abre cada cliente que ya tengas (`pc-casa`, `movil`…): si tiene su propio campo **DNS**, cámbialo también a `172.30.0.3` y guarda.

Los nombres de los menús cambian entre versiones

En la serie 15 de wg-easy, los ajustes por defecto de los clientes están en el panel de administración, y cada cliente se puede editar por separado. Si no encuentras el campo DNS, busca en la documentación de tu versión «client DNS». Lo importante es el resultado del paso 11: que la configuración descargada lleve `DNS = 172.30.0.3`.

Un solo DNS, a propósito

No pongas un segundo DNS de reserva (por ejemplo `1.1.1.1`) junto a AdGuard. Windows y Android preguntan a veces a los dos a la vez, y por el segundo se colarían los anuncios. Si AdGuard se cae, lo verás en seguida (no se abre ninguna web con la VPN) y lo arreglas.

Comprueba antes de seguir

En wg-easy, el DNS por defecto y el de cada cliente es `172.30.0.3`.

Paso 11 Episodio 13

Actualizar tus dispositivos

Por qué: el cambio de DNS está en el servidor, pero cada dispositivo tiene guardada su configuración antigua.

Dispositivo	Qué hacer
PC (Windows)	En wg-easy, descarga otra vez el `.conf` del cliente. En WireGuard: desactiva el túnel, bórralo y **Importar túnel(es) desde archivo** con el nuevo. Actívalo y borra el `.conf` de Descargas
Móvil	En la app WireGuard, borra el túnel y vuelve a escanear el código QR del cliente desde el panel de wg-easy

Comprueba en el PC que la configuración nueva lleva el DNS correcto: en WireGuard, selecciona el túnel y mira la línea **DNS servers**; debe decir `172.30.0.3`.

> **Comprueba antes de seguir**
> Los dos dispositivos conectan de nuevo y muestran `172.30.0.3` como servidor DNS.

Paso 12 Episodio 13

Comprobar el bloqueo

Por qué: con la VPN activada, preguntamos por un dominio de anuncios conocido y por uno normal.

EN TU PC · POWERSHELL

```
nslookup doubleclick.net
nslookup www.wikipedia.org
```

SALIDA ESPERADA

```
Servidor:  UnKnown
Address:  172.30.0.3

Nombre:  doubleclick.net
Addresses:  ::
          0.0.0.0

Servidor:  UnKnown
Address:  172.30.0.3

Respuesta no autoritativa:
Nombre:  dyna.wikimedia.org
Addresses:  ...
```

Ahora navega un poco por una web de noticias con anuncios, con y sin VPN, y compara. Después, en el panel de AdGuard, mira el **Panel de control**: verás las consultas y el porcentaje bloqueado, y en el **Registro de consultas**, cada dominio bloqueado en rojo.

> **Comprueba antes de seguir**
> - Con VPN: `doubleclick.net` da `0.0.0.0` y responde `172.30.0.3`.
> - Las webs normales funcionan.
> - El panel muestra consultas y un porcentaje de bloqueo (lo normal es entre un 10 % y un 30 %).

Paso 13 Episodio 13

Cuando se bloquea algo que necesitas

Por qué: a veces una web o una app deja de funcionar bien porque usa un dominio que está en una lista (un enlace de un correo de ofertas, una app de banco, un vídeo incrustado).

1. Reproduce el fallo con la VPN activada.
2. En AdGuard, **Registro de consultas**: busca las líneas en rojo de ese momento. Cerca del nombre de la web verás el dominio bloqueado.

3. Pasa el ratón por la línea y pulsa **Desbloquear**. AdGuard crea una regla de excepción.
4. Vuelve a probar. Las excepciones se ven y se borran en **Filtros → Reglas personalizadas**, con este formato:

CONTENIDO DEL ARCHIVO · FILTROS → REGLAS PERSONALIZADAS

```
@@||ejemplo-que-necesito.com^
```

Para descartar AdGuard en un momento

En el panel de control de AdGuard hay un botón para **Desactivar la protección** durante unos minutos. Si con ella desactivada la web funciona, el culpable es una lista; si sigue fallando, el problema es otro.

Comprueba antes de seguir

Sabes encontrar en el registro qué se ha bloqueado y crear una excepción.

Paso 14 Episodio 13

Comprobación final e instantánea

Por qué: repaso, reinicio y la instantánea que cierra la temporada de la VPN.

EN EL SERVIDOR

```
docker compose -f /opt/apps/vpn/compose.yaml ps
docker compose -f /opt/apps/adguard/compose.yaml ps
docker network inspect vpn_vpn --format '{{range .Containers}}{{.Name}} {{.IPv4Address}}
{{"\n"}}{{end}}'
sudo ss -tulpn | grep -E ':(53|8083) '
sudo nginx -t
```

Comprobación	Resultado correcto
Contenedores	`wg-easy` y `adguard` en `Up`
Red `vpn_vpn`	wg-easy en `172.30.0.2` y adguard en `172.30.0.3`
Puertos	**Nada** en el 53 del servidor (salvo `127.0.0.53`, el DNS propio de Ubuntu); el 8083 solo en `127.0.0.1`
Panel	Con VPN se abre; sin VPN, 403
Bloqueo	Con VPN, `nslookup doubleclick.net` da `0.0.0.0`

Reinicia el servidor y, al volver, conecta la VPN y repite el `nslookup`: AdGuard debe haber arrancado solo, después de wg-easy. Después, en IONOS, crea la instantánea `07-adguard`.

Copia de AdGuard

Toda la configuración está en `/opt/apps/adguard` (sobre todo `datos/conf/AdGuardHome.yaml`, que guarda tus listas, excepciones y el usuario con la contraseña cifrada). Inclúyela en tus copias de seguridad.

Comprueba antes de seguir

Todos tus dispositivos conectados a la VPN navegan sin anuncios ni rastreadores, con DNS cifrado y un panel que solo ves tú. Con esto termina la temporada de la VPN. La siguiente monta tu propio servidor de correo.

ANEXOS

Para cuando algo no sale

Si con la VPN no se abre ninguna web, lo primero: `docker compose -f /opt/apps/adguard/compose.yaml ps`. Si AdGuard está caído, tus dispositivos no pueden resolver ningún nombre.

Anexo A · Problemas frecuentes

Lo que ves	Qué pasa	Qué hacer
`network vpn_vpn declared as external, but could not be found`	La red de la VPN no existe (la VPN está parada o se llama de otra forma)	`docker network ls`; levanta antes la VPN (`cd /opt/apps/vpn && docker compose up -d`) o corrige el nombre en el compose
`Address already in use` al arrancar, con la IP 172.30.0.3	Otro contenedor tiene ya esa IP en la red	`docker network inspect vpn_vpn` y elige otra libre (`172.30.0.4`), y úsala también en wg-easy
Con la VPN no se abre ninguna web, pero `ping 1.1.1.1` responde	Falla el DNS: AdGuard parado o el DNS de los dispositivos mal puesto	Comprueba AdGuard (`ps` y `logs`) y que el túnel del dispositivo dice `DNS servers: 172.30.0.3`
El túnel dice `DNS servers: 172.30.0.3` pero no resuelve nada	El dispositivo no envía por el túnel el tráfico hacia `172.30.0.3` (cambiaste las «IPs permitidas» del cliente y ya no es `0.0.0.0/0`)	En wg-easy, en las IPs permitidas del cliente, añade `172.30.0.0/24` y vuelve a importar la configuración
`nslookup` responde desde otra dirección, no `172.30.0.3`	El dispositivo usa la configuración antigua de la VPN	Paso 11: vuelve a importar el `.conf` o a escanear el QR
El panel da **502**	Nginx no llega al 8083: AdGuard parado, o en el asistente elegiste otro puerto para el panel	Si cambiaste el puerto, edita `datos/conf/AdGuardHome.yaml` (apartado `http: address`) a `0.0.0.0:3000` y reinicia el contenedor
El panel da **403** con la VPN activada	El navegador no va por la VPN, o el snippet no coincide con tu rango	Igual que en la Guía 06: comprueba `ifconfig.me` y el `adguard.access.log`
La prueba de servidores DNS de subida falla	El servidor no llega a ellos por HTTPS	Desde el servidor: `curl -I https://dns.quad9.net`. Si falla, revisa la salida a Internet del servidor
Una app o web deja de funcionar	Un dominio que necesita está en una lista	Paso 13: registro de consultas y excepción

Anexo B · Puertos y redes al terminar esta guía

Puerto / dirección	Servicio	Accesible desde	Qué lo protege
443 → `adguard.TU_DOMINIO`	Panel de AdGuard (por Nginx)	Solo la VPN	`solo-vpn.conf` + usuario y contraseña
127.0.0.1:8083	Panel de AdGuard (contenedor)	Solo el propio servidor	Publicado en 127.0.0.1
172.30.0.3:53	DNS de AdGuard	Solo la red `vpn_vpn` (es decir, la VPN)	Sin publicar en el servidor: no es un DNS abierto

No se abre ningún puerto nuevo en el firewall de IONOS ni en UFW.

Lista de puertos internos

Puerto interno	Aplicación	Subdominio
127.0.0.1:8081	IT-Tools	`herramientas.TU_DOMINIO`
127.0.0.1:8082	WordPress	`blog.TU_DOMINIO`
127.0.0.1:8083	AdGuard Home	`adguard.TU_DOMINIO` (solo VPN)
127.0.0.1:51821	wg-easy	`vpn.TU_DOMINIO` (solo VPN)

Anexo C · Chuleta de AdGuard Home

Para...	Cómo
Estado y registros	`docker compose -f /opt/apps/adguard/compose.yaml ps` · `... logs --tail 50`
¿Qué DNS estoy usando? (PC)	`nslookup doubleclick.net`: debe responder `172.30.0.3` con `0.0.0.0`
Ver lo que se ha bloqueado	Panel → **Registro de consultas**, filtro «Bloqueado»
Permitir un dominio	Registro → **Desbloquear**, o regla `@@||dominio.com^`
Bloquear un dominio a mano	Filtros → Reglas personalizadas: `||dominio.com^`
Pausar el filtrado unos minutos	Panel de control → **Desactivar la protección**
Actualizar AdGuard	Lee las notas de la versión, haz copia de `/opt/apps/adguard`, cambia la versión en el compose y `docker compose pull && docker compose up -d`
Si tras un `down` de la VPN AdGuard no arranca	Primero `cd /opt/apps/vpn && docker compose up -d`, después `cd /opt/apps/adguard && docker compose up -d`



Si te atascas

Pega: la salida de `docker network inspect vpn_vpn` (la parte de `Containers`), el `nslookup doubleclick.net` desde el PC con la VPN, y las últimas líneas del registro de AdGuard.