EBENEZER IT · GUÍAS PASO A PASO

GUÍA 08 · EPISODIOS 14 A 16

Tu propio servidor de correo, con tu dominio

Postfix para enviar y recibir, Dovecot para guardar y leer, MariaDB y PostfixAdmin para gestionar dominios y buzones desde un panel, y DKIM, SPF y DMARC para que tus correos lleguen a la bandeja de entrada y no a spam. Seguro desde el primer día: solo puertos cifrados y nunca contraseñas sin cifrar.

~4–6 horas
la primera vez, en dos o tres sesiones

Requisito
Guías 01, 02 y 06 terminadas

Ubuntu 24.04 LTS
Postfix · Dovecot · OpenDKIM

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 · Cómo funciona el correo y preparación

Paso 01 Cómo viaja un correo

Paso 02 Comprobar el punto de partida

Paso 03 Registros DNS: MX y SPF

Paso 04 Abrir los puertos del correo

Paso 05 Certificados para el correo y el panel

Parte B · Postfix, Dovecot, MariaDB y PostfixAdmin

Paso 06 MariaDB y la base de datos del correo

Paso 07 PHP y PostfixAdmin

Paso 08 Configurar PostfixAdmin y su sitio

Paso 09 Crear las tablas y tu administrador

Paso 10 El usuario vmail y la carpeta del correo

Paso 11 Instalar Postfix

Paso 12 Postfix: las consultas a la base de datos

Paso 13 Postfix: la configuración principal

Paso 14 Postfix: el puerto 587 para enviar

Paso 15 Instalar y configurar Dovecot

Paso 16 Tu primer dominio y buzón

Paso 17 Primeras pruebas: enviar y recibir

Parte C · DKIM, DMARC y el 10/10

Paso 18 OpenDKIM: instalar y crear la clave

Paso 19 OpenDKIM: tablas y conexión con Postfix

Paso 20 Publicar la clave DKIM

Paso 21 DMARC

Paso 22 La prueba de verdad: mail-tester y Gmail

Paso 23 Comprobación final e instantánea

Anexos

A Problemas frecuentes

B Puertos al terminar esta guía

C Chuleta del correo

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-08-correo.html**. Al copiar desde un PDF se pueden romper los saltos de línea y las comillas.

ANTES DE EMPEZAR

Qué vas a construir

Al terminar tendrás correo con tu dominio (`tu.nombre@TU_DOMINIO`) en tu propio servidor: buzones para ti y para quien quieras, un panel para crearlos sin tocar la terminal, y la configuración que hace que Gmail y Outlook confíen en tus correos.

Pieza	Qué hace	Dónde vive
Postfix	El «cartero»: recibe el correo que llega de Internet (puerto 25) y envía el de tus usuarios (puerto 587)	Servicio del sistema
Dovecot	El «buzón»: guarda los correos en disco, comprueba las contraseñas y deja leerlos con IMAP cifrado (puerto 993)	Servicio del sistema
MariaDB	La base de datos con la lista de dominios, buzones, contraseñas cifradas y alias	Servicio del sistema, solo en `127.0.0.1`
PostfixAdmin	El panel web para crear dominios, buzones y alias	`https://postfixadmin.TU_DOMINIO`, solo desde la VPN
OpenDKIM	Firma cada correo que sale, para demostrar que es tuyo de verdad	Servicio del sistema, solo en `127.0.0.1`

El recorrido de un correo

1. **Envías** desde tu móvil: se conecta al puerto **587**, cifra la conexión y se identifica con tu buzón y tu contraseña. Postfix comprueba la contraseña preguntando a Dovecot.
2. OpenDKIM **firma** el correo y Postfix lo entrega al servidor del destinatario por el puerto **25**.
3. Cuando alguien te **escribe**, su servidor busca el registro **MX** de tu dominio, llega a tu Postfix por el 25, y Postfix pregunta a MariaDB si ese buzón existe.
4. Si existe, se lo pasa a Dovecot, que lo guarda en `/var/mail/vhosts/TU_DOMINIO/usuario/`.
5. **Lees** el correo desde el móvil o el PC por IMAP cifrado, en el puerto **993**.

Lo que dejamos para la Guía 09

El **webmail** (Roundcube), la configuración de tus programas de correo, el filtro de spam de entrada y **fail2ban** para el correo van en la Guía 09 (episodios 17 y 18). Aquí terminamos con el correo funcionando, seguro y con un 10/10 en las pruebas.

Valores de esta guía

Valor	Qué es
TU_DOMINIO	Tu dominio

Valor	Qué es
`TU_IP`	La IP del VPS
`TU_BUZON`	La parte de antes de la @ de tu primer buzón (por ejemplo, tu nombre)
`TU_CORREO_EXTERNO`	Un correo tuyo de fuera (Gmail, Outlook) para las pruebas
`TU_VERSION_POSTFIXADMIN`	La versión de PostfixAdmin (paso 7)
`TU_PASS_BD_ADMIN`	Contraseña del usuario de la base de datos para PostfixAdmin (paso 6)
`TU_PASS_BD_CORREO`	Contraseña del usuario de solo lectura para Postfix y Dovecot (paso 6)
`TU_HASH_SETUP`	La huella de la contraseña del asistente de PostfixAdmin (paso 9)

Recordatorio

- Haz una instantánea en IONOS antes de empezar.
- Esta guía da por hechos el nombre del servidor `mail.TU_DOMINIO`, el DNS inverso (PTR) y el puerto 25 desbloqueado (Guía 01), Nginx con `tls.conf` y la carpeta `/var/www/letsencrypt` (Guía 02), y la VPN con el snippet `solo-vpn.conf` (Guía 06).
- 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

Cómo funciona el correo y preparación

Antes de instalar nada, entendemos qué piezas deciden si un correo llega o va a spam, y dejamos listos el DNS, los puertos y los certificados. Corresponde al **episodio 14**.

Paso 1 Episodio 14

Cómo viaja un correo

Por qué: el 90 % de los problemas del correo propio no están en el servidor, sino en el DNS.

Registro	Qué responde	Si falta o está mal
MX	«El correo de `TU_DOMINIO` se entrega en `mail.TU_DOMINIO`»	Nadie te puede escribir
PTR (DNS inverso)	«La IP `TU_IP` se llama `mail.TU_DOMINIO`»	Muchos servidores rechazan tu correo o lo mandan a spam
SPF	«Solo mi servidor de correo puede enviar en nombre de `TU_DOMINIO`»	Cualquiera puede hacerse pasar por ti; tus correos parecen sospechosos
DKIM	Una firma en cada correo: «lo envió de verdad `TU_DOMINIO` y nadie lo ha modificado»	Gmail no puede comprobar que el correo es tuyo
DMARC	«Si un correo dice ser de `TU_DOMINIO` y no pasa SPF ni DKIM, haz esto con él»	Desde 2024, Gmail y Yahoo lo exigen a quien envía mucho correo; sin él, peor reputación

El PTR ya lo configuraste en la Guía 01. MX y SPF van en el paso 3, DKIM en el 20 y DMARC en el 21.

Los tres puertos del correo

Puerto	Para qué	Quién lo usa
25 (SMTP)	Recibir correo de otros servidores	Gmail, Outlook... cuando te escriben. Sin contraseña: es la «puerta del buzón»
587 (Submission)	Enviar tu correo	Tu móvil y tu PC. Siempre cifrado y con usuario y contraseña
993 (IMAPS)	Leer tu correo	Tu móvil y tu PC. Siempre cifrado

¿Y los puertos 110, 143 y 465?
El **110** (POP3) y el **143** (IMAP) admiten conexiones sin cifrar: no los abriremos. El **465** es una alternativa al 587 que no hace falta. Un puerto, un motivo.

Comprueba antes de seguir
Sabes explicar qué hace cada uno de los cinco registros y por qué el 25 no pide contraseña pero el 587 sí.

Paso 2 Episodio 14

Comprobar el punto de partida

Por qué: tres cosas de la Guía 01 sin las que el correo no funcionará: el nombre del servidor, el DNS inverso y el puerto 25 de salida.

EN EL SERVIDOR

```
hostname -f
timeout 5 bash -c '</dev/tcp/gmail-smtp-in.l.google.com/25' && echo "Puerto 25: ABIERTO" ||
echo "Puerto 25: BLOQUEADO"
sudo ss -tlnp | grep -E ':(25|587|993|3306) '
```

EN TU PC · POWERSHELL

```
Resolve-DnsName TU_IP -Type PTR -Server 1.1.1.1
Resolve-DnsName mail.TU_DOMINIO -Type A -Server 1.1.1.1
```

Comprueba antes de seguir

- `hostname -f` responde `mail.TU_DOMINIO`.
- El puerto 25 sale **ABIERTO**.
- El PTR de TU_IP es `mail.TU_DOMINIO` y `mail.TU_DOMINIO` apunta a TU_IP.

Paso 3 Episodio 14

Registros DNS: MX y SPF

Por qué: el MX dice dónde entregarte el correo y el SPF quién puede enviarlo.

En el DNS de tu dominio en IONOS, **borra** los registros **MX** que apunten a servidores de IONOS (`mx00.ionos.es`, `mx01.ionos.es`...) y cualquier **TXT** que empiece por `v=spf1`. Después crea estos:

Tipo	Nombre de host	Valor	Prioridad / TTL
MX	`@`	`mail.TU_DOMINIO`	Prioridad 10 · 1 hora
TXT	`@`	`v=spf1 mx -all`	1 hora
TXT	`mail`	`v=spf1 a -all`	1 hora
A	`postfixadmin`	TU_IP	1 hora

Un solo SPF por nombre

Si dejas dos registros TXT con `v=spf1` en el mismo nombre, los dos dejan de valer y el SPF falla. Si en el futuro envías también desde otro servicio (por ejemplo, un boletín), se añade a **ese mismo** registro.

EN TU PC · POWERSHELL

```
Resolve-DnsName TU_DOMINIO -Type MX -Server 1.1.1.1
Resolve-DnsName TU_DOMINIO -Type TXT -Server 1.1.1.1
```

Comprueba antes de seguir

- El MX de `TU_DOMINIO` es solo `mail.TU_DOMINIO` (sin servidores de IONOS).
- Hay un único TXT con `v=spf1 mx -all`.

Paso 4 Episodio 14

Abrir los puertos del correo

Por qué: los tres puertos del paso 1, en las dos capas de cortafuegos: la de IONOS y UFW.

En IONOS, **Red → Políticas de firewall**, añade tres reglas de entrada TCP desde cualquier origen: **25**, **587** y **993**. En el servidor:

EN EL SERVIDOR

```
sudo ufw allow 25/tcp comment 'SMTP: correo entrante'
sudo ufw allow 587/tcp comment 'SMTP: envio con usuario'
sudo ufw allow 993/tcp comment 'IMAPS: leer el correo'
sudo ufw status | grep -E '\b(25|110|143|465|587|993|995)/tcp'
```

SALIDA ESPERADA

```
25/tcp                     ALLOW       Anywhere                   # SMTP: correo entrante
587/tcp                    ALLOW       Anywhere                   # SMTP: envio con usuario
993/tcp                    ALLOW       Anywhere                   # IMAPS: leer el correo
25/tcp (v6)                ALLOW       Anywhere (v6)              # SMTP: correo entrante
...
```

Comprueba antes de seguir

En UFW y en IONOS están abiertos el 25, el 587 y el 993, y ningún otro puerto de correo.

Paso 5 Episodio 14

Certificados para el correo y el panel

Por qué: Postfix y Dovecot usarán el certificado de `mail.TU_DOMINIO` para cifrar; PostfixAdmin tendrá el suyo.

5.1 · Sitios para Let's Encrypt

`mail.TU_DOMINIO` solo necesita el puerto 80 para pedir su certificado; quien entre ahí con el navegador va a tu web.

EN EL SERVIDOR

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

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

    location / {
        return 301 https://www.TU_DOMINIO;
    }
}
EOF
sudo tee /etc/nginx/sites-available/postfixadmin.TU_DOMINIO > /dev/null <<'EOF'
server {
    listen 80;
    listen [::]:80;
    server_name postfixadmin.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/mail.TU_DOMINIO /etc/nginx/sites-enabled/
sudo ln -s /etc/nginx/sites-available/postfixadmin.TU_DOMINIO /etc/nginx/sites-enabled/
sudo nginx -t && sudo systemctl reload nginx
```

5.2 · Los certificados

EN EL SERVIDOR

```
sudo certbot certonly --webroot -w /var/www/letsencrypt -d mail.TU_DOMINIO --dry-run
sudo certbot certonly --webroot -w /var/www/letsencrypt -d mail.TU_DOMINIO --cert-name
mail.TU_DOMINIO
sudo certbot certonly --webroot -w /var/www/letsencrypt -d postfixadmin.TU_DOMINIO --cert-name
postfixadmin.TU_DOMINIO
```

5.3 · Avisar a Postfix y Dovecot cuando se renueve

EN EL SERVIDOR

```
sudo tee /etc/letsencrypt/renewal-hooks/deploy/recargar-correo.sh > /dev/null <<'EOF'
#!/bin/sh
# Tras renovar el certificado del correo, Postfix y Dovecot cargan el nuevo
case "$RENEWED_LINEAGE" in
    */mail.TU_DOMINIO) systemctl reload postfix dovecot ;;
esac
EOF
sudo chmod +x /etc/letsencrypt/renewal-hooks/deploy/recargar-correo.sh
```

Comprueba antes de seguir

`sudo certbot certificates` muestra `mail.TU_DOMINIO` y `postfixadmin.TU_DOMINIO`, y el script del *hook* es ejecutable (`ls -l` muestra `-rwxr-xr-x`).

PARTE B

Postfix, Dovecot, MariaDB y PostfixAdmin

Montamos el corazón del correo: la base de datos con los buzones, el panel para gestionarlos, el cartero (Postfix) y el buzón (Dovecot). Al final enviarás y recibirás tu primer correo. Corresponde al **episodio 15**.

Paso 6 Episodio 15

MariaDB y la base de datos del correo

Por qué: los dominios, buzones y alias se guardan en una base de datos.

EN EL SERVIDOR

```
sudo apt update
sudo apt install mariadb-server
sudo ss -tlnp | grep 3306
```

SALIDA ESPERADA

```
LISTEN 0  80  127.0.0.1:3306  0.0.0.0:*  users:(("mariadbd",...))
```

MariaDB escucha solo en `127.0.0.1`: no es accesible desde Internet. (Este MariaDB es del sistema; el de WordPress de la Guía 05 vive en su contenedor y no publica ningún puerto, así que no chocan.) Genera las dos contraseñas y apúntalas en tu gestor:

EN EL SERVIDOR

```
openssl rand -hex 20
openssl rand -hex 20
```

La primera será `TU_PASS_BD_ADMIN` y la segunda `TU_PASS_BD_CORREO`. Ahora la base de datos y los usuarios:

EN EL SERVIDOR

```
sudo mariadb <<'EOF'
CREATE DATABASE mailserver CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;
CREATE USER 'postfixadmin'@'localhost' IDENTIFIED BY 'TU_PASS_BD_ADMIN';
GRANT ALL PRIVILEGES ON mailserver.* TO 'postfixadmin'@'localhost';
CREATE USER 'mailuser'@'127.0.0.1' IDENTIFIED BY 'TU_PASS_BD_CORREO';
GRANT SELECT ON mailserver.* TO 'mailuser'@'127.0.0.1';
FLUSH PRIVILEGES;
EOF
sudo mariadb -e "SELECT user, host FROM mysql.user;"
```

Comprueba antes de seguir

La consulta final muestra `postfixadmin | localhost` y `mailuser | 127.0.0.1`, y MariaDB escucha solo en `127.0.0.1:3306`.

Paso 7 Episodio 15

PHP y PostfixAdmin

Por qué: PostfixAdmin es una aplicación PHP.

EN EL SERVIDOR

```
sudo apt install php8.3-fpm php8.3-mysql php8.3-mbstring
systemctl is-active php8.3-fpm
```

Para la versión: `github.com/postfixadmin/postfixadmin/releases`, la marcada como **Latest** (por ejemplo `3.3.15`). Apúntala como `TU_VERSION_POSTFIXADMIN`, sin la palabra `postfixadmin-` delante.

EN EL SERVIDOR

```
cd /tmp
wget https://github.com/postfixadmin/postfixadmin/archive/refs/tags/postfixadmin-
TU_VERSION_POSTFIXADMIN.tar.gz
tar -xzf postfixadmin-TU_VERSION_POSTFIXADMIN.tar.gz
ls -d postfixadmin-*/
sudo mv postfixadmin-postfixadmin-TU_VERSION_POSTFIXADMIN /var/www/postfixadmin
sudo chown -R root:root /var/www/postfixadmin
sudo mkdir -p /var/www/postfixadmin/templates_c
sudo chown www-data:www-data /var/www/postfixadmin/templates_c
rm postfixadmin-TU_VERSION_POSTFIXADMIN.tar.gz
```

Comprueba antes de seguir

`php8.3-fpm` está `active` y existe `/var/www/postfixadmin/public/index.php`.

Paso 8 Episodio 15

Configurar PostfixAdmin y su sitio

Por qué: PostfixAdmin se configura en un archivo propio, `config.local.php`, que no se pisa al actualizar.

Créalo con `sudo nano /var/www/postfixadmin/config.local.php`. Deja `TU_HASH_SETUP` tal cual: lo cambiarás en el paso 9.

CONTENIDO DEL ARCHIVO · /VAR/WWW/POSTFIXADMIN/CONFIG.LOCAL.PHP

```
<?php
// Ajustes propios de PostfixAdmin (no se pisan al actualizar)
$CONF['configured'] = true;
$CONF['setup_password'] = 'TU_HASH_SETUP';

// Base de datos
$CONF['database_type'] = 'mysqli';
$CONF['database_host'] = 'localhost';
$CONF['database_user'] = 'postfixadmin';
$CONF['database_password'] = 'TU_PASS_BD_ADMIN';
$CONF['database_name'] = 'mailserver';

// Contraseñas de los buzones: bcrypt, el formato que espera Dovecot
$CONF['encrypt'] = 'php_crypt:BLOWFISH';
$CONF['password_validation'] = array('/.{12}/' => 'password_too_short 12');

// Carpetas: dominio/usuario/ (lo mismo que espera Dovecot)
$CONF['domain_path'] = 'YES';
$CONF['domain_in_mailbox'] = 'NO';

// Alias que se crean con cada dominio nuevo, apuntando a tu buzón
$CONF['default_aliases'] = array(
    'abuse'      => 'TU_BUZON@TU_DOMINIO',
    'hostmaster' => 'TU_BUZON@TU_DOMINIO',
    'postmaster' => 'TU_BUZON@TU_DOMINIO',
    'webmaster'  => 'TU_BUZON@TU_DOMINIO',
);

// Pie de página
$CONF['footer_text'] = 'Volver a TU_DOMINIO';
$CONF['footer_link'] = 'https://www.TU_DOMINIO';
```

EN EL SERVIDOR

```
sudo chown root:www-data /var/www/postfixadmin/config.local.php
sudo chmod 640 /var/www/postfixadmin/config.local.php
```

El archivo tiene la contraseña de la base de datos: lo lee PHP (`www-data`) y nadie más. Ahora el sitio completo:

EN EL SERVIDOR

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

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

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

# HTTPS: PostfixAdmin, solo desde la VPN
server {
    listen 443 ssl http2;
    listen [::]:443 ssl http2;
    server_name postfixadmin.TU_DOMINIO;

    ssl_certificate     /etc/letsencrypt/live/postfixadmin.TU_DOMINIO/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/postfixadmin.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;

    root /var/www/postfixadmin/public;
    index index.php;

    # Se activa en el paso 9, cuando el asistente ya ha terminado
    # location = /setup.php { deny all; }

    location / {
        try_files $uri $uri/ =404;
    }

    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php/php8.3-fpm.sock;
    }

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

EN EL SERVIDOR

```
sudo nginx -t && sudo systemctl reload nginx
```

Comprueba antes de seguir

- Con la VPN activada, `https://postfixadmin.TU_DOMINIO` abre PostfixAdmin (aunque diga que falta configurar la base de datos: es normal).
- Sin la VPN, **403 Forbidden**.

Paso 9 Episodio 15

Crear las tablas y tu administrador

Por qué: el asistente de PostfixAdmin crea las tablas en la base de datos y tu cuenta de administrador del panel.

1. Con la VPN activada, abre `https://postfixadmin.TU_DOMINIO/setup.php`.
2. En **Generate setup_password**, escribe una contraseña larga (guárdala en tu gestor) y pulsa el botón. Te mostrará una línea como `$CONF['setup_password'] = '$2y$10$...';`.
3. Copia solo la parte entre comillas (empieza por `$2y$`) y ponla en `config.local.php` en lugar de `TU_HASH_SETUP`: `sudo nano /var/www/postfixadmin/config.local.php`.
4. Recarga `setup.php` y entra con esa contraseña. Verás el resultado de la comprobación del entorno y la creación de las tablas (*Database update*): todo en verde.
5. En **Add Superadmin Account**: la contraseña del asistente, como administrador `TU_BUZON@TU_DOMINIO` y otra contraseña larga distinta (será la de entrar al panel).

Tu usuario del panel no es un buzón

El administrador de PostfixAdmin se escribe como una dirección de correo, pero no hace falta que el buzón exista. Es solo el nombre para entrar al panel.

Con el administrador creado, bloquea el asistente:

EN EL SERVIDOR

```
sudo sed -i 's|^    # location = /setup.php { deny all; }|    location = /setup.php { deny all;
}|' /etc/nginx/sites-available/postfixadmin.TU_DOMINIO
grep -n "setup.php" /etc/nginx/sites-available/postfixadmin.TU_DOMINIO
sudo nginx -t && sudo systemctl reload nginx
```

Comprueba antes de seguir

- Entras en `https://postfixadmin.TU_DOMINIO` con tu administrador.
- `https://postfixadmin.TU_DOMINIO/setup.php` da **403**, también con la VPN.

Paso 10 Episodio 15

El usuario vmail y la carpeta del correo

Por qué: los buzones no son usuarios de Ubuntu: son filas en la base de datos.

EN EL SERVIDOR

```
sudo groupadd -g 5000 vmail
sudo useradd -g vmail -u 5000 -d /var/mail/vhosts -s /usr/sbin/nologin vmail
sudo mkdir -p /var/mail/vhosts
sudo chown vmail:vmail /var/mail/vhosts
sudo chmod 770 /var/mail/vhosts
ls -ld /var/mail/vhosts
```

Comprueba antes de seguir

`ls -ld` muestra `drwxrwx--- ... vmail vmail ... /var/mail/vhosts`.

Paso 11 Episodio 15

Instalar Postfix

Por qué: al instalarlo, Postfix hace dos preguntas.

EN EL SERVIDOR

```
sudo apt install postfix postfix-mysql
```

Pregunta	Respuesta
Tipo genérico de configuración de correo	**Sitio de Internet** (*Internet Site*)
Nombre del sistema de correo	`TU_DOMINIO` (sin `mail.` delante)

Comprueba antes de seguir

`systemctl is-active postfix` responde `active` y `postconf mail_version` muestra la versión.

Paso 12 Episodio 15

Postfix: las consultas a la base de datos

Por qué: Postfix tiene que hacer tres preguntas a MariaDB: «¿este dominio es mío?», «¿existe este buzón?» y «¿esta dirección es un alias de otra?».

EN EL SERVIDOR

```
sudo mkdir -p /etc/postfix/sql

sudo tee /etc/postfix/sql/dominios.cf > /dev/null <<'EOF'
user = mailuser
password = TU_PASS_BD_CORREO
hosts = 127.0.0.1
dbname = mailserver
query = SELECT domain FROM domain WHERE domain='%s' AND backupmx = 0 AND active = 1
EOF

sudo tee /etc/postfix/sql/buzones.cf > /dev/null <<'EOF'
user = mailuser
password = TU_PASS_BD_CORREO
hosts = 127.0.0.1
dbname = mailserver
query = SELECT maildir FROM mailbox WHERE username='%s' AND active = 1
EOF

sudo tee /etc/postfix/sql/alias.cf > /dev/null <<'EOF'
user = mailuser
password = TU_PASS_BD_CORREO
hosts = 127.0.0.1
dbname = mailserver
query = SELECT goto FROM alias WHERE address='%s' AND active = 1
EOF

sudo chown root:postfix /etc/postfix/sql/*.cf
sudo chmod 640 /etc/postfix/sql/*.cf
ls -l /etc/postfix/sql/
```

Comprueba antes de seguir

`ls -l` muestra los tres archivos como `-rw-r----- root postfix`. Las consultas las probaremos en el paso 16, cuando exista un dominio.

Paso 13 Episodio 15

Postfix: la configuración principal

Por qué: en vez de editar `main.cf` a mano, usamos `postconf -e`: cambia cada ajuste en su sitio y no deja duplicados.

13.1 · Identidad

EN EL SERVIDOR

```
sudo postconf -e \
  'myhostname = mail.TU_DOMINIO' \
  'mydomain = TU_DOMINIO' \
  'myorigin = $mydomain' \
  'mydestination = localhost' \
  'inet_protocols = ipv4' \
  'smtpd_banner = $myhostname ESMTP' \
  'disable_vrfy_command = yes' \
  'smtpd_helo_required = yes' \
  'message_size_limit = 26214400' \
  'recipient_delimiter = +'
```

13.2 · Cifrado

EN EL SERVIDOR

```
sudo postconf -e \
  'smtpd_tls_cert_file = /etc/letsencrypt/live/mail.TU_DOMINIO/fullchain.pem' \
  'smtpd_tls_key_file = /etc/letsencrypt/live/mail.TU_DOMINIO/privkey.pem' \
  'smtpd_tls_security_level = may' \
  'smtpd_tls_auth_only = yes' \
  'smtpd_tls_protocols = ⩾TLSv1.2' \
  'smtpd_tls_mandatory_protocols = ⩾TLSv1.2' \
  'smtpd_tls_loglevel = 1' \
  'smtp_tls_security_level = may' \
  'smtp_tls_loglevel = 1'
```

13.3 · Quién puede enviar a través de tu servidor

EN EL SERVIDOR

```
sudo postconf -e \
  'smtpd_sasl_type = dovecot' \
  'smtpd_sasl_path = private/auth' \
  'smtpd_sasl_auth_enable = no' \
  'smtpd_sasl_security_options = noanonymous' \
  'smtpd_sasl_tls_security_options = noanonymous' \
  'smtpd_relay_restrictions = permit_mynetworks, permit_sasl_authenticated,
defer_unauth_destination'
```

13.4 · Dominios y buzones virtuales

EN EL SERVIDOR

```
sudo postconf -e \
  'virtual_mailbox_domains = mysql:/etc/postfix/sql/dominios.cf' \
  'virtual_mailbox_maps = mysql:/etc/postfix/sql/buzones.cf' \
  'virtual_alias_maps = mysql:/etc/postfix/sql/alias.cf' \
  'virtual_transport = lmtp:unix:private/dovecot-lmtp'
sudo postconf -n | wc -l
```

Comprueba antes de seguir

`sudo postconf -n` muestra todos los ajustes de este paso. No reinicies Postfix todavía: falta Dovecot.

Paso 14 Episodio 15

Postfix: el puerto 587 para enviar

Por qué: los servicios que Postfix escucha (25, 587…) se definen en `master.cf`.

EN EL SERVIDOR

```
sudo postconf -M submission/inet="submission inet n - y - - smtpd"
sudo postconf -P \
  "submission/inet/syslog_name=postfix/submission" \
  "submission/inet/smtpd_tls_security_level=encrypt" \
  "submission/inet/smtpd_sasl_auth_enable=yes" \
  "submission/inet/smtpd_client_restrictions=permit_sasl_authenticated,reject" \
  "submission/inet/smtpd_relay_restrictions=permit_sasl_authenticated,reject" \
  "submission/inet/milter_macro_daemon_name=ORIGINATING"
sudo postconf -Mf submission/inet
```

SALIDA ESPERADA

```
submission inet  n       -       y       -       -       smtpd
    -o syslog_name=postfix/submission
    -o smtpd_tls_security_level=encrypt
    -o smtpd_sasl_auth_enable=yes
    -o smtpd_client_restrictions=permit_sasl_authenticated,reject
    -o smtpd_relay_restrictions=permit_sasl_authenticated,reject
    -o milter_macro_daemon_name=ORIGINATING
```

Comprueba antes de seguir

La salida coincide con la de arriba.

Paso 15 Episodio 15

Instalar y configurar Dovecot

Por qué: Dovecot guarda el correo, comprueba contraseñas para sí mismo y para Postfix, y deja leer el correo por IMAP cifrado.

EN EL SERVIDOR

```
sudo apt install dovecot-imapd dovecot-lmtpd dovecot-mysql
grep -n "include_try local.conf" /etc/dovecot/dovecot.conf
grep -n "auth-system.conf.ext" /etc/dovecot/conf.d/10-auth.conf
```

15.1 · Desactivar los usuarios del sistema

EN EL SERVIDOR

```
sudo sed -i 's/^!include auth-system.conf.ext/#!include auth-system.conf.ext/'
/etc/dovecot/conf.d/10-auth.conf
grep -n "auth-system.conf.ext" /etc/dovecot/conf.d/10-auth.conf
```

Debe salir con `#` delante: `#!include auth-system.conf.ext`.

15.2 · Cómo consulta Dovecot la base de datos

EN EL SERVIDOR

```
sudo tee /etc/dovecot/dovecot-sql.conf.ext > /dev/null <<'EOF'
driver = mysql
connect = host=127.0.0.1 dbname=mailserver user=mailuser password=TU_PASS_BD_CORREO
default_pass_scheme = BLF-CRYPT

password_query = SELECT username AS user, password FROM mailbox WHERE username = '%u' AND
active = '1'
user_query = SELECT 5000 AS uid, 5000 AS gid FROM mailbox WHERE username = '%u' AND active =
'1'
EOF
sudo chown root:dovecot /etc/dovecot/dovecot-sql.conf.ext
sudo chmod 640 /etc/dovecot/dovecot-sql.conf.ext
```

15.3 · La configuración de Dovecot

Créala con `sudo nano /etc/dovecot/local.conf`:

CONTENIDO DEL ARCHIVO · /ETC/DOVECOT/LOCAL.CONF

```
# Ebenezer IT · Dovecot: buzones virtuales con MariaDB (Guía 08)
# Se carga al final, así que manda sobre conf.d/

protocols = imap lmtp
mail_location = maildir:/var/mail/vhosts/%d/%n
auth_username_format = %Lu

# Nunca contraseñas sin cifrar
disable_plaintext_auth = yes
auth_mechanisms = plain login

# Cifrado obligatorio, con el certificado de mail.TU_DOMINIO
ssl = required
ssl_cert = </etc/letsencrypt/live/mail.TU_DOMINIO/fullchain.pem
ssl_key = </etc/letsencrypt/live/mail.TU_DOMINIO/privkey.pem
ssl_min_protocol = TLSv1.2

# Usuarios y contraseñas: en MariaDB
passdb {
  driver = sql
  args = /etc/dovecot/dovecot-sql.conf.ext
}
userdb {
  driver = sql
  args = /etc/dovecot/dovecot-sql.conf.ext
}

# Solo IMAP cifrado (993); el 143 sin cifrar, apagado
service imap-login {
  inet_listener imap {
    port = 0
  }
  inet_listener imaps {
    port = 993
    ssl = yes
  }
}

# Entrega desde Postfix (LMTP): solo Postfix puede usar este socket
service lmtp {
  unix_listener /var/spool/postfix/private/dovecot-lmtp {
    mode = 0600
    user = postfix
    group = postfix
  }
}

# Comprobación de contraseñas para Postfix (SASL)
service auth {
  unix_listener /var/spool/postfix/private/auth {
    mode = 0660
    user = postfix
    group = postfix
  }
}
```

```
# Carpetas que se crean y se suscriben solas en cada buzón
namespace inbox {
  mailbox Drafts {
    auto = subscribe
  }
  mailbox Junk {
    auto = subscribe
  }
  mailbox Sent {
    auto = subscribe
  }
  mailbox Trash {
    auto = subscribe
  }
}
```

15.4 · Arrancar Dovecot y Postfix

EN EL SERVIDOR

```
sudo doveconf -n > /dev/null && echo "configuracion correcta"
sudo systemctl restart dovecot
sudo postfix check
sudo systemctl restart postfix
sudo ss -tlnp | grep -E ':(25|110|143|587|993|995) '
```

SALIDA ESPERADA

```
configuracion correcta
LISTEN 0  100  0.0.0.0:25   0.0.0.0:*  users:(("master",...))
LISTEN 0  100  0.0.0.0:587  0.0.0.0:*  users:(("master",...))
LISTEN 0  100  0.0.0.0:993  0.0.0.0:*  users:(("dovecot",...))
```

Comprueba antes de seguir

Dovecot y Postfix están `active`, y solo escuchan el 25, el 587 y el 993.

Paso 16 Episodio 15

Tu primer dominio y buzón

Por qué: el servidor está listo, pero aún no sabe de ningún dominio.

En PostfixAdmin (con la VPN activada):

1. **Lista de dominios** → **Nuevo dominio**: `TU_DOMINIO`. Deja marcado **Añadir alias por defecto** (crea `abuse@`, `postmaster@`...). Guarda.
2. **Lista de buzones** → **Añadir buzón**: usuario `TU_BUZON`, tu nombre y una contraseña larga (mínimo 12 caracteres). Guárdala en tu gestor.

EN EL SERVIDOR

```
sudo postmap -q TU_DOMINIO mysql:/etc/postfix/sql/dominios.cf
sudo postmap -q TU_BUZON@TU_DOMINIO mysql:/etc/postfix/sql/buzones.cf
sudo postmap -q postmaster@TU_DOMINIO mysql:/etc/postfix/sql/alias.cf
sudo doveadm user TU_BUZON@TU_DOMINIO
sudo doveadm auth test TU_BUZON@TU_DOMINIO
```

SALIDA ESPERADA

```
TU_DOMINIO
TU_DOMINIO/TU_BUZON/
TU_BUZON@TU_DOMINIO
field   value
uid     5000
gid     5000
...
Password:
passdb: TU_BUZON@TU_DOMINIO auth succeeded
```

Comprueba antes de seguir

Las tres consultas de `postmap` responden, `doveadm user` muestra `uid 5000` y `doveadm auth test` dice `auth succeeded`.

Paso 17 Episodio 15

Primeras pruebas: enviar y recibir

Por qué: probamos los tres caminos (enviar con usuario, recibir de fuera y que nadie pueda usar tu servidor para enviar spam) antes de configurar ningún móvil.

EN EL SERVIDOR

```
sudo apt install swaks
```

17.1 · Enviar con tu buzón por el 587

EN EL SERVIDOR

```
swaks --server mail.TU_DOMINIO --port 587 --tls \
  --auth LOGIN --auth-user TU_BUZON@TU_DOMINIO \
  --from TU_BUZON@TU_DOMINIO --to TU_CORREO_EXTERNO \
  --header "Subject: Prueba desde mi servidor" \
  --body "Hola. Este correo sale de mi propio servidor."
```

Mira en TU_CORREO_EXTERNO: puede que llegue a **spam**. Es normal todavía: falta DKIM y DMARC (parte C). Lo importante ahora es que llega.

17.2 · Recibir

Desde TU_CORREO_EXTERNO, responde a ese correo. Después:

EN EL SERVIDOR

```
sudo grep -E "status=sent|lmtp" /var/log/mail.log | tail -n 5
sudo doveadm mailbox status -u TU_BUZON@TU_DOMINIO messages INBOX
```

SALIDA ESPERADA

```
... postfix/lmtp[...]: ...: to=<TU_BUZON@TU_DOMINIO>, relay=mail.TU_DOMINIO[private/dovecot-
lmtp], ... status=sent (250 2.0.0 <TU_BUZON@TU_DOMINIO> ... Saved)
INBOX messages=1
```

17.3 · Que nadie pueda usar tu servidor para enviar spam

EN EL SERVIDOR

```
swaks --server mail.TU_DOMINIO --port 25 --from alguien@ejemplo.com --to TU_CORREO_EXTERNO
swaks --server mail.TU_DOMINIO --port 25 --quit-after EHLO
```

Comprueba antes de seguir

- El correo de prueba llega a tu correo externo (aunque sea a spam).
- La respuesta llega a tu buzón: `INBOX messages=1`.
- El 25 responde **Relay access denied** a un extraño y no ofrece `AUTH`.

PARTE C

DKIM, DMARC y el 10/10

Firmamos cada correo con DKIM, publicamos la política DMARC y lo comprobamos con mail-tester y con Gmail. Corresponde al **episodio 16**.

Paso 18 Episodio 16

OpenDKIM: instalar y crear la clave

Por qué: DKIM funciona con un par de claves: la **privada** se queda en el servidor y firma cada correo; la **pública** se publica en el DNS para que cualquiera compruebe la firma.

EN EL SERVIDOR

```
sudo apt install opendkim opendkim-tools
sudo mkdir -p /etc/opendkim/keys/TU_DOMINIO
sudo opendkim-genkey -b 2048 -d TU_DOMINIO -D /etc/opendkim/keys/TU_DOMINIO -s dkim2026 -v
sudo chown -R opendkim:opendkim /etc/opendkim
sudo chmod 700 /etc/opendkim/keys
sudo ls -l /etc/opendkim/keys/TU_DOMINIO/
```

SALIDA ESPERADA

```
-rw------- 1 opendkim opendkim 1679 ... dkim2026.private
-rw------- 1 opendkim opendkim  510 ... dkim2026.txt
```

Comprueba antes de seguir

Existen `dkim2026.private` y `dkim2026.txt`, propiedad de `opendkim`.

Paso 19 Episodio 16

OpenDKIM: tablas y conexión con Postfix

Por qué: tres tablas pequeñas le dicen a OpenDKIM qué clave usar para cada remitente y en quién confiar.

19.1 · Las tablas

EN EL SERVIDOR

```
echo "dkim2026._domainkey.TU_DOMINIO
TU_DOMINIO:dkim2026:/etc/opendkim/keys/TU_DOMINIO/dkim2026.private" | sudo tee
/etc/opendkim/key.table
echo "*@TU_DOMINIO dkim2026._domainkey.TU_DOMINIO" | sudo tee /etc/opendkim/signing.table
printf '127.0.0.1\nlocalhost\n' | sudo tee /etc/opendkim/trusted.hosts
sudo chown opendkim:opendkim /etc/opendkim/*.table /etc/opendkim/trusted.hosts
```

19.2 · La configuración de OpenDKIM

Sustituye el archivo entero: `sudo cp /etc/opendkim.conf /etc/opendkim.conf.orig` y después `sudo nano /etc/opendkim.conf`, borra su contenido y pega:

CONTENIDO DEL ARCHIVO · /ETC/OPENDKIM.CONF

```
Syslog                  yes
SyslogSuccess           yes
UMask                   007
UserID                  opendkim
PidFile                 /run/opendkim/opendkim.pid
Socket                  inet:8891@127.0.0.1

Mode                    sv
MTA                     ORIGINATING
Canonicalization        relaxed/simple
OversignHeaders         From

KeyTable                refile:/etc/opendkim/key.table
SigningTable            refile:/etc/opendkim/signing.table
InternalHosts           /etc/opendkim/trusted.hosts

TrustAnchorFile         /usr/share/dns/root.key
```

19.3 · Conectar Postfix con OpenDKIM

EN EL SERVIDOR

```
sudo systemctl restart opendkim
sudo ss -tlnp | grep 8891
sudo postconf -e \
  'smtpd_milters = inet:127.0.0.1:8891' \
  'non_smtpd_milters = $smtpd_milters' \
  'milter_default_action = accept'
sudo systemctl reload postfix
```

Comprueba antes de seguir

OpenDKIM escucha en `127.0.0.1:8891` y `sudo postconf smtpd_milters` muestra `inet:127.0.0.1:8891`.

Paso 20 Episodio 16

Publicar la clave DKIM

Por qué: la clave pública va en un registro TXT con un nombre especial.

EN EL SERVIDOR

```
sudo grep -o '"[^"]*"' /etc/opendkim/keys/TU_DOMINIO/dkim2026.txt | tr -d '"\n'; echo
```

SALIDA ESPERADA

```
v=DKIM1; h=sha256; k=rsa; p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA...IDAQAB
```

En el DNS de IONOS, crea un registro **TXT**: nombre de host `dkim2026._domainkey`, valor la línea completa que empieza por `v=DKIM1`. Espera unos minutos y comprueba:

EN EL SERVIDOR

```
sudo opendkim-testkey -d TU_DOMINIO -s dkim2026 -vvv
```

SALIDA ESPERADA

```
opendkim-testkey: using default configfile /etc/opendkim.conf
opendkim-testkey: checking key 'dkim2026._domainkey.TU_DOMINIO'
opendkim-testkey: key not secure
opendkim-testkey: key OK
```

Comprueba antes de seguir

`opendkim-testkey` termina con `key OK`.

Paso 21 Episodio 16

DMARC

Por qué: DMARC une SPF y DKIM: le dice al mundo qué hacer con un correo que dice ser tuyo pero no pasa las comprobaciones, y te pide informes de lo que pasa con tu dominio.

En el DNS de IONOS, un registro **TXT** con nombre de host `_dmarc` y valor:

PANEL DE IONOS · TXT _DMARC

```
v=DMARC1; p=none; rua=mailto:postmaster@TU_DOMINIO; fo=1
```

Cómo endurecer DMARC con el tiempo

Cuándo	Política	Qué pasa con un correo falso
Ahora	`p=none`	Se entrega, y lo ves en los informes
Tras 2–4 semanas de informes sin sorpresas	`p=quarantine`	Va a spam
Cuando todo lo legítimo pase siempre	`p=reject`	Se rechaza: nadie puede suplantar tu dominio

Lee los informes sin abrir XML

Los informes son archivos comprimidos con XML. Para entenderlos, súbelos a un analizador gratuito (busca «DMARC report analyzer») o simplemente comprueba que en todos tu IP aparece con `dkim=pass` y `spf=pass`.

EN TU PC · POWERSHELL

```
Resolve-DnsName _dmarc.TU_DOMINIO -Type TXT -Server 1.1.1.1
```

Comprueba antes de seguir

El registro `_dmarc` responde con `v=DMARC1; p=none; ...`.

Paso 22 Episodio 16

La prueba de verdad: mail-tester y Gmail

Por qué: mail-tester revisa tu correo como lo haría un filtro de spam (DNS, firma, listas negras, contenido) y le pone nota.

1. Abre `https://www.mail-tester.com` en el navegador. Te da una dirección del estilo `test-xxxxx@srv1.mail-tester.com` (cópiala, cambia en cada visita).
2. Envíale un correo de verdad, con asunto y texto normal (un correo vacío o de una palabra baja la nota):

```
EN EL SERVIDOR
```

```
swaks --server mail.TU_DOMINIO --port 587 --tls \
  --auth LOGIN --auth-user TU_BUZON@TU_DOMINIO \
  --from TU_BUZON@TU_DOMINIO --to test-xxxxx@srv1.mail-tester.com \
  --header "Subject: Comprobación de mi servidor de correo" \
  --body "Hola. Estoy comprobando que mi servidor de correo está bien configurado: SPF, DKIM y
DMARC. Un saludo."
```

1. Vuelve a mail-tester y pulsa **Comprobar tu puntuación**.

Apartado de mail-tester	Qué debe decir
Autenticación	SPF, DKIM y DMARC correctos (en verde)
DNS inverso	Tu IP resuelve a `mail.TU_DOMINIO` y coincide
Listas negras	No apareces en ninguna (o solo en alguna menor que no afecta a la nota)

En **Gmail**, abre el correo de prueba que le enviaste en el paso 17 (o envía otro) → menú de los tres puntos → **Mostrar original**:

```
SALIDA ESPERADA
```

```
SPF:   PASS con la IP TU_IP
DKIM:  'PASS' con el dominio TU_DOMINIO
DMARC: 'PASS'
```

mail-tester tiene un límite
La versión gratuita permite 3 pruebas al día. Si no sale 10/10, lee qué apartado resta puntos, corrígelo y vuelve a probar al día siguiente si hace falta.

Comprueba antes de seguir

- mail-tester da **10/10** (o 9/10 por algún detalle del contenido, no de la configuración).
- Gmail muestra **PASS** en SPF, DKIM y DMARC.
- `sudo grep "DKIM-Signature field added" /var/log/mail.log | tail -n 2` muestra que OpenDKIM ha firmado.

Paso 23 Episodio 16

Comprobación final e instantánea

Por qué: repaso de todo lo montado, un reinicio para comprobar que arranca solo, y la instantánea.

EN EL SERVIDOR

```
for s in mariadb php8.3-fpm postfix dovecot opendkim; do printf "%-12s %s\n" $s $(systemctl is-active $s); done
sudo ss -tlnp | grep -E ':(25|110|143|465|587|993|995|3306|8891) '
sudo postconf smtpd_tls_auth_only smtpd_sasl_auth_enable
sudo doveconf disable_plaintext_auth ssl
mailq | tail -n 1
```

Comprobación	Resultado correcto
Servicios	Los cinco `active`
Puertos públicos	Solo **25**, **587** y **993**; **3306** y **8891** solo en `127.0.0.1`; nada en 110, 143, 465 ni 995
Postfix	`smtpd_tls_auth_only = yes` y `smtpd_sasl_auth_enable = no` (solo se activa en el 587)
Dovecot	`disable_plaintext_auth = yes` y `ssl = required`
Cola de correo	`Mail queue is empty`
Panel	Con VPN se abre; sin VPN, 403; `setup.php`, 403 siempre

Reinicia el servidor y, al volver, envía otro correo con `swaks` (paso 17.1) y responde desde fuera. Después, en IONOS, crea la instantánea `08-correo`.

Qué copiar en tus copias de seguridad

- Los correos: `/var/mail/vhosts`.
- La base de datos: `sudo mariadb-dump --single-transaction mailserver | gzip > mailserver.sql.gz`.
- La configuración: `/etc/postfix`, `/etc/dovecot`, `/etc/opendkim`, `/etc/opendkim.conf` y `/var/www/postfixadmin/config.local.php`. La clave DKIM es imprescindible: sin ella, tendrías que publicar una nueva.
- Lo automatizaremos en la temporada de copias de seguridad.

Comprueba antes de seguir

Tienes tu propio correo con tu dominio, cifrado de punta a punta, que nadie puede usar para enviar spam y que Gmail reconoce como legítimo. En la Guía 09: el webmail, tus programas de correo, el filtro de spam y fail2ban.

ANEXOS

Para cuando algo no sale

El registro del correo lo cuenta casi todo: `sudo tail -f /var/log/mail.log` mientras repites la prueba (Ctrl+C para salir). Postfix, Dovecot y OpenDKIM escriben ahí.

Anexo A · Problemas frecuentes

Lo que ves	Qué pasa	Qué hacer
`warning: do not list domain TU_DOMINIO in BOTH mydestination and virtual_mailbox_domains`	El dominio está en los dos sitios	`sudo postconf -e 'mydestination = localhost'` y `sudo systemctl reload postfix`
`postmap -q` no responde nada	La consulta no encuentra el dato, o no puede conectar con MariaDB	Comprueba el dominio o buzón en el panel (activo). Prueba la contraseña: `mariadb -u mailuser -p -h 127.0.0.1 mailserver`
`doveadm auth test`: `auth failed`	Contraseña incorrecta, buzón inactivo o formato de contraseña distinto	Cambia la contraseña del buzón en el panel y repite. Revisa `default_pass_scheme = BLF-CRYPT` y `encrypt = 'php_crypt:BLOWFISH'`
`status=deferred (... Connection timed out)` al enviar a Gmail	El puerto 25 de salida está bloqueado	Paso 2: pide a IONOS que lo desbloqueen
`status=bounced (... 550 5.7.1 ... PTR ... )`	Tu IP no tiene DNS inverso o no coincide	Guía 01, paso 5
`lmtp ... Permission denied` o `No such file` con `private/dovecot-lmtp`	Dovecot no ha creado el socket o con otro dueño	Revisa el bloque `service lmtp` de `local.conf` y `sudo systemctl restart dovecot`
`SASL authentication failure` en el 587	Usuario o contraseña incorrectos (el usuario es la dirección completa)	Prueba con `doveadm auth test`. Si allí funciona, revisa lo que escribiste en `swaks`
El móvil o swaks: error de certificado	Te conectas con un nombre distinto de `mail.TU_DOMINIO`, o el certificado caducó	Usa siempre `mail.TU_DOMINIO`. `sudo certbot certificates` para ver la fecha
`ss` no muestra el 8891 de OpenDKIM	El servicio arranca con el socket de `/etc/default/opendkim` en vez del de `opendkim.conf`	En `/etc/default/opendkim`, pon `SOCKET=inet:8891@127.0.0.1`; luego `sudo /lib/opendkim/opendkim.service.generate` (regenera el servicio con ese ajuste), `sudo systemctl daemon-reload` y `sudo systemctl restart opendkim`
Gmail: `DKIM: FAIL` o sin DKIM	OpenDKIM no firma o el TXT no coincide	`opendkim-testkey` (paso 20) y busca `DKIM-Signature field added` en `mail.log`
Gmail: `SPF: SOFTFAIL` o `FAIL`	Hay dos SPF, o el MX no apunta a tu servidor	Paso 3: un solo TXT `v=spf1` y el MX en `mail.TU_DOMINIO`

Lo que ves	Qué pasa	Qué hacer
El correo llega a spam con todo en PASS	Dominio o IP nuevos, sin reputación todavía	Normal las primeras semanas. Pide a quien te escriba que lo marque como «no es spam» y envía correo normal, sin envíos masivos

Anexo B · Puertos al terminar esta guía

Puerto	Servicio	Accesible desde	Qué lo protege
25	Postfix: correo entrante	Todo Internet (tiene que serlo)	No admite contraseñas; solo entrega a tus dominios (no es relé abierto)
587	Postfix: envío de tus usuarios	Todo Internet	Cifrado obligatorio + usuario y contraseña
993	Dovecot: leer el correo (IMAPS)	Todo Internet	Cifrado obligatorio + usuario y contraseña
443 → `postfixadmin.TU_DOMINIO`	PostfixAdmin (por Nginx)	Solo la VPN	`solo-vpn.conf` + usuario y contraseña; `setup.php` bloqueado
127.0.0.1:3306	MariaDB	Solo el propio servidor	Escucha solo en 127.0.0.1
127.0.0.1:8891	OpenDKIM	Solo el propio servidor	Escucha solo en 127.0.0.1

Abiertos en IONOS y UFW: **25, 587 y 993**. Cerrados a propósito: 110, 143, 465 y 995.

Anexo C · Chuleta del correo

Para…	Cómo
Ver el registro en directo	`sudo tail -f /var/log/mail.log`
Ver la cola de correo	`mailq` (vacía es lo normal)
Reintentar lo que está en cola	`sudo postqueue -f`
Tu configuración de Postfix	`sudo postconf -n` · servicios: `sudo postconf -M`
Tu configuración de Dovecot	`sudo doveconf -n`
Probar una consulta de Postfix	`sudo postmap -q dato mysql:/etc/postfix/sql/archivo.cf`
Probar la contraseña de un buzón	`sudo doveadm auth test buzon@TU_DOMINIO`
Cuántos correos tiene un buzón	`sudo doveadm mailbox status -u buzon@TU_DOMINIO messages INBOX`
Probar la clave DKIM	`sudo opendkim-testkey -d TU_DOMINIO -s dkim2026 -vvv`
Enviar un correo de prueba	`swaks` (paso 17.1)
Dar correo a otro dominio (un cliente)	MX, SPF y DMARC en su DNS; dominio y buzones en PostfixAdmin; su clave DKIM (pasos 18–20) y una línea más en `key.table` y `signing.table`

Si te atascas

Pega: las últimas 30 líneas de `/var/log/mail.log` justo después de repetir la prueba, la salida de `sudo postconf -n` y la de `sudo doveconf -n`. **Nunca** pegues los archivos de `/etc/postfix/sql/` ni `dovecot-sql.conf.ext`: llevan la contraseña de la base de datos.