Práctica A6.D1: Servicio de correo electrónico

En esta práctica vamos a aprender a instalar y a configurar un servicio de correo electrónico en un sistema Linux sobre una instancia de AWS.

El contenido de los archivos *hostname* y *hosts* es muy importante. Antes de comenzar prestemos atención a la forma de cambiar el hostname en una máquina AWS tal y como indica la documentación de AWS.

Para que la actualización del nombre de host persista, Se debe verificar que la configuración *preserve_hostname* cloud-init esté establecida en *true*. Modifica el siguiente fichero:

```
sudo nano /etc/cloud/cloud.cfg
```

Si la configuración *preserve_hostname* no aparece en la lista, agregue la siguiente línea de texto al final del archivo:

```
preserve_hostname: true
```

Ejecute el comando

```
sudo hostnamectl set-hostname mail
```

y esto cambiará el archivo *hostname* al valor indicado; el nombre de tu sistema (sin dominio)

El archivo */etc/hosts* debería quedar de la forma:

```
  GNU nano 6.2
127.0.0.1 localhost
127.0.1.1 mail.mlls.line.pm mail

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

En este caso el nombre corto de mi equipo es *mail* y el dominio para el que estamos instalando nuestro correo es *mlls.line.pm*, es decir, las cuentas de nuestro correo serán del tipo *usuario@mlls.line.pm*

En la introducción al servicio de correo ya instalamos Postfix y probamos a enviar correo desde el mismo servidor con la herramienta telnet y con mail.

Como **Amazon bloquea la salida por el puerto 25**, desde nuestro servidor **NO resulta posible enviar correos hacia el exterior** (gmail, yahoo, etc). En cualquier caso, deberíamos revisar las técnicas propuestas para intentar que tu servidor de correo sea considerado fiable: SPF, DKIM y DMARC. En uno de los puntos de esta práctica se os pedirá que implementéis estas técnicas en vuestro servidor de correo y en vuestro dominio dinámico de dnsexit.

A tener en cuenta:

- El **servidor de correo será instalado en una instancia EC2 de tipo small y con un disco de 16 GB**. Se le debe **asociar una IP elástica** que luego se utilizará para la creación de los registros en dnsexit.
- Antes de crear la instancia sería recomendable **asignarle un Grupo de Seguridad** que permita el tráfico entrante a los puertos TCP siguientes desde cualquier dirección IP (no es necesario incluir todos los puertos ahora, sólo conforme los vayamos usando):

  25 (SMTP)

  80 (HTTP)

  110 (POP3)

  143 (IMAP)

  443 (HTTPS)

  465 (SMTPS)

  993 (IMAPS)

  995 (POP3S)

  El puerto 22 (SSH) debería abrirse pero, en la práctica, sólo para las IP usadas para la administración remota.

- El dominio de correo electrónico que vamos a utilizar será uno dinámico de DNSExit, del tipo **xxx.line.pm** o similar, pero teniendo en cuenta que **debe** incluir tus iniciales **xxx**.

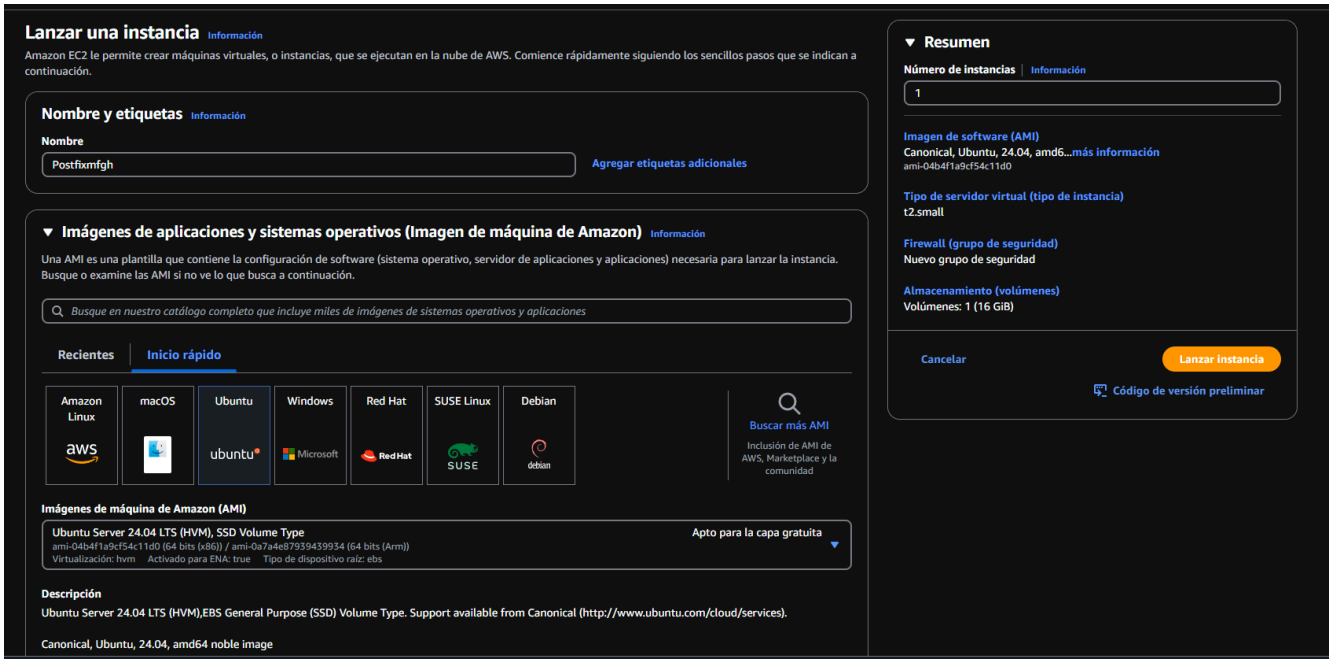
  El nombre corto del equipo será **mail** por lo que el nombre largo será **mail.xxx.line.pm**

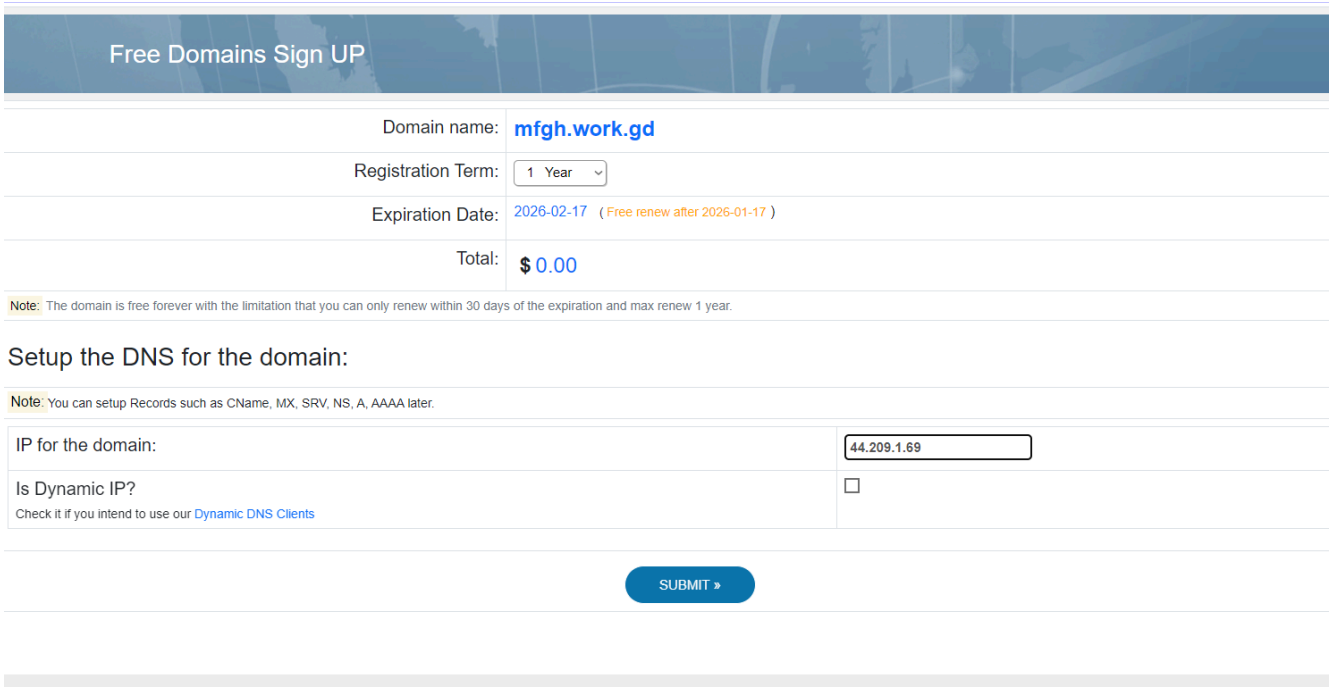
  Tus cuentas de correo serán del tipo **usuario@xxx.line.pm**

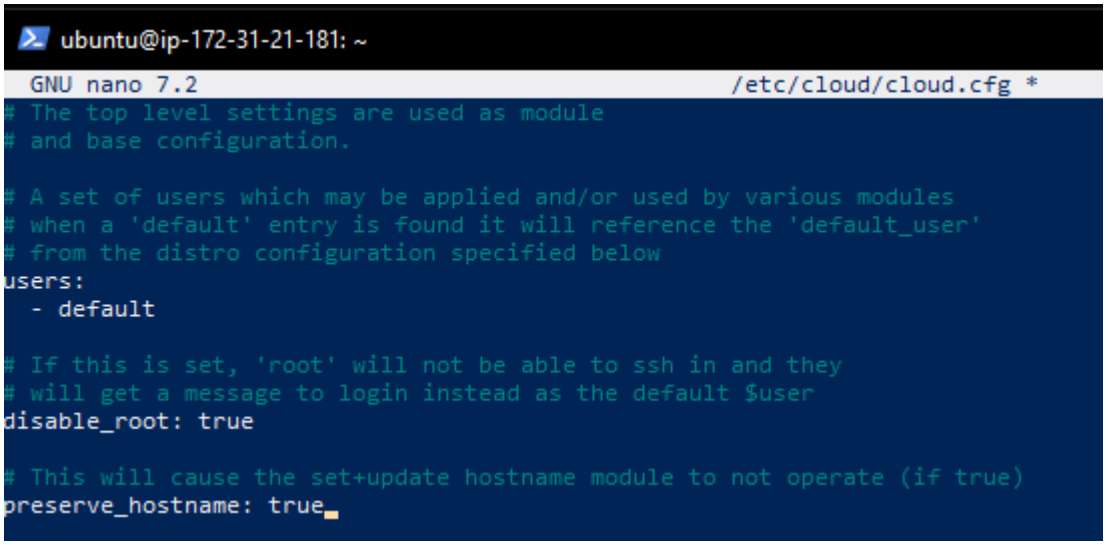
Existen numerosas referencias y tutoriales por internet en los que se intenta explicar, con más o menos éxito, el proceso de instalación de un servidor de correo electrónico funcional. En la lista de enlaces interesantes de esta práctica se encuentran varias de esas referencias. Las que he catalogado como **Referencia Base** son las que están probadas y usaremos para resolver los pasos de esta práctica.

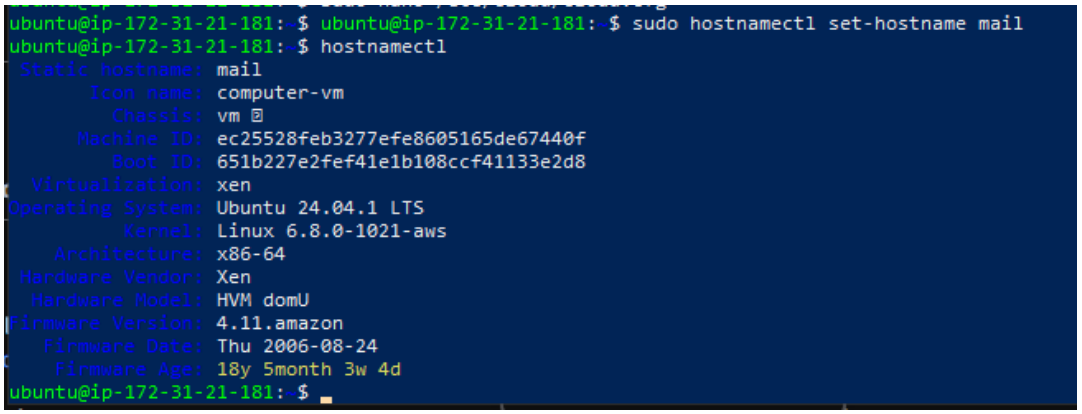
Nunca debemos olvidar que la fuente de información principal de cualquier servicio de red debe ser la página oficial de la herramienta o del protocolo en cuestión.

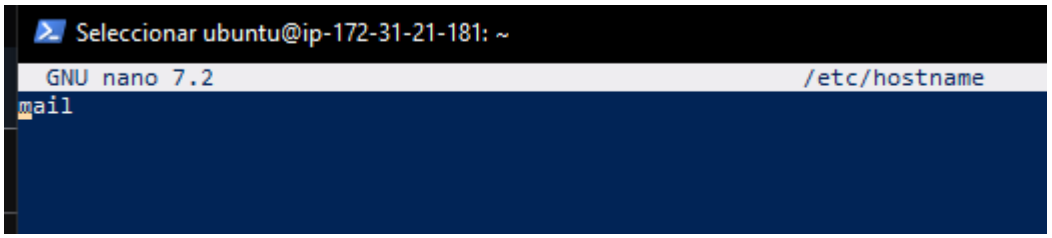
Los siguientes **pasos** deben realizarse sobre la instancia EC2 de AWS con AMI de Ubuntu.

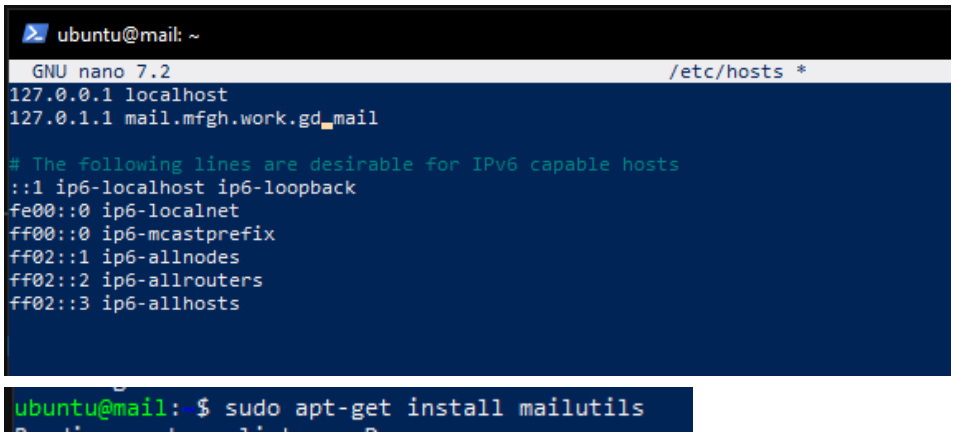






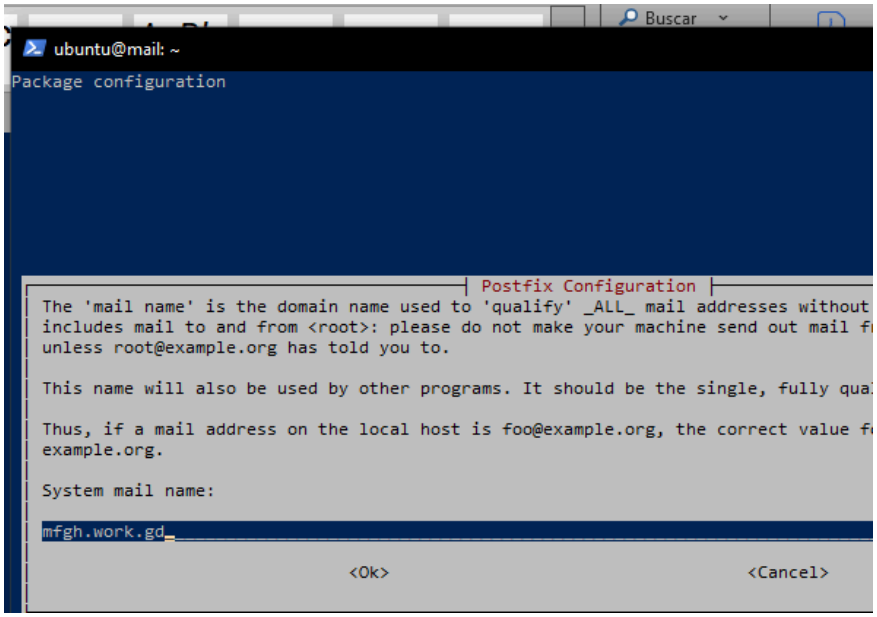


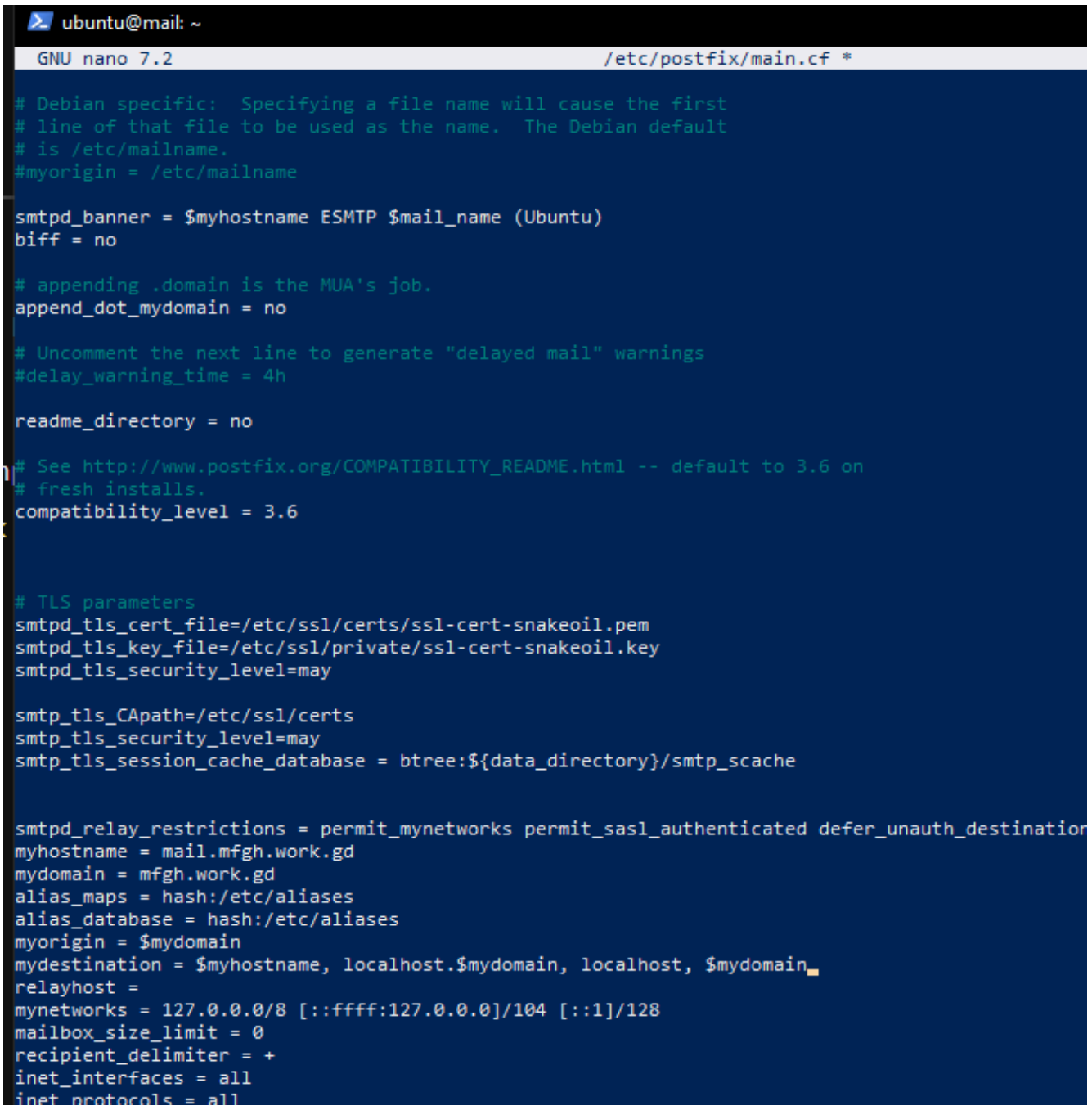


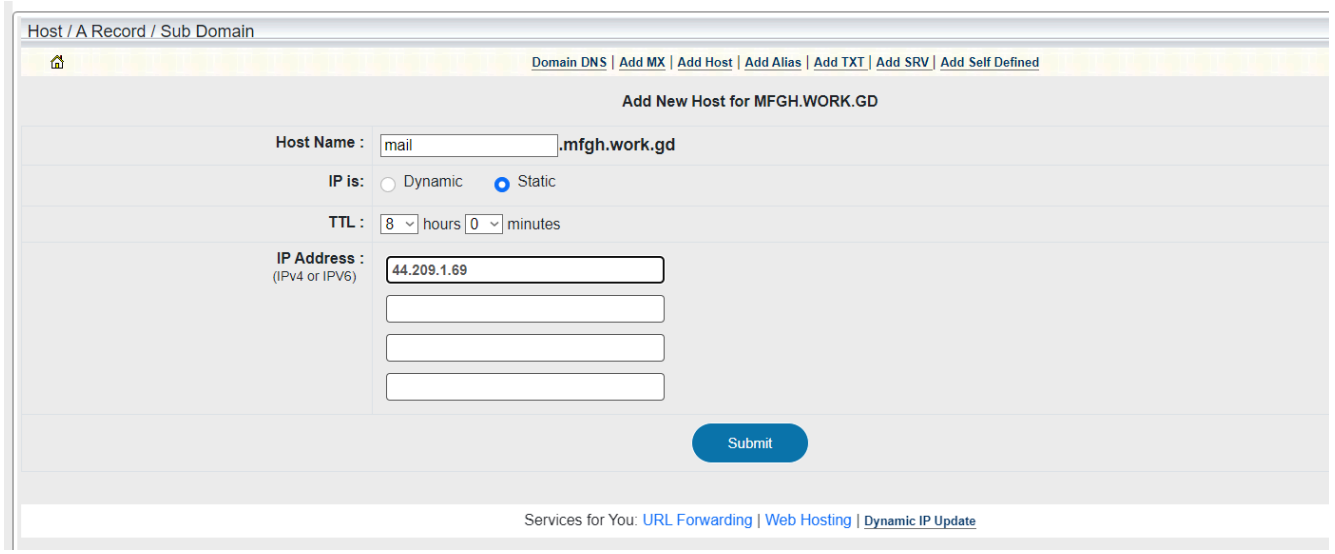


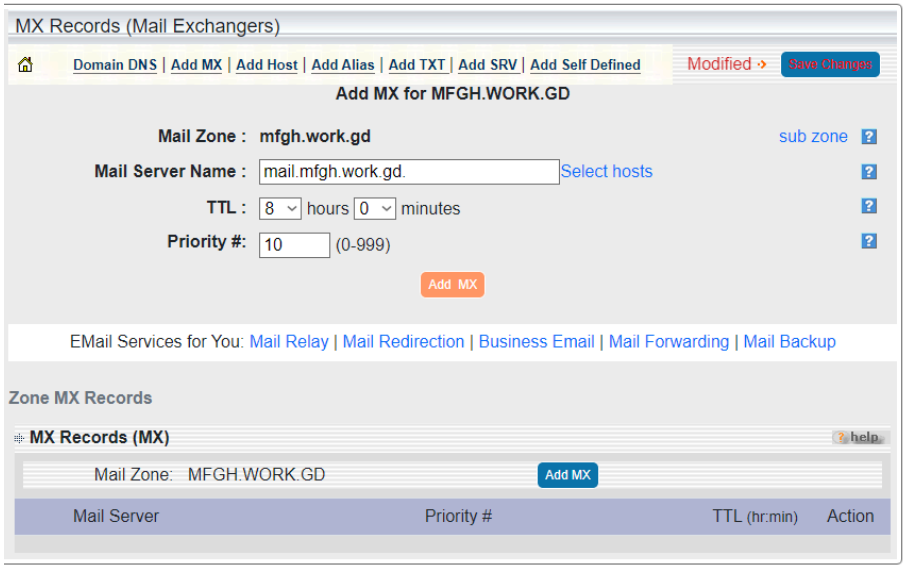
Paso 1: (1,5 ptos.)

Instalación del servidor Postfix, configuración básica y prueba de envío de correos en local, y recepción de correos desde un agente externo. Se debe configurar adecuadamente DNSexit, capturando pantallas del mismo.

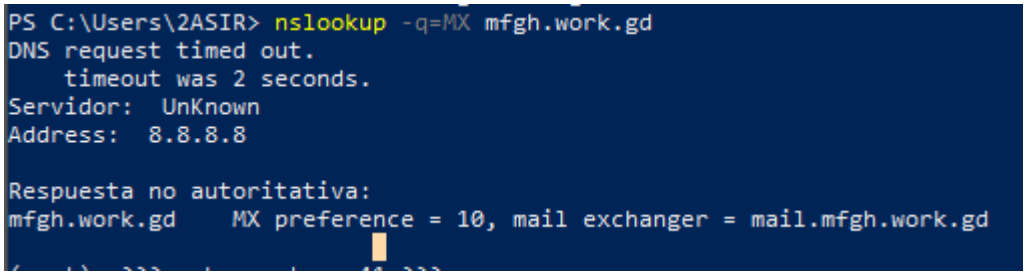

Buscar
ubuntu@mail: ~
Package configuration
Postfix Configuration
The 'mail name' is the domain name used to 'qualify' _ALL_ mail addresses without
includes mail to and from <root>: please do not make your machine send out mail f
unless root@example.org has told you to.
This name will also be used by other programs. It should be the single, fully qua
Thus, if a mail address on the local host is foo@example.org, the correct value f
example.org.
System mail name:
mfgh.work.gd
<Ok>
<Cancel>


ubuntu@mail: ~
GNU nano 7.2 /etc/postfix/main.cf *
Debian specific: Specifying a file name will cause the first
line of that file to be used as the name. The Debian default
is /etc/mailname.
#myorigin = /etc/mailname
smtpd_banner = $myhostname ESMTP $mail_name (Ubuntu)
biff = no
# appending .domain is the MUA's job.
append_dot_mydomain = no
# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h
readme_directory = no
# See http://www.postfix.org/COMPATIBILITY_README.html -- default to 3.6 on
# fresh installs.
compatibility_level = 3.6
# TLS parameters
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
smtpd_tls_security_level=may
smtp_tls_CApath=/etc/ssl/certs
smtp_tls_security_level=may
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache
smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_unauth_destinatior
myhostname = mail.mfgh.work.gd
mydomain = mfgh.work.gd
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = $mydomain
mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
relayhost =
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
inet_protocols = all

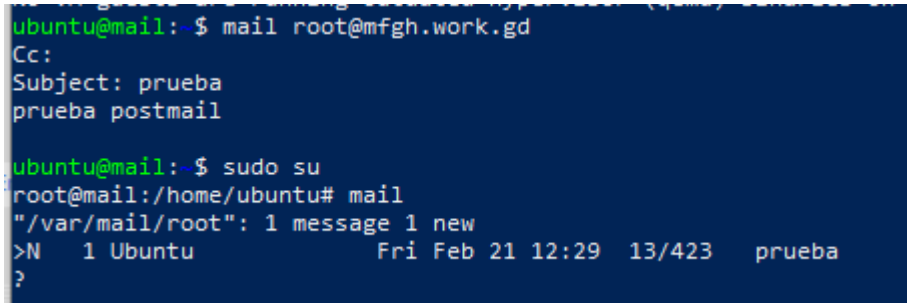


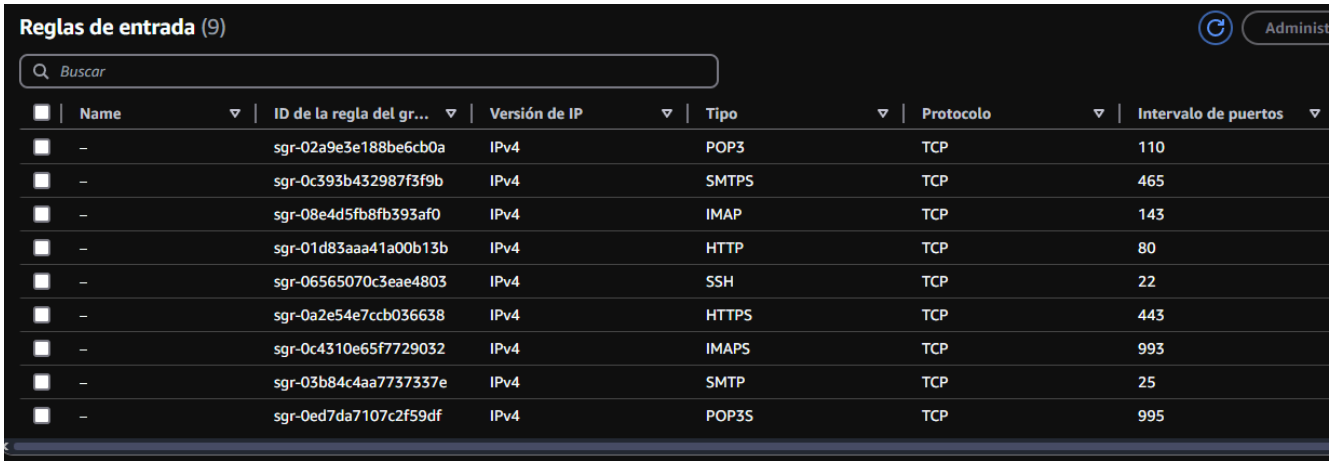


Vemos que se resuelve el registro MX.



Probamos que se envían correos entre sí y además conecta con el exterior, abriendo los puertos necesarios para la práctica.

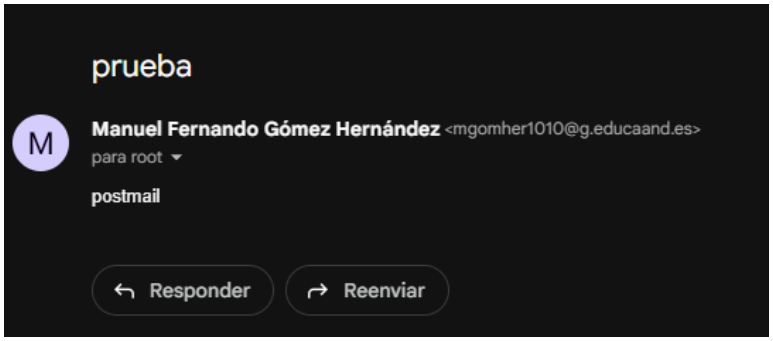


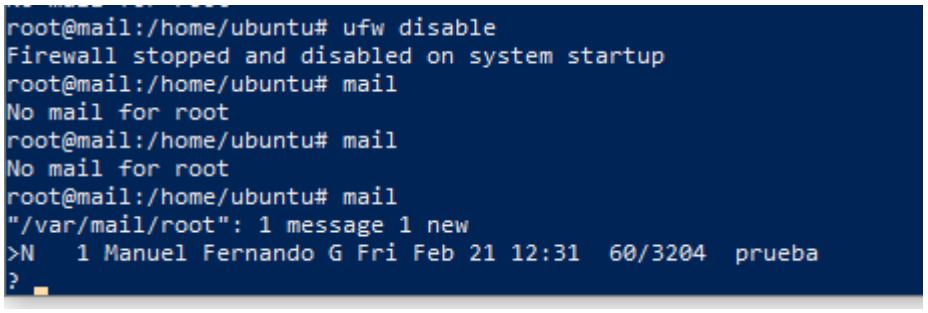


Reglas de entrada (9)

Buscar

Name	ID de la regla del gr...	Versión de IP	Tipo	Protocolo	Intervalo de puertos
–	sgr-02a9e3e188be6cb0a	IPv4	POP3	TCP	110
–	sgr-0c393b432987f3f9b	IPv4	SMTPS	TCP	465
–	sgr-08e4d5fb8fb393af0	IPv4	IMAP	TCP	143
–	sgr-01d83aaa41a00b13b	IPv4	HTTP	TCP	80
–	sgr-06565070c3eae4803	IPv4	SSH	TCP	22
–	sgr-0a2e54e7ccb036638	IPv4	HTTPS	TCP	443
–	sgr-0c4310e65f7729032	IPv4	IMAPS	TCP	993
–	sgr-03b84c4aa7737337e	IPv4	SMTP	TCP	25
–	sgr-0ed7da7107c2f59df	IPv4	POP3S	TCP	995





```
root@mail:/home/ubuntu# ufw disable
Firewall stopped and disabled on system startup
root@mail:/home/ubuntu# mail
No mail for root
root@mail:/home/ubuntu# mail
No mail for root
root@mail:/home/ubuntu# mail
"/var/mail/root": 1 message 1 new
>N   1 Manuel Fernando G Fri Feb 21 12:31  60/3204  prueba
?
```

Paso 2:

Instalación de Dovecot con cifrado TLS. Ahora ya podremos enviar y recibir correos desde una máquina distinta a nuestro servidor, utilizando un cliente de correo como Thunderbird.

Instalo certbot, apache y certbot para apache.

```
ubuntu@mail:~$ sudo apt install certbot
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  python3-acme python3-certbot python3-configargparse python3-icu python3-josepy python3-parsedatetime
  python3-rfc3339
Suggested packages:
  python-certbot-doc python3-certbot-apache python3-certbot-nginx python-acme-doc
The following NEW packages will be installed:
  certbot python3-acme python3-certbot python3-configargparse python3-icu python3-josepy python3-parsedatetime
  python3-rfc3339
0 upgraded, 8 newly installed, 0 to remove and 69 not upgraded.
Need to get 1031 kB of archives.
After this operation, 5323 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-josepy all 1.14.0-1 [22.1 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-rfc3339 all 1.1-4 [6744 B]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-acme all 2.9.0-1 [48.5 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-configargparse all 1.7-1 [31.
]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-parsedatetime all 2.6-3 [32.8
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-certbot all 2.9.0-1 [267 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 python3-icu amd64 2.12-1build2 [534 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 certbot all 2.9.0-1 [89.2 kB]
Fetched 1031 kB in 0s (23.7 MB/s)
Preconfiguring packages ...
Selecting previously unselected package python3-josepy.
(Reading database ...
```

```
ubuntu@mail:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

```
ubuntu@mail:~$ sudo apt install python3-certbot-apache
Reading package lists... Done
Building dependency tree... Done
```

Creo el archivo de configuración de apache y le añadimos contenido.

```
  GNU nano 7.2                          /etc/apache2/sites-available/sitiocorre.conf *
<VirtualHost *:80>
        ServerName mail.mfgh.work.gd

        DocumentRoot /var/www/html/
</VirtualHost>
```

Activamos este sitio y además reiniciamos apache.

```
ubuntu@mail:~$ ubuntu@mail:~$ sudo a2ensite sitiocorre.conf
Enabling site sitiocorre.
To activate the new configuration, you need to run:
  systemctl reload apache2
```

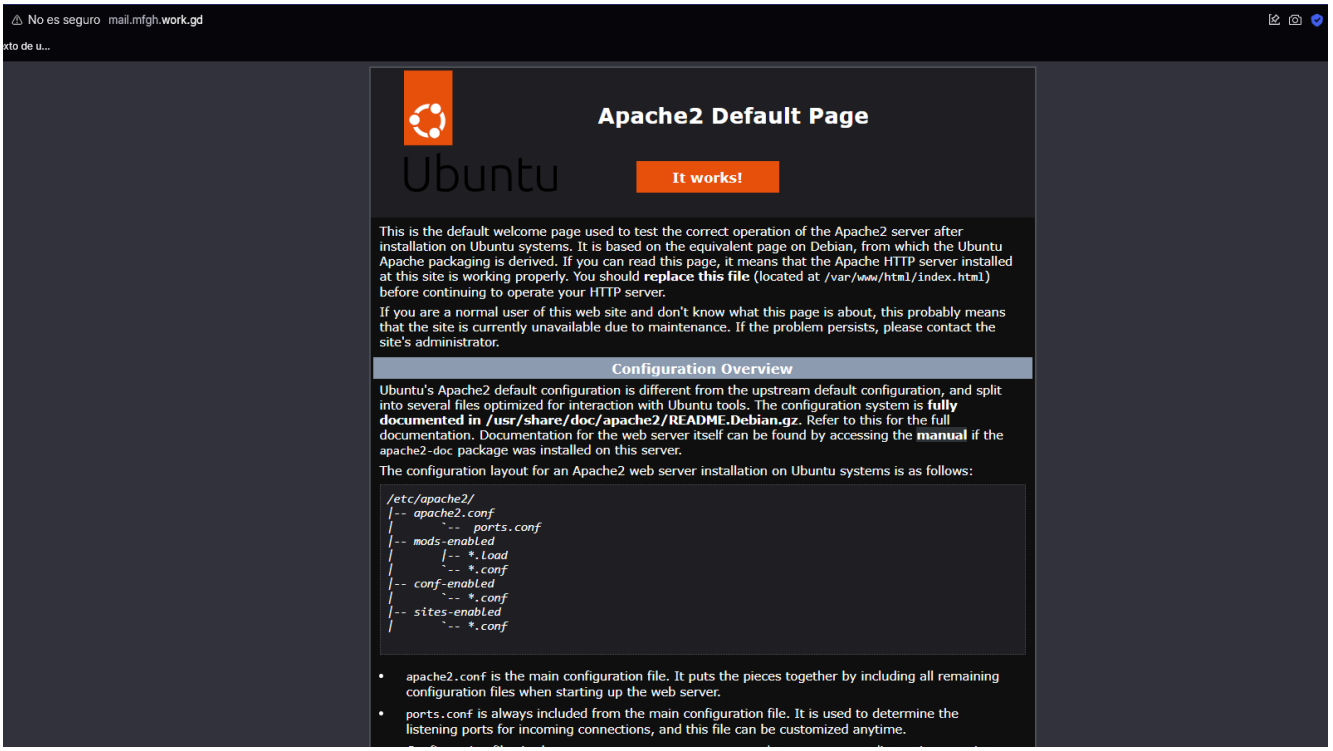
Desactivamos el sitio por defecto.

```
ubuntu@mail:~$ sudo a2dissite 000-default
Site 000-default disabled.
To activate the new configuration, you need to run:
  systemctl reload apache2
ubuntu@mail:~$ sudo systemctl reload apache2
ubuntu@mail:~$
```

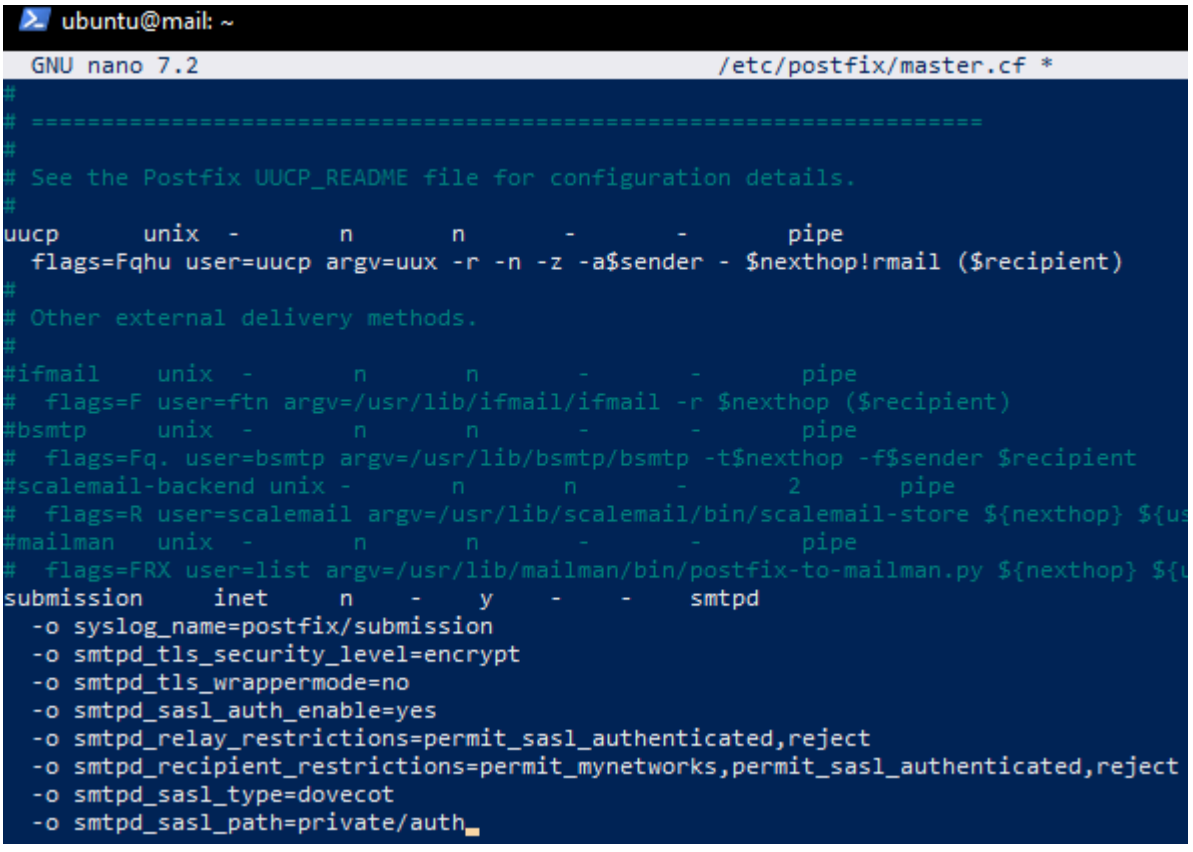
Creamos el certificado.

```
ubuntu@mail:~$ sudo systemctl reload apache2
ubuntu@mail:~$ sudo certbot certonly -a apache --agree-tos --no-eff-email --staple-ocsp --email root@mfgh.work.gd -d mail.mfgh.work.gd
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Account registered.
```

Podemos ver que el sitio funciona.



Añadimos esto en /etc/postfix/master.cf



Editamos las ultimas partes de las rutas de los certificados.

```
# TLS parameters
smtpd_tls_cert_file=/etc/letsencrypt/live/mail.mfgh.work.gd/fullchain.pem
smtpd_tls_key_file=/etc/letsencrypt/live/mail.mfgh.work.gd/privkey.pem
smtpd_tls_security_level=may
```

El servicio está activo

```
ubuntu@mail:~$ sudo ss -lnpt | grep master
LISTEN 0      100          0.0.0.0:587       0.0.0.0:*    users:(("master",pid=5643,fd=95))
LISTEN 0      100          0.0.0.0:25        0.0.0.0:*    users:(("master",pid=5643,fd=13))
LISTEN 0      100             [::]:587          [::]:*    users:(("master",pid=5643,fd=96))
LISTEN 0      100             [::]:25           [::]:*    users:(("master",pid=5643,fd=14))
ubuntu@mail:~$
```

Instalo dovecot.

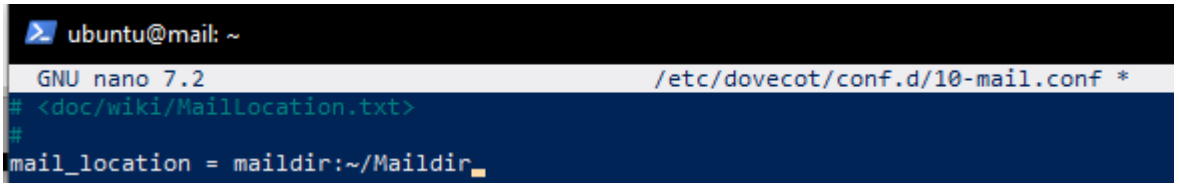
```
ubuntu@mail:~$ sudo apt install dovecot-core dovecot-imapd
```

```
ubuntu@mail:~$ sudo apt install dovecot-pop3d
Reading package lists... Done
```

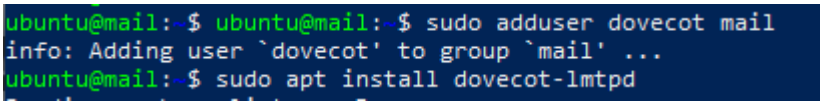
Entramos al fichero de configuración y descomentamos.

```
ubuntu@mail: ~
  GNU nano 7.2                                  /etc/dovecot/dovecot.conf *
# Enable installed protocols
protocols = imap
!include_try /usr/share/dovecot/protocols.d/*.protocol
```

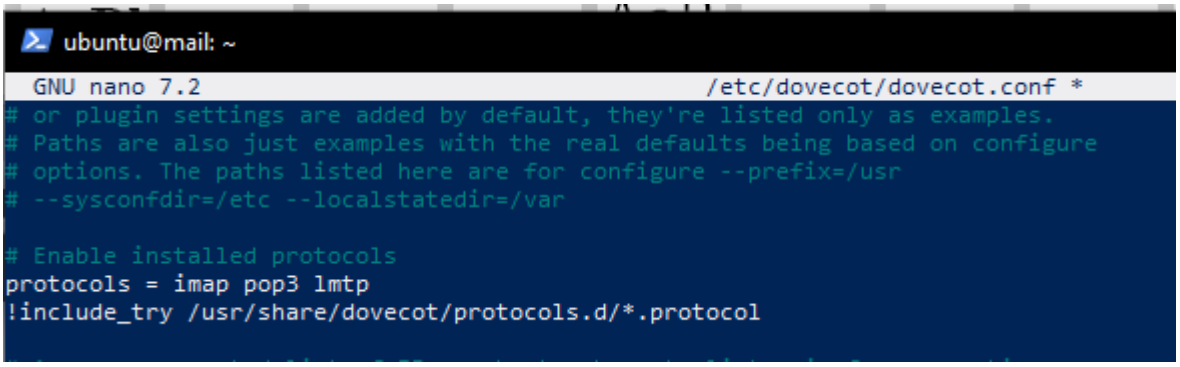
En este archivo de configuración modificamos este valor.



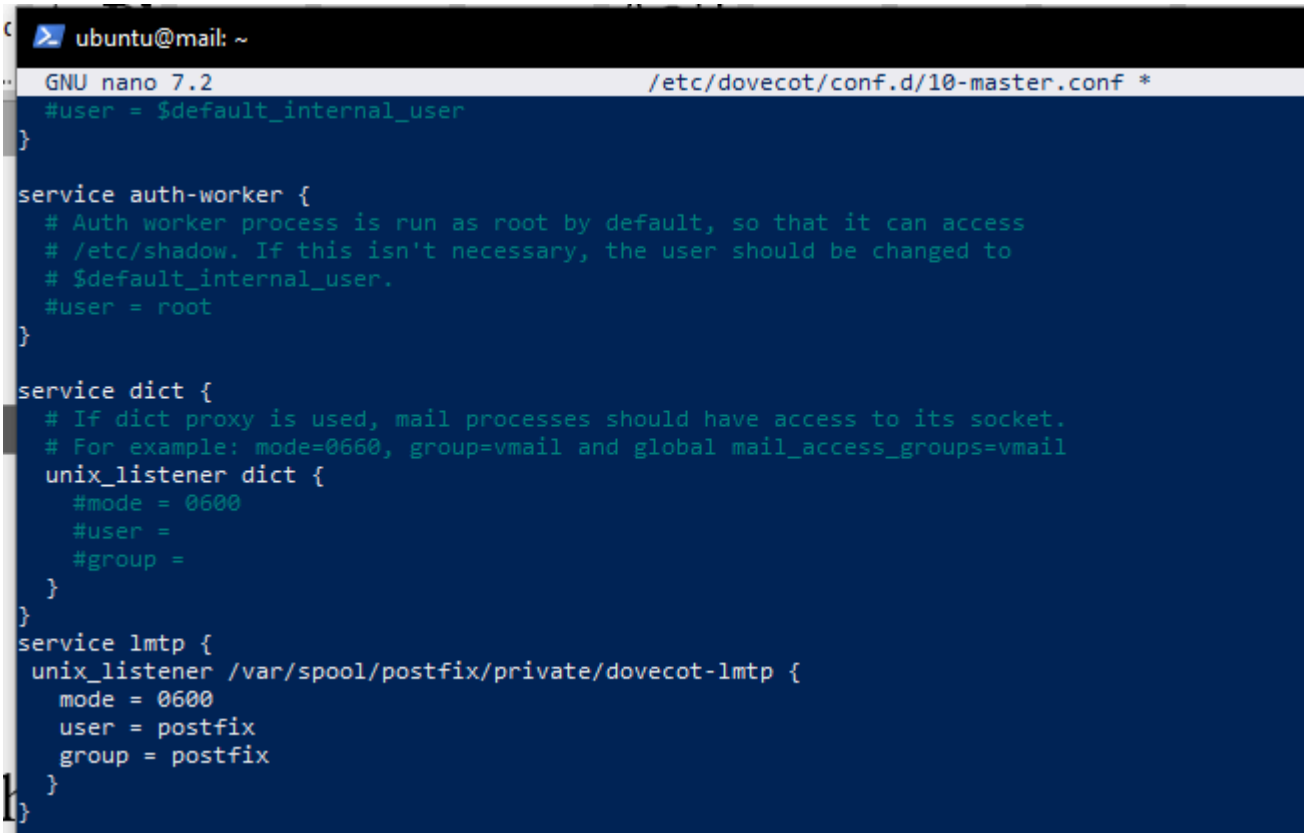
Añadimos el usuario correspondiente e instalamos lmtpd.



En dovecot.conf añadimos los protocolos.



Accedemos a /etc/dovecot/conf.d/10-master.conf y añadimos esto.



Accedemos a /etc/postfix/main.cf y añadimos esto.

```
inet_interfaces = all
inet_protocols = all
mailbox_transport = lmtp:unix:private/dovecot-lmtp
smtputf8_enable = no
```

Entramos a /etc/dovecot/conf.d/10-auth.conf y descomentamos esto.

```
ubuntu@mail: ~
 GNU nano 7.2                                 /etc/dovecot/conf.d/10-auth.conf *
##
## Authentication processes
##

# Disable LOGIN command and all other plaintext authentications unless
# SSL/TLS is used (LOGINDISABLED capability). Note that if the remote IP
# matches the local IP (ie. you're connecting from the same computer), the
# connection is considered secure and plaintext authentication is allowed.
# See also ssl=required setting.
disable_plaintext_auth = yes
```

Añadimos esto.

```
# Space separated list of wanted authentication mechanisms:
#   plain login digest-md5 cram-md5 ntlm rpa apop anonymous gssapi otp
#   gss-spnego
# NOTE: See also disable_plaintext_auth setting.
auth_mechanisms = plain login
```

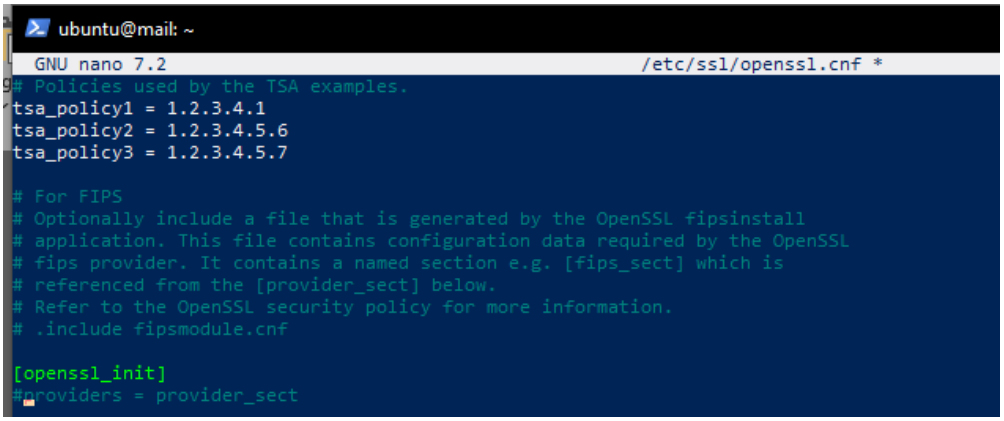
Vamos a 10-ssl.conf y cambiamos esto.

```
ubuntu@mail: ~
 GNU nano 7.2                                 /etc/dovecot/conf.d/10-ssl.conf *
##
## SSL settings
##

# SSL/TLS support: yes, no, required. <doc/wiki/SSL.txt>
ssl = required

# PEM encoded X.509 SSL/TLS certificate and private key. They're opened before
# dropping root privileges, so keep the key file unreadable by anyone but
# root. Included doc/mkcert.sh can be used to easily generate self-signed
# certificate, just make sure to update the domains in dovecot-openssl.cnf
ssl_cert = </etc/letsencrypt/live/mail.mfgh.work.gd/fullchain.pem
ssl_key = </etc/letsencrypt/live/mail.mfgh.work.gd/privkey.pem
```

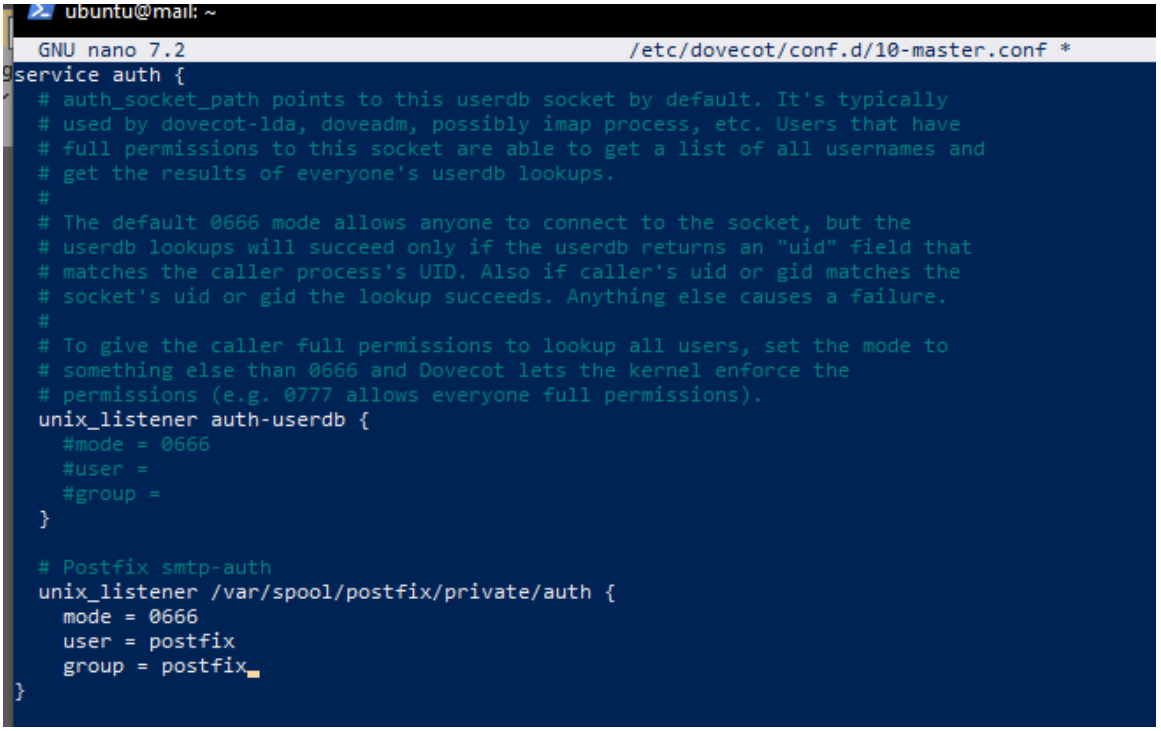
Descomentamos.

```
# version, and LATEST matches
ssl_min_protocol = TLSv1.2
```

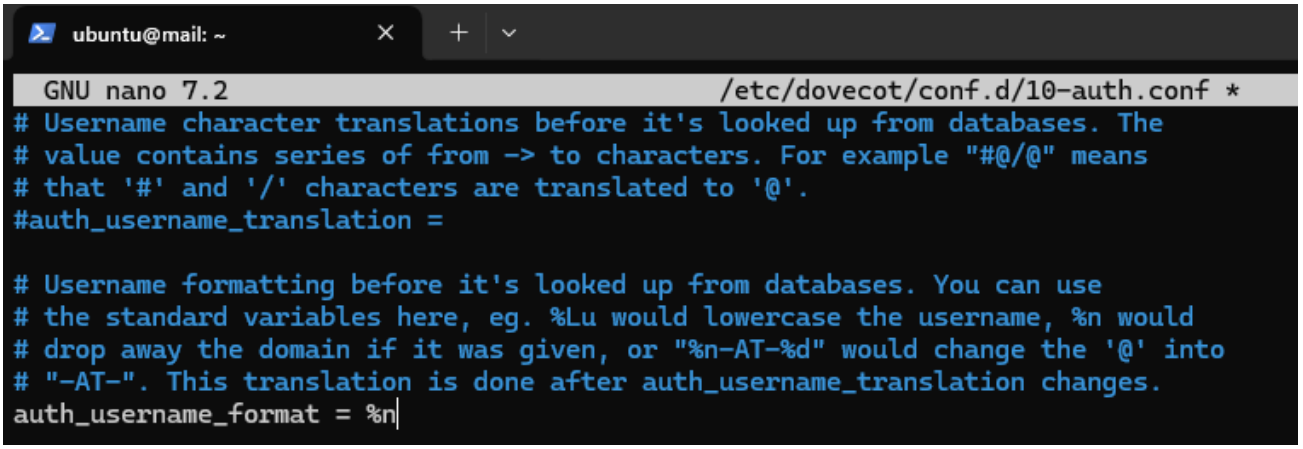
Vamos a /etc/ssl/openssl.cnf y comentamos esto.



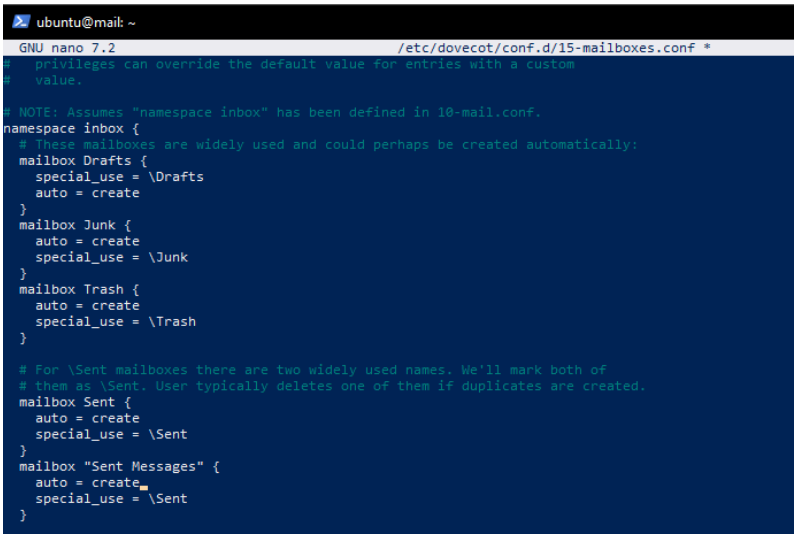
Vamos a /etc/dovecot/conf.d/10-master.conf y cambiamos para que tenga esto.



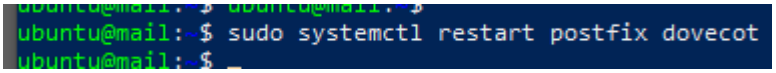
(Este cambio lo hice en mi casa por eso el cambio en el fondo) Descomento esa línea para que al enviar el correo busque el nombre sin el @ en el sistema.



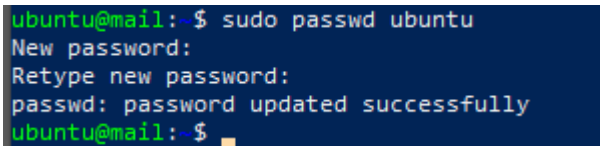
Entramos a /etc/dovecot/conf.d/15-mailboxes.conf y añadimos a todas las mailbox auto = create



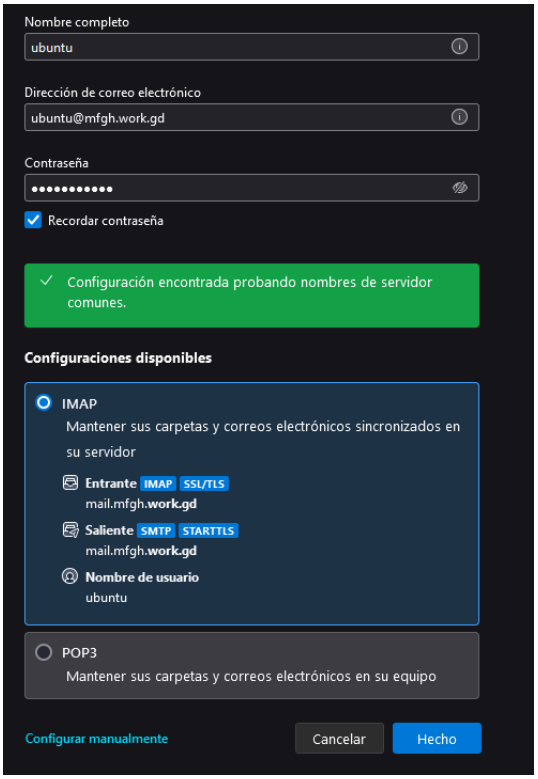
Reiniciamos dovecot.



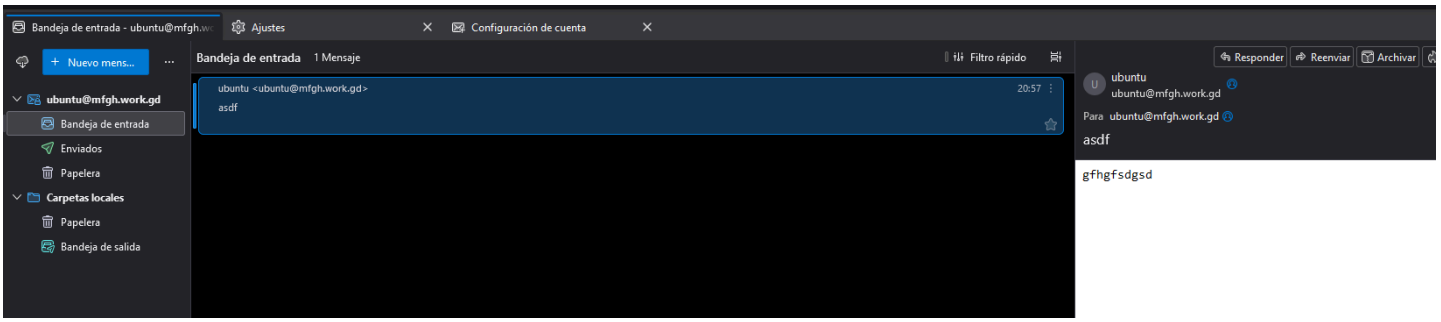
Para la prueba, vamos a darle una contraseña a Ubuntu.



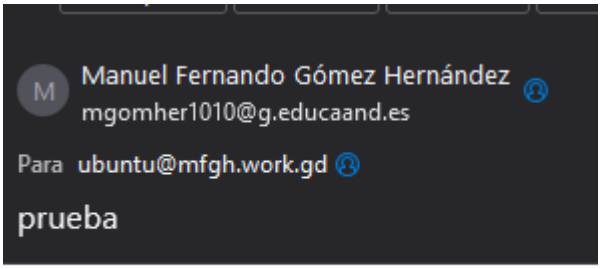
Instalamos thunderbird en la máquina Windows, y vemos que funciona.



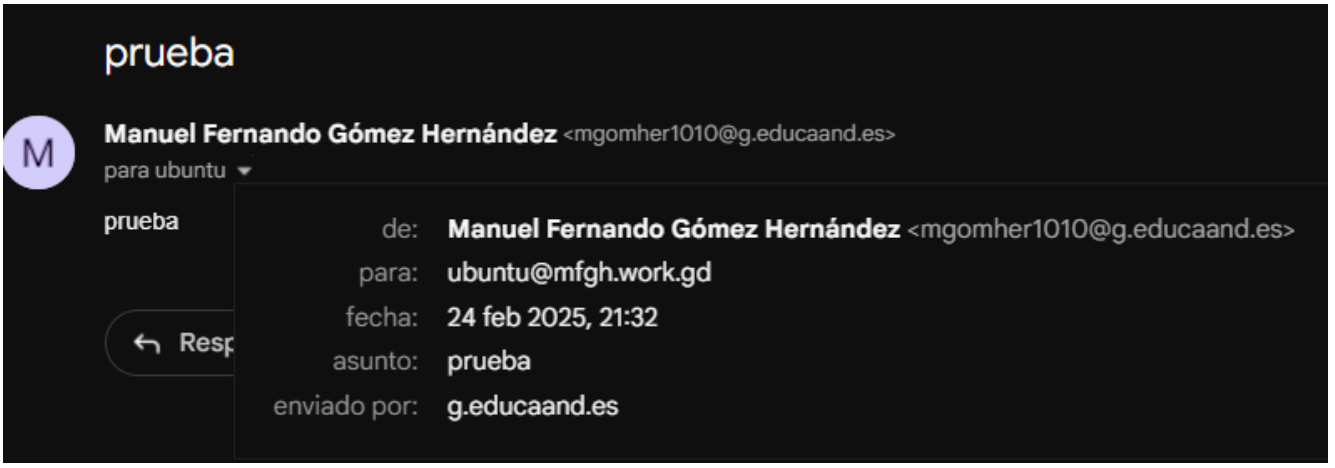
Vamos a enviar un email de prueba.



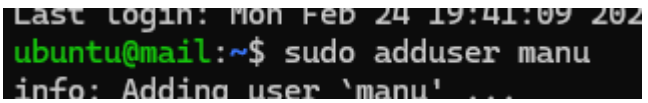
Ha funcionado, ahora, desde mi correo administrativo, voy a enviar un correo.



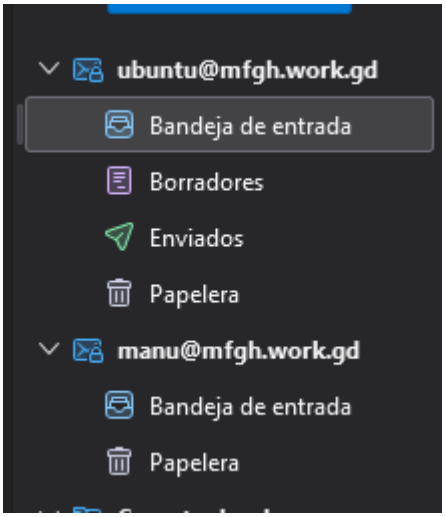
prueba



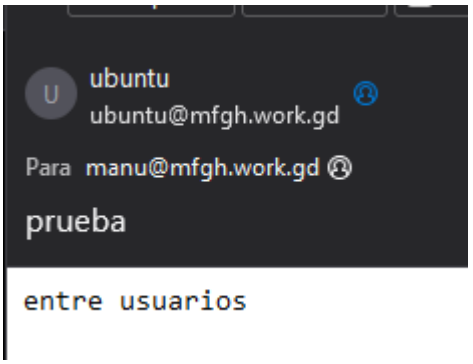
Creo otro usuario para las pruebas.

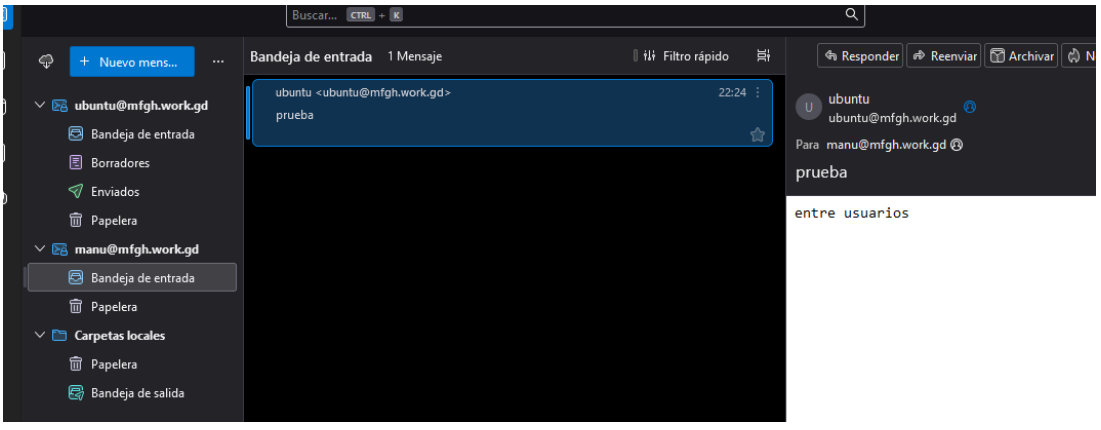


Tenemos las dos bandejas de entrada.



Envío correo de prueba.





Paso 3:

Creación de cuentas de correo virtuales utilizando MariaDB y PostfixAdmin. Configura tu dominio de correo xxx.line.pm, crea varios usuarios virtuales y prueba el envío y recepción de correo. Para enviar correos y evitar el bloqueo del puerto 25, puede utilizar el cliente de correo desde una máquina Windows Server en AWS.

```
Last login: Mon Feb 24 21:20:19 2025 from 213.94.3.8
ubuntu@mail:~$ sudo apt install mariadb-server mariadb-client
Reading package lists... Done
```

```
ubuntu@mail:~$ sudo systemctl status mariadb
● mariadb.service - MariaDB 10.11.8 database server
     Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-02-25 21:50:23 UTC; 25s ago
       Docs: man:mariadbd(8)
             https://mariadb.com/kb/en/library/systemd/
   Main PID: 2181 (mariadbd)
     Status: "Taking your SQL requests now..."
      Tasks: 12 (limit: 15435)
     Memory: 78.8M (peak: 81.7M)
        CPU: 333ms
     CGroup: /system.slice/mariadb.service
             └─2181 /usr/sbin/mariadbd
```

Hago la instalación segura.

```
ubuntu@mail:~$ sudo mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] n
 ... skipping.

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
 ... Success!


By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them.  This is intended only for testing, and to make the installation
go a bit smoother.  You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n]
 ... Success!

Normally, root should only be allowed to connect from 'localhost'.  This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n]
 ... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access.  This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n]
```

Descargamos postfixadmin.

```
ubuntu@mail:~$ wget https://github.com/postfixadmin/postfixadmin/archive/postfixadmin-3.3.11.tar.gz
--2025-02-25 21:54:52--  https://github.com/postfixadmin/postfixadmin/archive/postfixadmin-3.3.11.tar.gz
Resolving github.com (github.com)... 140.82.113.4
Connecting to github.com (github.com)|140.82.113.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/postfixadmin/postfixadmin/tar.gz/refs/tags/postfixadmin-3.3.11 [following]
--2025-02-25 21:54:52--  https://codeload.github.com/postfixadmin/postfixadmin/tar.gz/refs/tags/postfixadmin-3.3.11
Resolving codeload.github.com (codeload.github.com)... 140.82.113.10
Connecting to codeload.github.com (codeload.github.com)|140.82.113.10|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/x-gzip]
Saving to: 'postfixadmin-3.3.11.tar.gz'

postfixadmin-3.3.11.tar.gz        [ <=>                                           ]   1.78M  11.2MB/s    in 0.2s

2025-02-25 21:54:52 (11.2 MB/s) - 'postfixadmin-3.3.11.tar.gz' saved [1869783]

ubuntu@mail:~$ sudo mkdir -p /var/www/
ubuntu@mail:~$ sudo tar xvf postfixadmin-3.3.11.tar.gz -C /var/www/
postfixadmin-postfixadmin-3.3.11/
```

Nos aseguramos de cambiarle el nombre.

```
ubuntu@mail:~$ sudo mv /var/www/postfixadmin-postfixadmin-3.3.11/ /var/www/postfixadmin
```

Necesitamos un directorio templates_c.

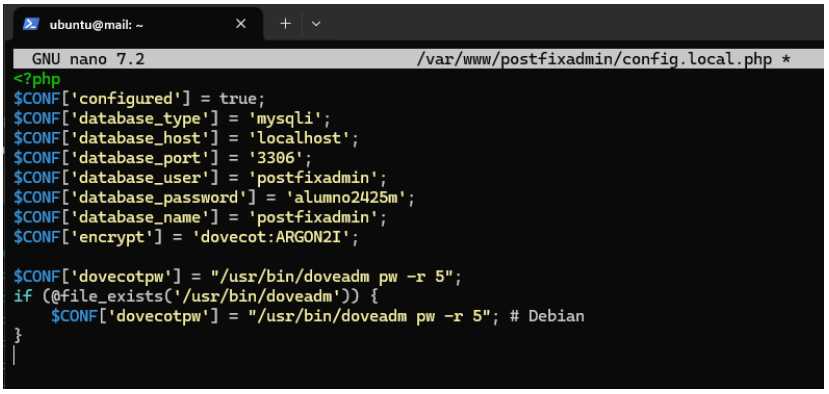
```
ubuntu@mail:~$ sudo mkdir -p /var/www/postfixadmin/templates_c
ubuntu@mail:~$ sudo apt install acl
Reading package lists... Done
```

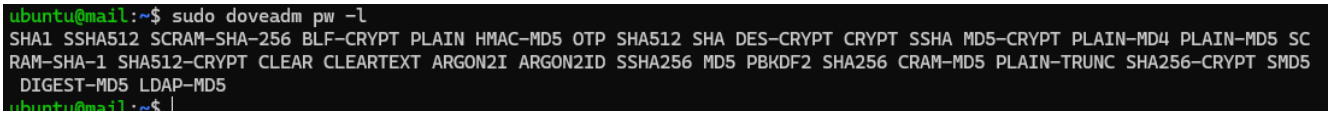
```
ubuntu@mail:~$ sudo setfacl -R -m u:www-data:rwx /var/www/postfixadmin/templates_c/
ubuntu@mail:~$ sudo setfacl -R -m u:www-data:rx /etc/letsencrypt/live/ /etc/letsencrypt/archive/
```

Vamos a crear una base de datos usuario para PostFixAdmin.

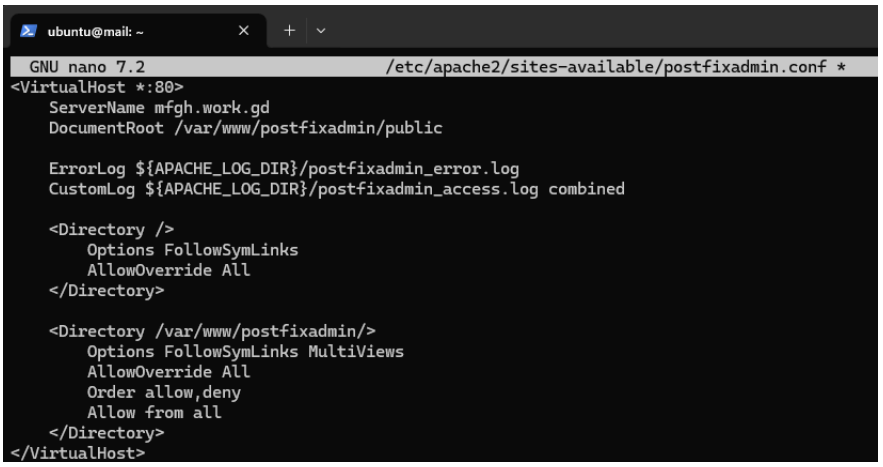
```
ubuntu@mail:~$ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 39
Server version: 10.11.8-MariaDB-0ubuntu0.24.04.1 Ubuntu 24.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE postfixadmin;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> CREATE USER 'postfixadmin'@'localhost' IDENTIFIED BY 'alumno2425m';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON postfixadmin.* TO 'postfixadmin'@'localhost';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]>
```

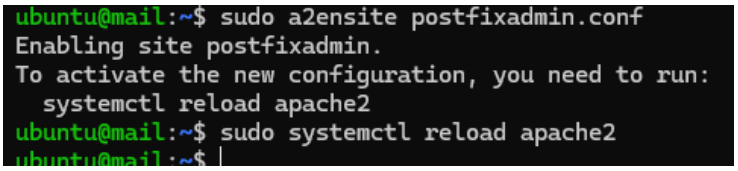
El archivo de configuración predeterminado de PostfixAdmin es config.inc.php. Necesitamos crear un fichero config.local.php y agregar configuraciones personalizadas.





Creamos un sitio virtual en apache y lo activamos.

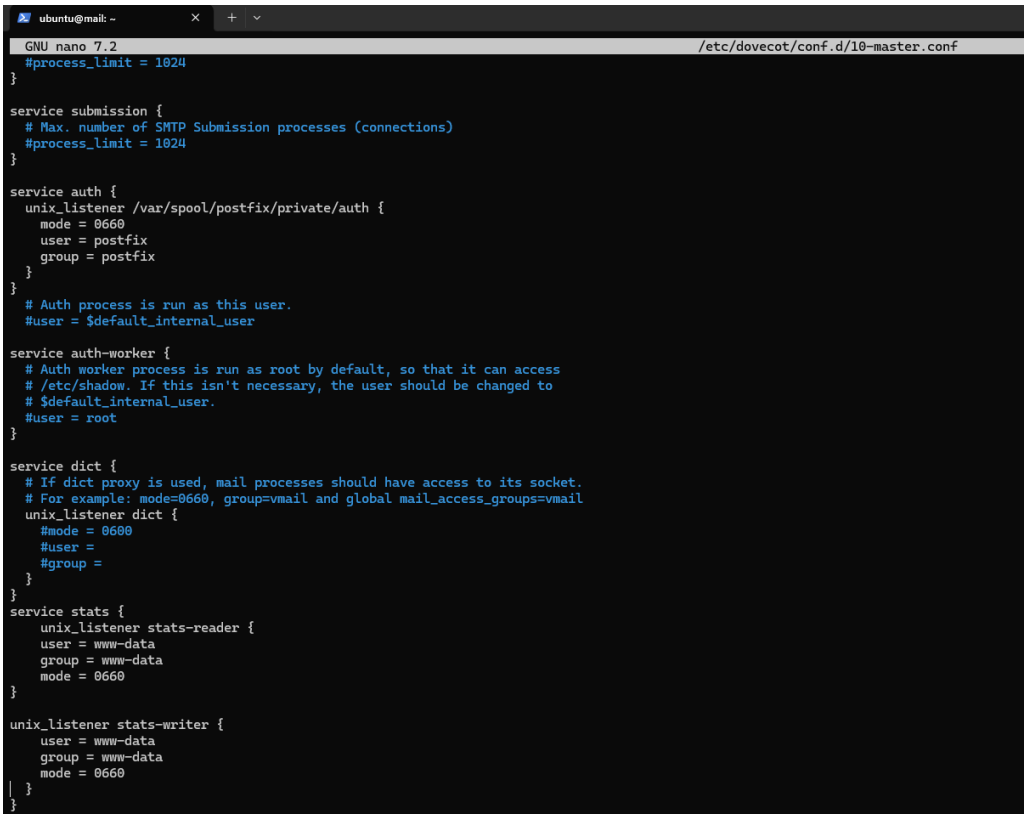




Descargamos todo lo de php en la versión 8.3

También la librería de apache.

Añadimos lo siguiente en dovecot.

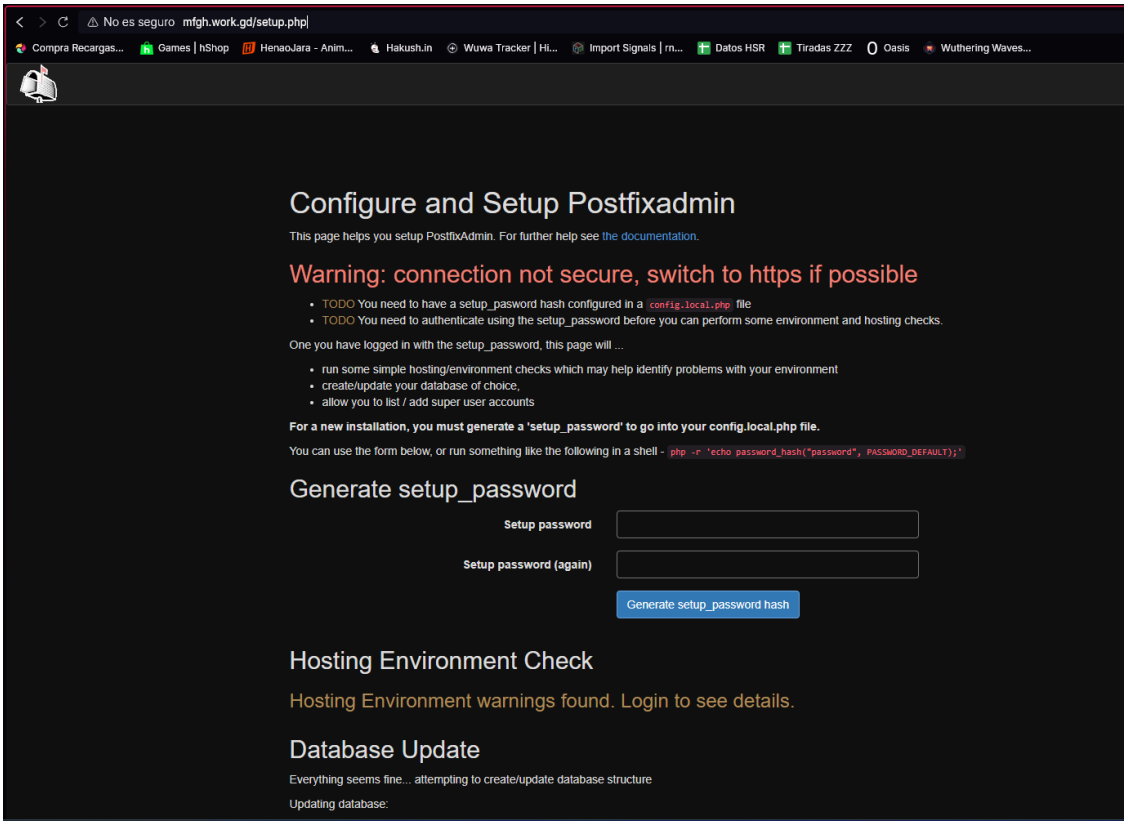


Añadimos a dovecot al servidor web y le damos permisos.

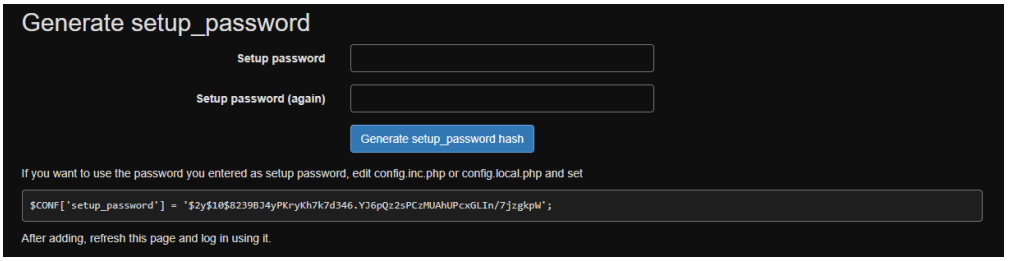
```
ubuntu@mail:~$ sudo gpasswd -a www-data dovecot
Adding user www-data to group dovecot
```

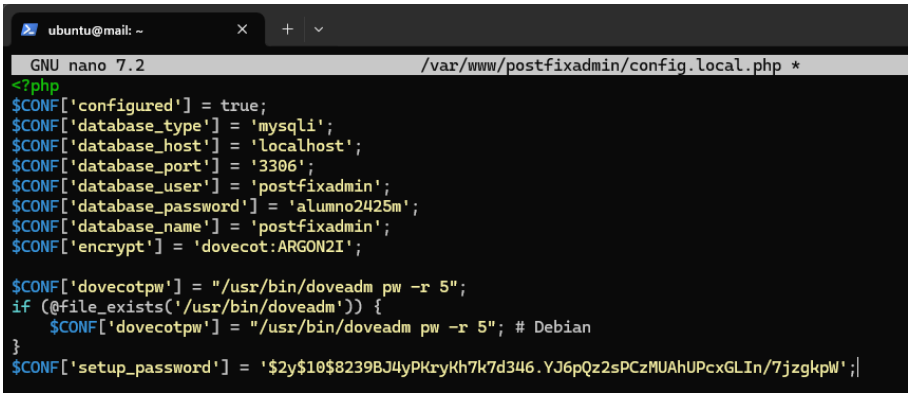
```
ubuntu@mail:~$ sudo systemctl restart dovecot
ubuntu@mail:~$ sudo setfacl -R -m u:www-data:rwx /var/run/dovecot/stats-reader /var/run/dovecot/stats-writer
```

Ahora accedemos al setup.php.

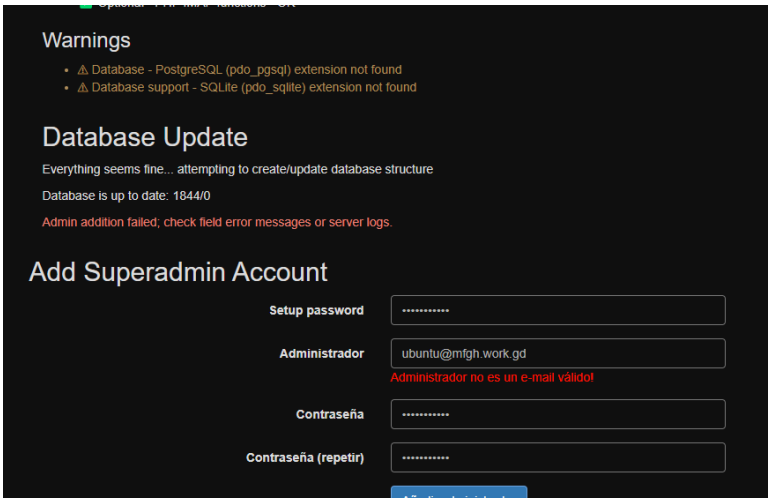


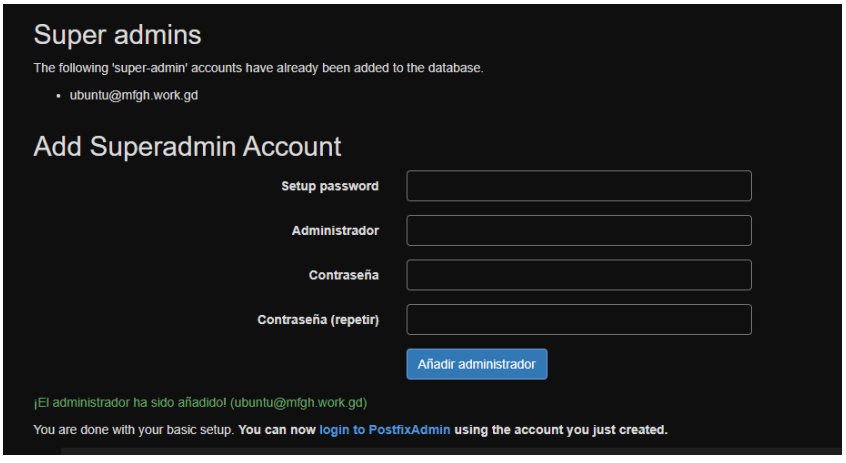
Introduzco mi contraseña (alumno2425m) y me genera un Hash, y esa línea que nos da vamos a añadirla a config.local.php.



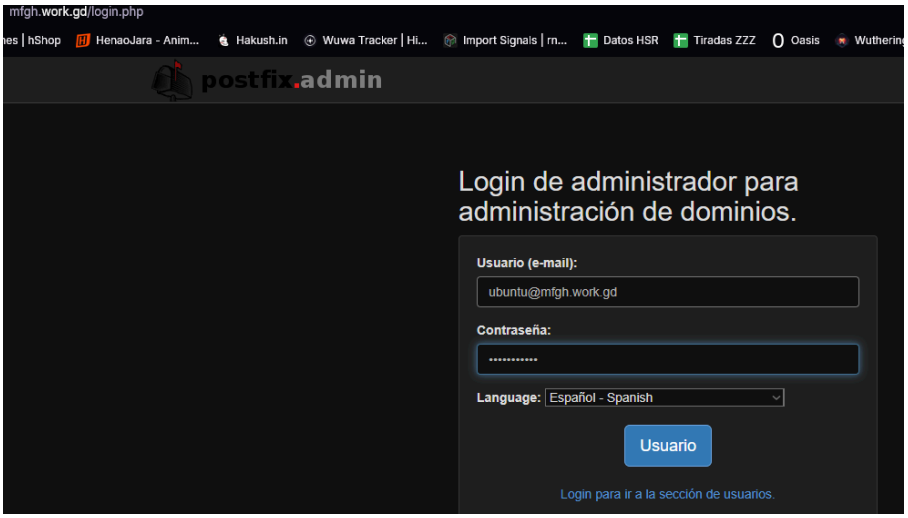


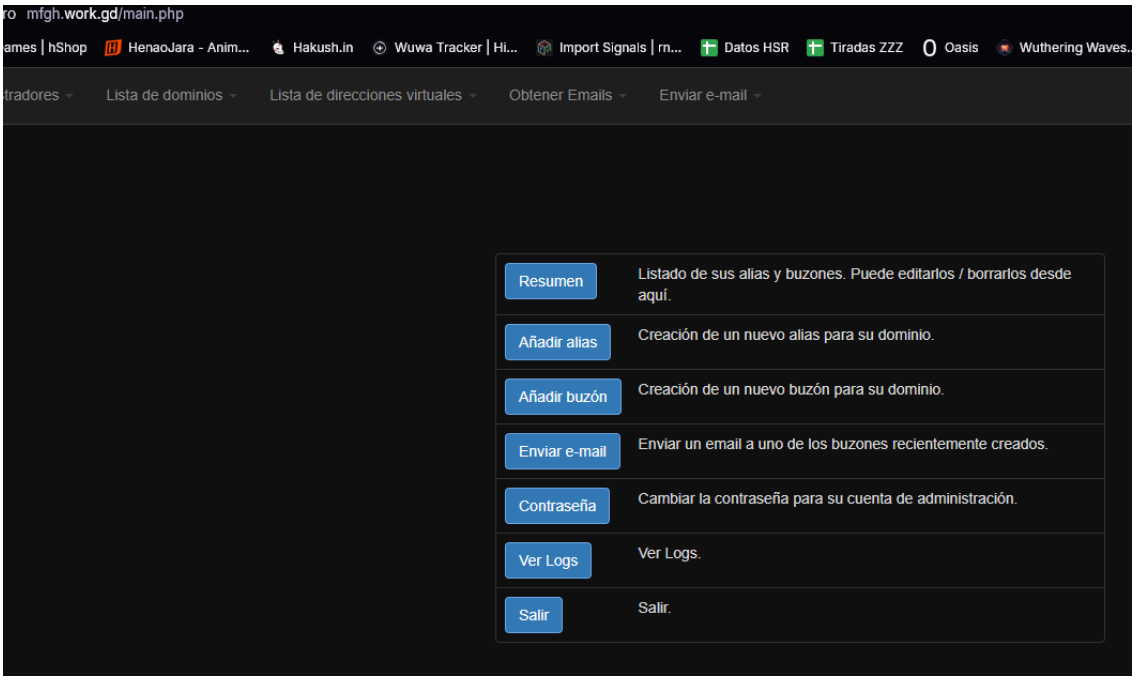
Ahora recargamos la página y añadimos un super admin.



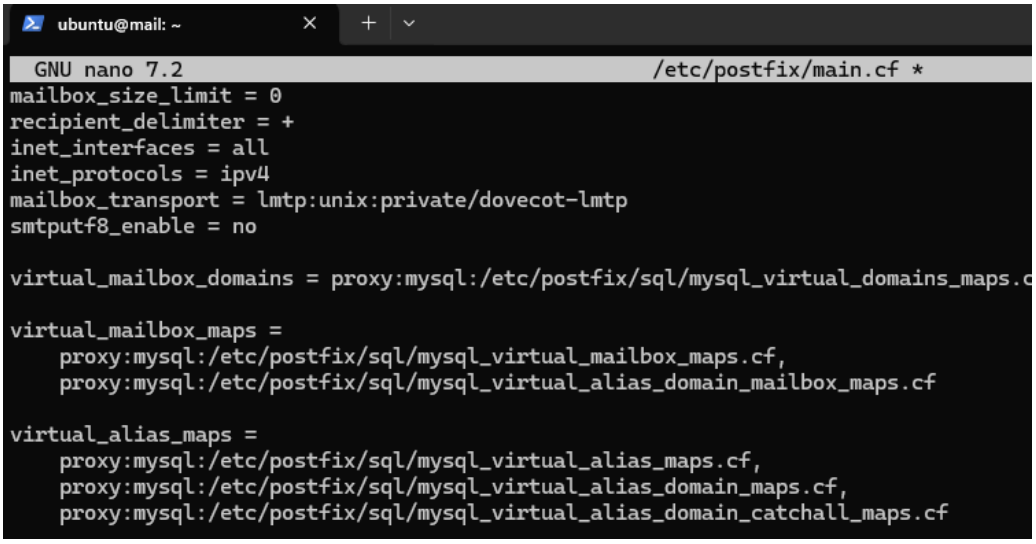


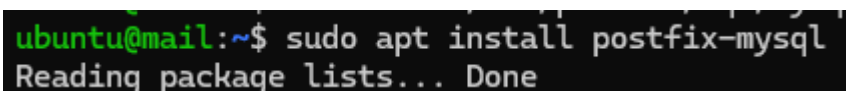
Ahora podemos ir a login.php y logear.



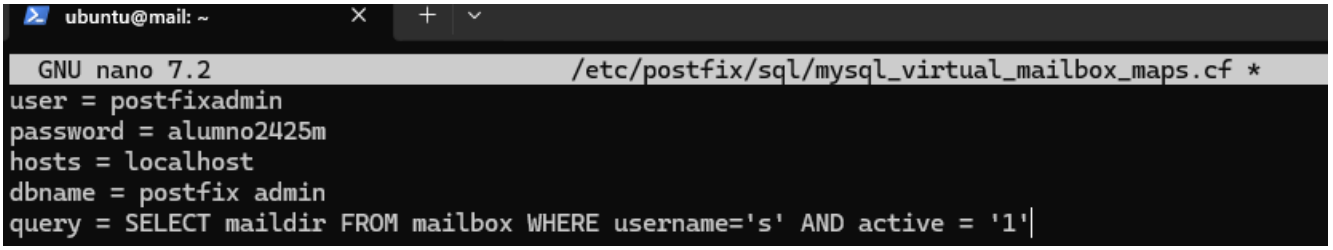


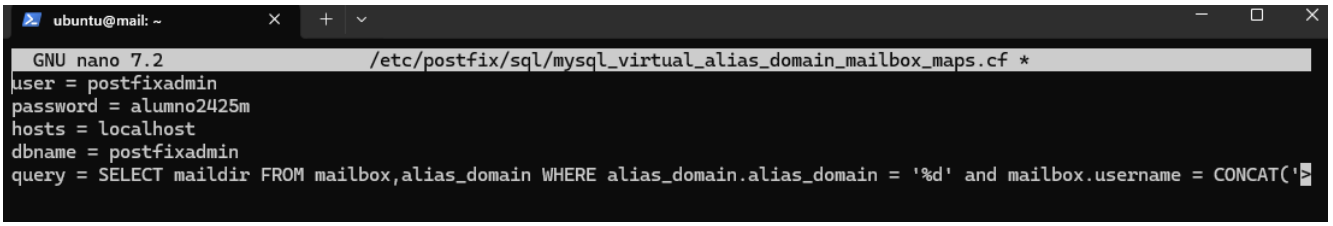
Ahora vamos a main.cf y añadimos esto (empezando desde virtual_mailbox_domains).

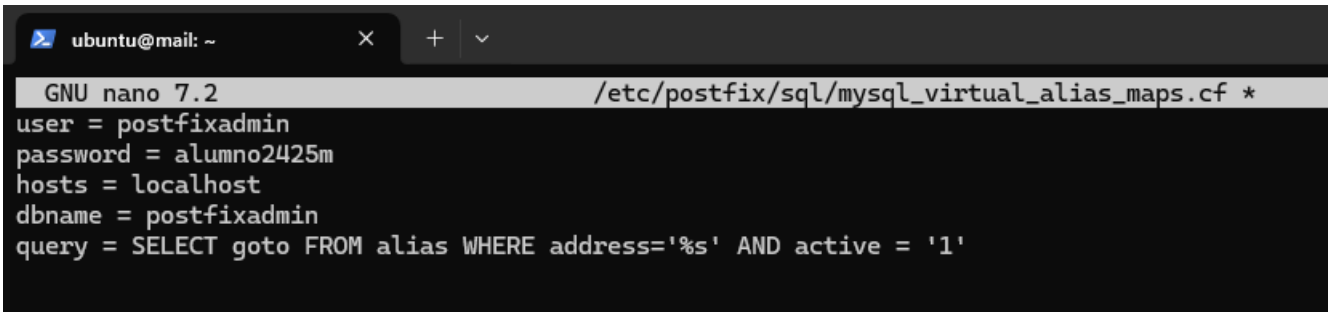


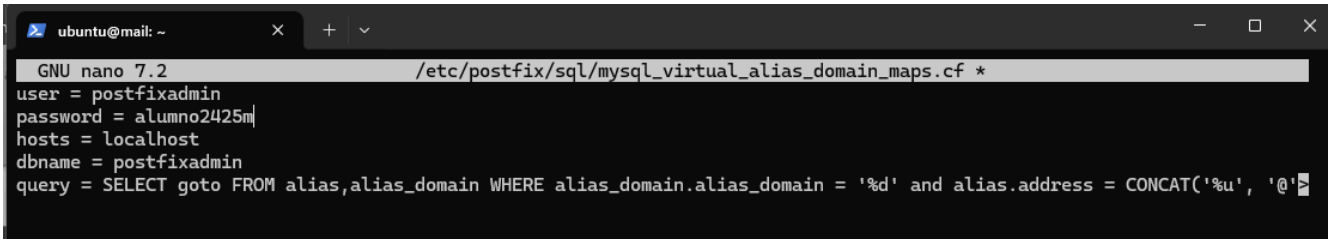


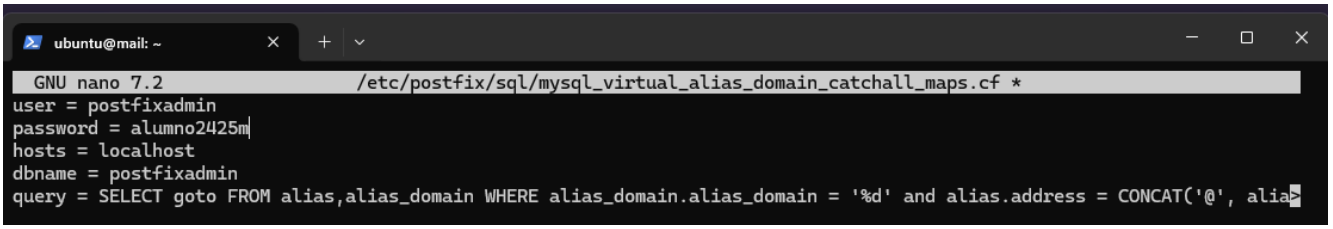
De forma predeterminada, Postfix envía correos electrónicos solo a usuarios con una cuenta Unix local, así que debemos configurar Postfix para que utilice dominios de buzones virtuales.











Dado que las contraseñas de la base de datos se almacenan como texto sin formato, solo los usuarios postfix y root pueden leerlos lo que se hace ejecutando los siguientes comandos:

El mydestination contiene una lista de nombres de dominio que recibirán correos electrónicos entregados a cuentas Unix locales. En la parte 1, agregamos el nombre de dominio de apex a mydestination

A continuación, debemos cambiar el valor del mydestination en Postfix.

```
ubuntu@mail:~$ postconf mydestination
mydestination = mail.mfgh.work.gd, localhost.mfgh.work.gd, localhost, mfgh.work.gd, $myhostname, localhost.$mydomain
```

Volvamos a main.cf a añadir lo siguiente.

```
ubuntu@mail: ~
 GNU nano 7.2                                    /etc/postfix/main.cf *

virtual_mailbox_domains = proxy:mysql:/etc/postfix/sql/mysql_virtual_domains_maps.cf

virtual_mailbox_maps =
    proxy:mysql:/etc/postfix/sql/mysql_virtual_mailbox_maps.cf,
    proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_mailbox_maps.cf

virtual_alias_maps =
    proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_maps.cf,
    proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_maps.cf,
    proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_catchall_maps.cf

virtual_transport = lmtp:unix:private/dovecot-lmtp

virtual_mailbox_base = /var/vmail
virtual_minimum_uid = 2000
virtual_uid_maps = static:2000
virtual_gid_maps = static:2000
```

Reseteamos y creamos al usuario vmail y creamos, dando permisos, la carpeta /var/mail:

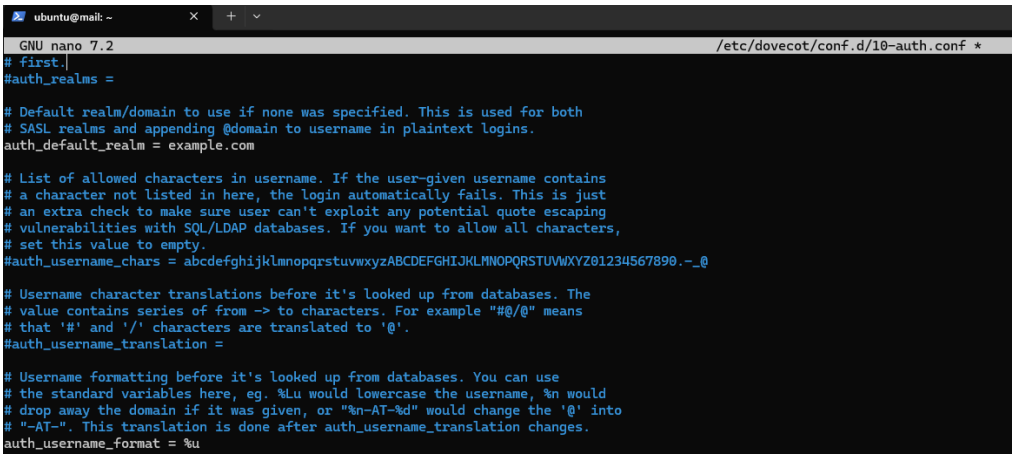
```
ubuntu@mail:~$ sudo systemctl restart postfix
ubuntu@mail:~$ sudo adduser vmail --system --group --uid 2000 --disabled-login --no-create-home
info: Adding system user `vmail' (UID 2000) ...
info: Adding new group `vmail' (GID 2000) ...
info: Adding new user `vmail' (UID 2000) with group `vmail' ...
useradd warning: vmail's uid 2000 is greater than SYS_UID_MAX 999
info: Not creating `/nonexistent'.
ubuntu@mail:~$ sudo mkdir /var/vmail/
ubuntu@mail:~$ sudo chown vmail:vmail /var/vmail -R
ubuntu@mail:~$
```

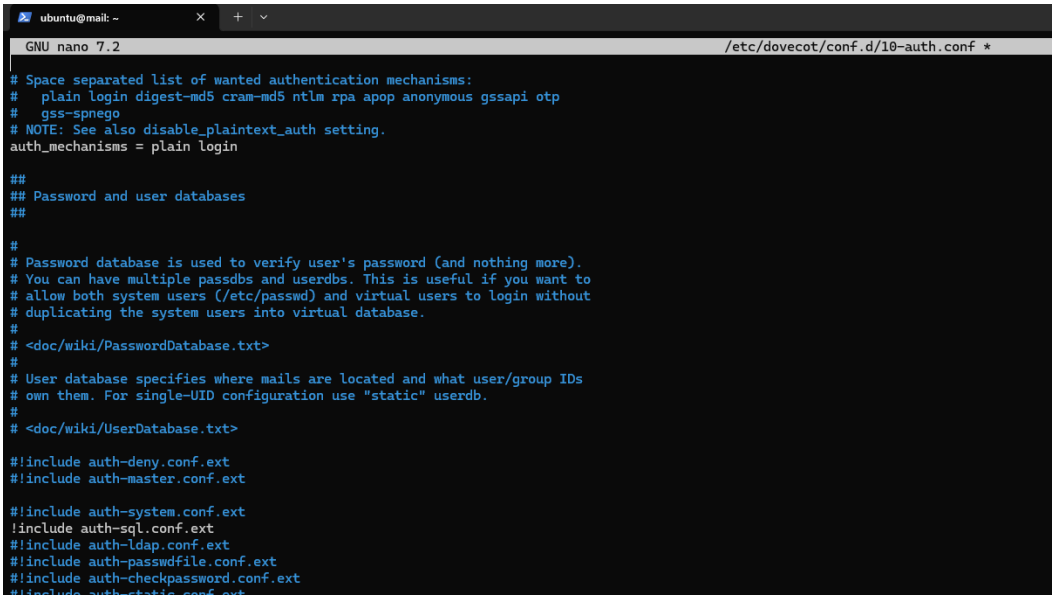
```
ubuntu@mail:~$ sudo apt install dovecot-mysql
Reading package lists... Done
```

Ahora vamos a 10-mail.conf a añadir lo siguiente.

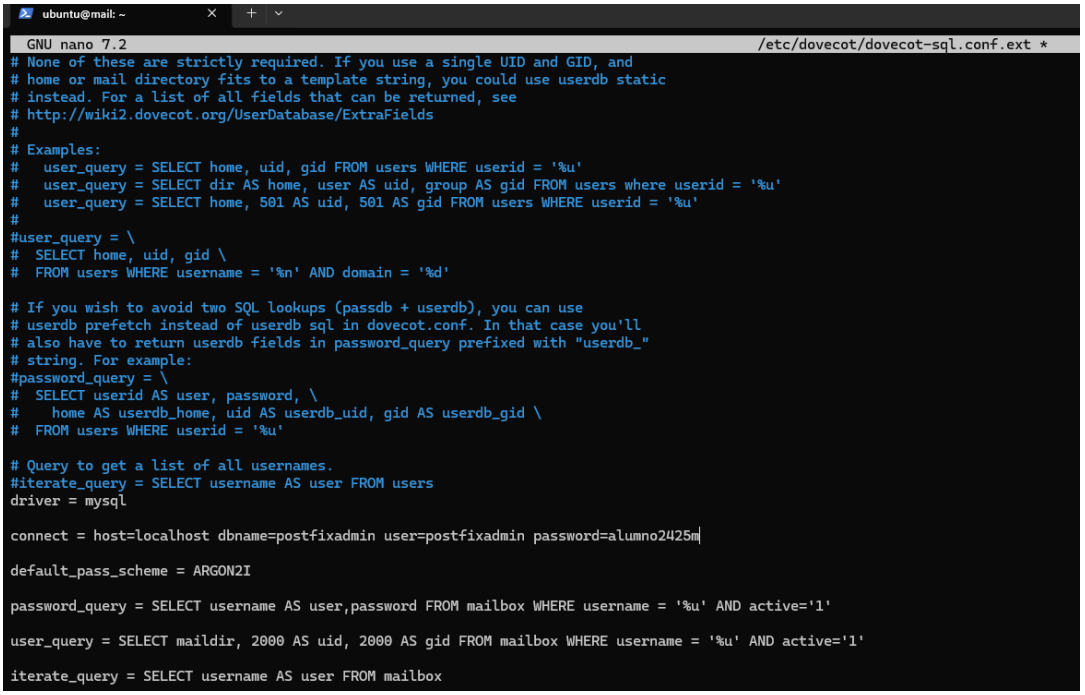
```
ubuntu@mail: ~
 GNU nano 7.2                                /etc/dovecot/conf.d/10-mail.conf *
#   mail_location = mbox:/var/mail/%d/%1n/%n:INDEX=/var/indexes/%d/%1n/%n
#
# <doc/wiki/MailLocation.txt>
#
mail_location = maildir:~/Maildir
mail_home = /var/vmail/%d/%n/
```

Ahora a 10-auth.conf y ajustamos lo siguiente.

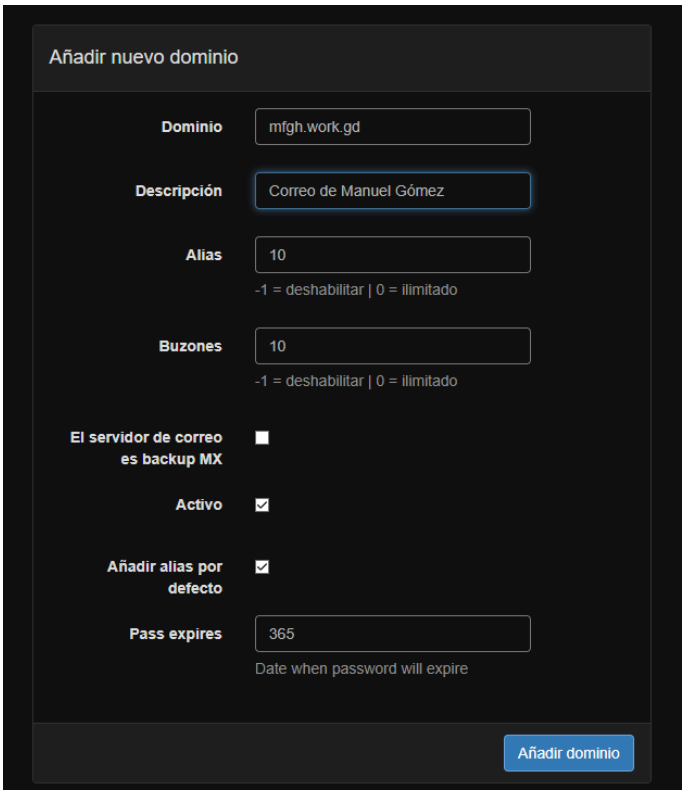




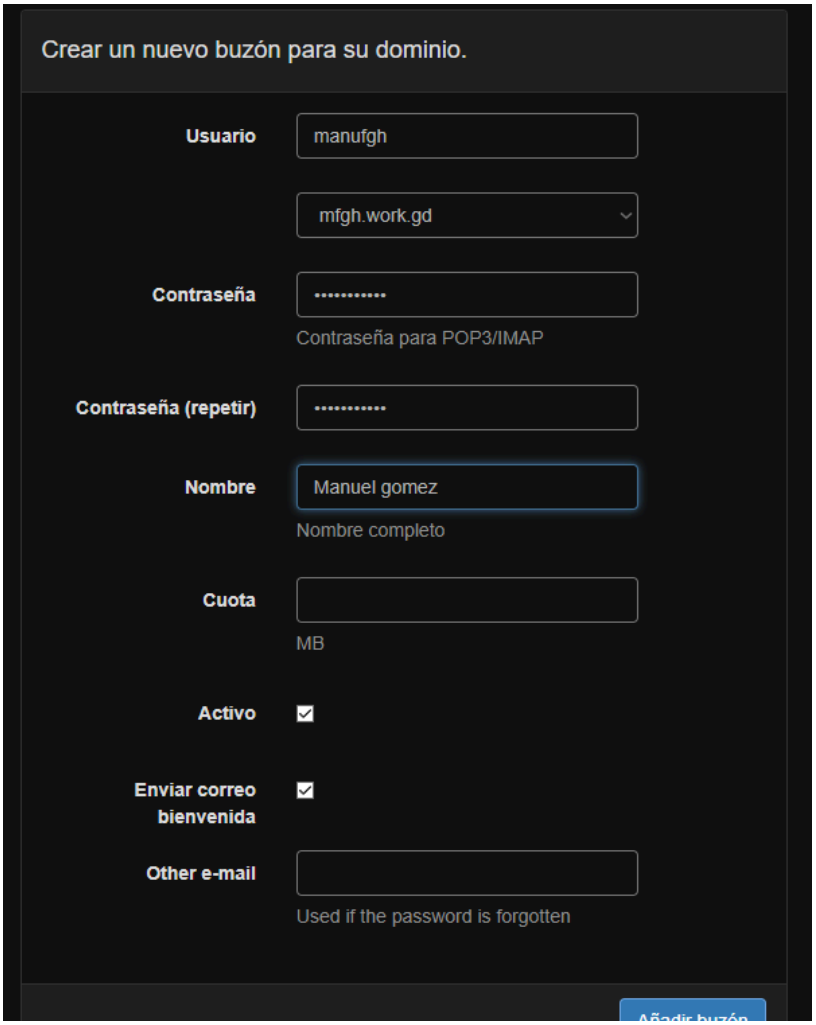
Ahora vamos a dovecot-sql.conf.ext y añadimos lo siguiente al final.



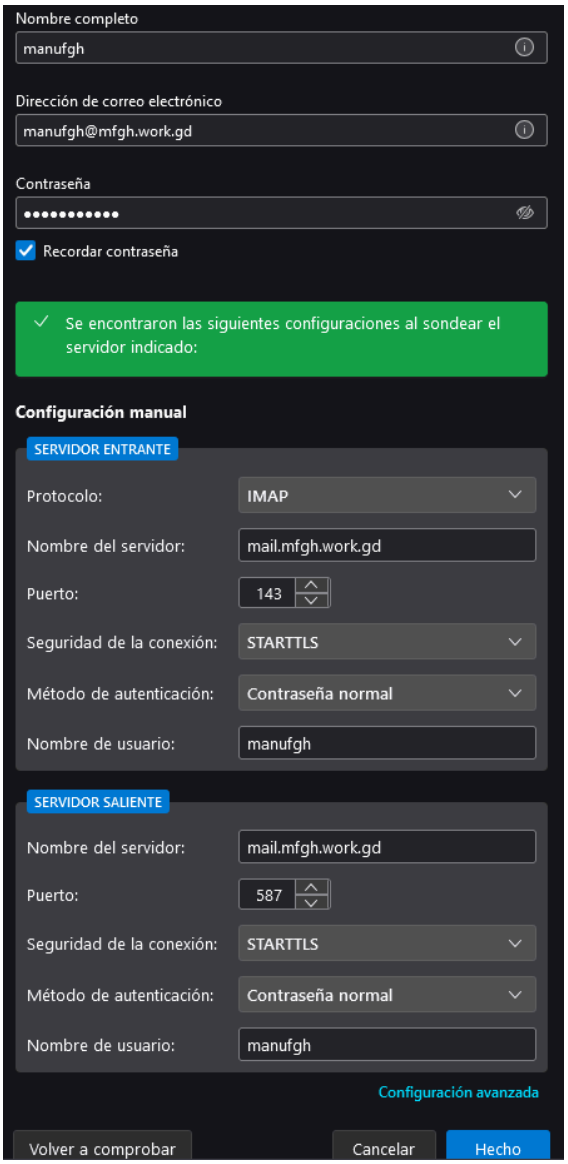
Con esto hecho, ya podemos volver al dominio. Y vamos a añadir un nuevo dominio.



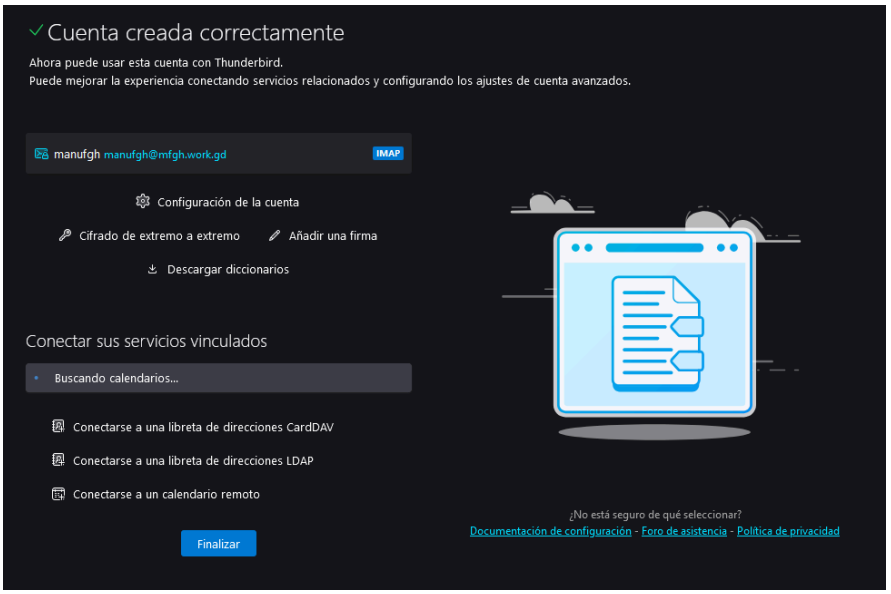
Y también un usuario para el dominio.



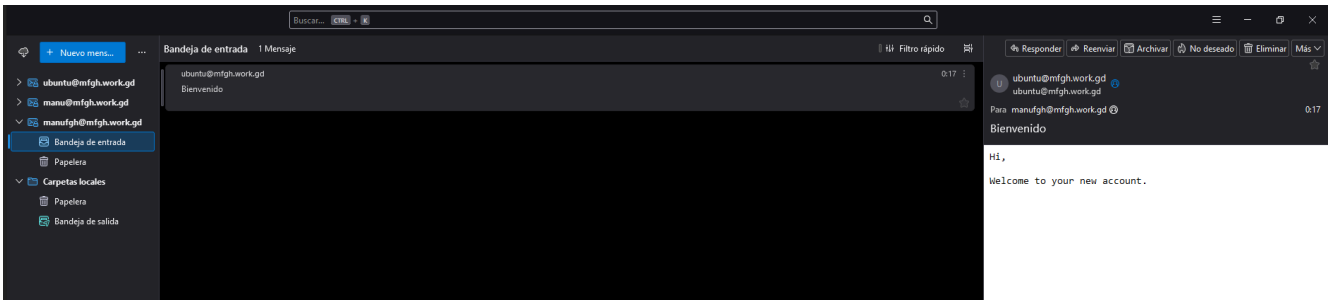
Iniciamos sesión con el mismo en Thunderbird, y configuramos los puertos manualmente.



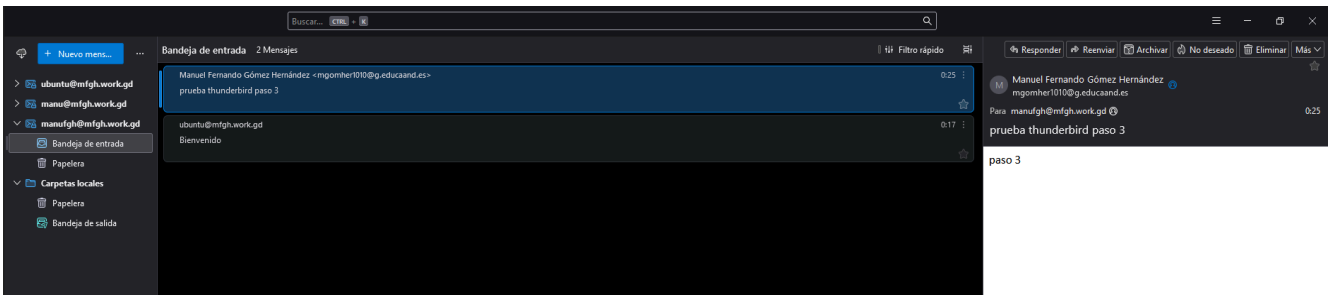
Y listo.



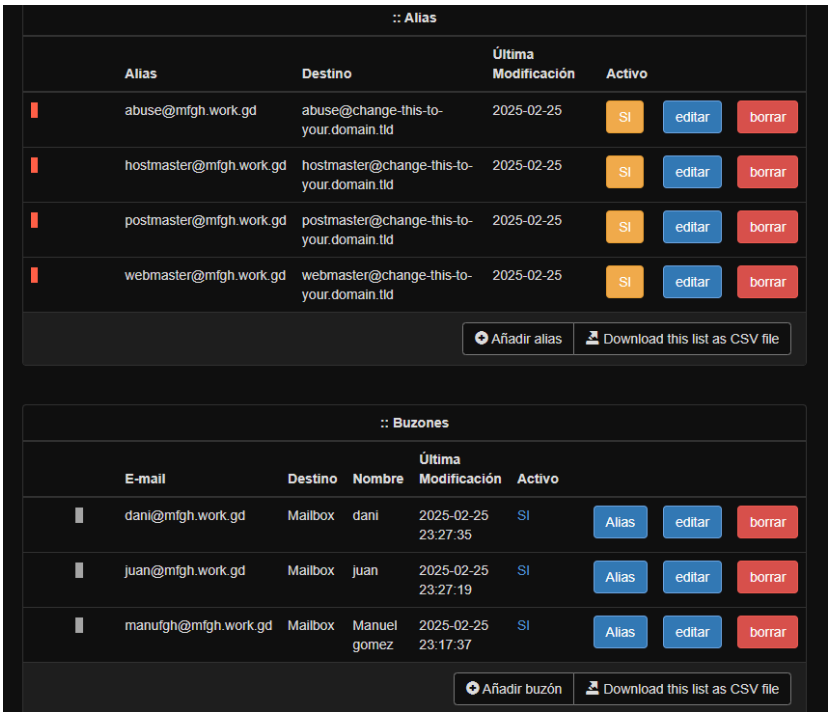
También ha llegado el correo de bienvenida.



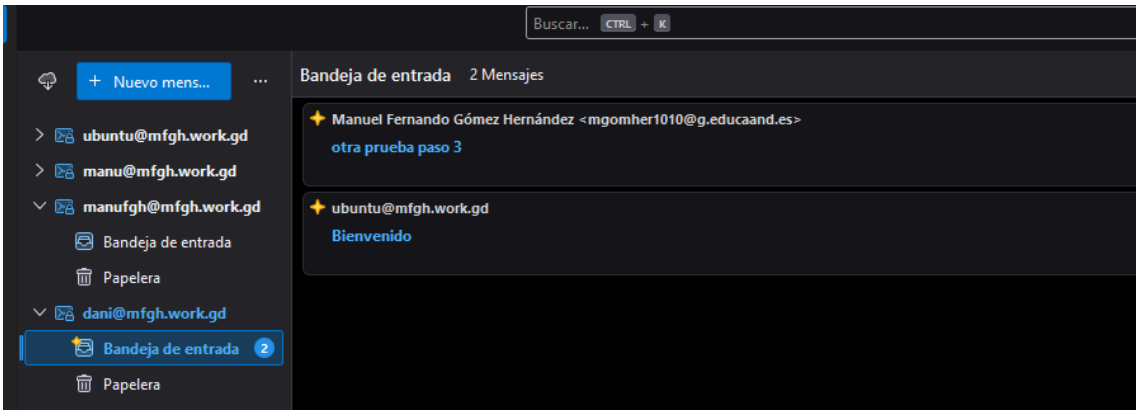
El correo desde mi cuenta de educaand ha funcionado.



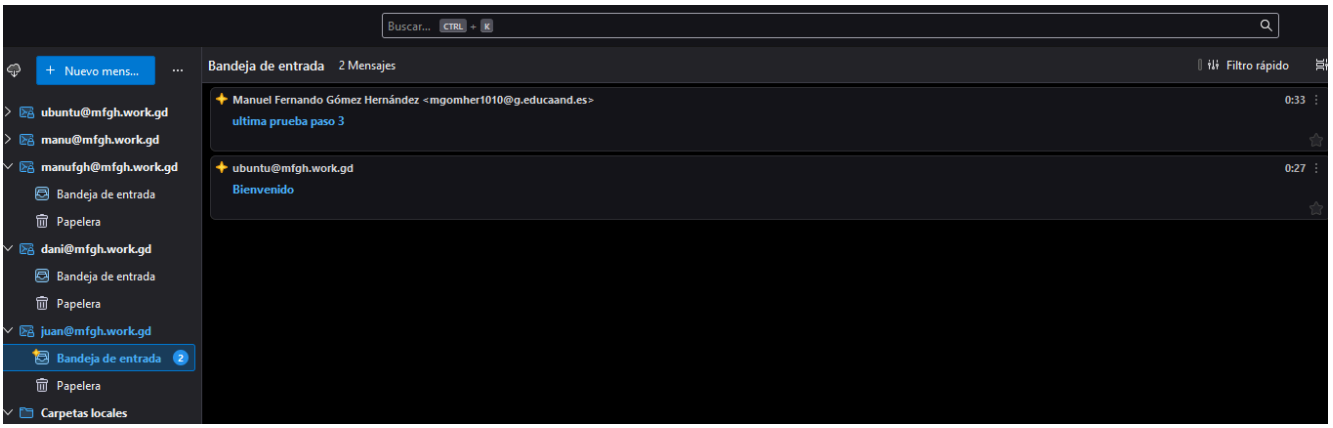
Probando con más usuarios.



Con el usuario dani funciona.



Y con el usuario Juan también funciona.

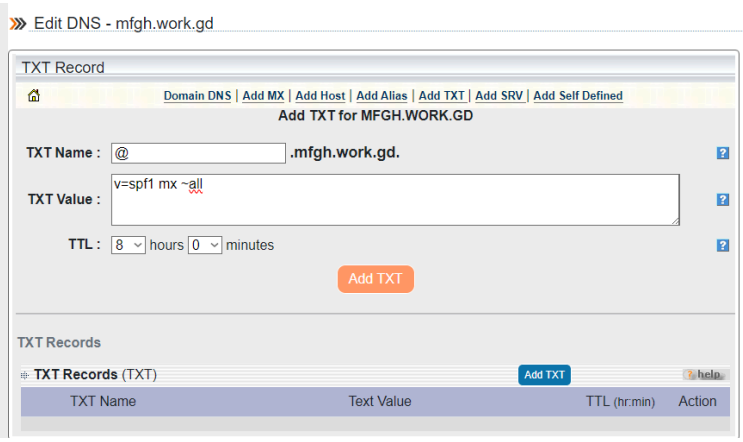


Paso 4:

Para luchar contra el spam y tener un servidor de correo confiable para el mundo exterior, es necesaria la instalación de varias técnicas: SPF y DKIM. Configura estas mejoras al correo y verifica su correcto funcionamiento mediante el correcto PASS de los controles en los códigos fuentes de los correos electrónicos y, mediante el uso de herramientas de chequeo externas como las que te indican en el tutorial de referencia. Se debe configurar adecuadamente DNSexit capturando pantallas del mismo.

La guía que estamos utilizando también implementa la técnica de autenticación de correo DMARC. En esta práctica sólo se va a pedir su implementación de manera opcional para **subir 1 punto** la calificación final de la misma.

Vamos a empezar por crear un nuevo registro TXT en el dominio.



Vemos que se ha creado exitosamente.

```
ubuntu@mail:~$ dig mfgh.work.gd txt

; <<>> DiG 9.18.30-0ubuntu0.24.04.2-Ubuntu <<>> mfgh.work.gd txt
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 30531
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;mfgh.work.gd.                  IN      TXT

;; AUTHORITY SECTION:
mfgh.work.gd.           1200    IN      SOA     ns1.dnsexit.com. admin.netdorm.com. 2000060701 86000 180000 1814400 1200

;; Query time: 1631 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Tue Feb 25 23:41:28 UTC 2025
;; MSG SIZE  rcvd: 106
```

Vamos a instalar lo siguiente.

```
ubuntu@mail:~$ sudo apt install postfix-policyd-spf-python
```

Añadimos esta línea al final de master.cf

```
ubuntu@mail: ~
 GNU nano 7.2                                                    /etc/postfix/master.cf *
#
uucp      unix  -       n       n       -       -       pipe
  flags=Fqhu user=uucp argv=uux -r -n -z -a$sender - $nexthop!rmail ($recipient)
#
# Other external delivery methods.
#
#ifmail    unix  -       n       n       -       -       pipe
#  flags=F user=ftn argv=/usr/lib/ifmail/ifmail -r $nexthop ($recipient)
#bsmtp     unix  -       n       n       -       -       pipe
#  flags=Fq. user=bsmtp argv=/usr/lib/bsmtp/bsmtp -t$nexthop -f$sender $recipient
#scalemail-backend unix -       n       n       -       2       pipe
#  flags=R user=scalemail argv=/usr/lib/scalemail/bin/scalemail-store ${nexthop} ${user} ${extension}
#mailman   unix  -       n       n       -       -       pipe
#  flags=FRX user=list argv=/usr/lib/mailman/bin/postfix-to-mailman.py ${nexthop} ${user}
submission     inet     n    -    y    -    -    smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_security_level=encrypt
  -o smtpd_tls_wrappermode=no
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_relay_restrictions=permit_sasl_authenticated,reject
  -o smtpd_recipient_restrictions=permit_mynetworks,permit_sasl_authenticated,reject
  -o smtpd_sasl_type=dovecot
  -o smtpd_sasl_path=private/auth

policyd-spf  unix  -       n       n       -       0       spawn
    user=policyd-spf argv=/usr/bin/policyd-spf
```

Ahora en main.cf

```
ubuntu@mail: ~
 GNU nano 7.2                                                    /etc/postfix/main.cf *
virtual_alias_maps =
    proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_maps.cf,
    proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_maps.cf,
    proxy:mysql:/etc/postfix/sql/mysql_virtual_alias_domain_catchall_maps.cf

virtual_transport = lmtp:unix:private/dovecot-lmtp

virtual_mailbox_base = /var/vmail
virtual_minimum_uid = 2000
virtual_uid_maps = static:2000
virtual_gid_maps = static:2000

policyd-spf_time_limit = 3600
smtpd_recipient_restrictions =
   permit_mynetworks,
   permit_sasl_authenticated,
   reject_unauth_destination,
   check_policy_service unix:private/policyd-spf
```

```
ubuntu@mail:~$ sudo systemctl restart postfix
```

Ahora instalaremos las tools de opendkim.

```
ubuntu@mail:~$ sudo apt install opendkim opendkim-tools
```

Añadimos al usuario postfix al grupo que se ha debido crear.

```
ubuntu@mail:~$ sudo gpasswd -a postfix opendkim
Adding user postfix to group opendkim
ubuntu@mail:~$
```

Ahora entramos a sudo /etc/opendkim.conf y lo editamos para que esté así.

```
ubuntu@mail: ~
 GNU nano 7.2                                                    /etc/opendkim.conf
# This is a basic configuration for signing and verifying. It can easily be
# adapted to suit a basic installation. See opendkim.conf(5) and
# /usr/share/doc/opendkim/examples/opendkim.conf.sample for complete
# documentation of available configuration parameters.

Syslog                  yes
Logwhy                  yes
SyslogSuccess           yes
#LogWhy                 no

# Common signing and verification parameters. In Debian, the "From" header is
# oversigned, because it is often the identity key used by reputation systems
# and thus somewhat security sensitive.
Canonicalization        relaxed/simple
Mode                    sv
SubDomains              no
OversignHeaders         From

AutoRestart         yes
AutoRestartRate     10/1M
Background          yes
DNSTimeout          5
SignatureAlgorithm  rsa-sha256
# Signing domain, selector, and key (required). For example, perform signing
```

```
ubuntu@mail: ~
 GNU nano 7.2                                                    /etc/opendkim.conf *
OversignHeaders         From

AutoRestart         yes
AutoRestartRate     10/1M
Background          yes
DNSTimeout          5
SignatureAlgorithm  rsa-sha256
# Signing domain, selector, and key (required). For example, perform signing
# for domain "example.com" with selector "2020" (2020._domainkey.example.com),
# using the private key stored in /etc/dkimkeys/example.private. More granular
# setup options can be found in /usr/share/doc/opendkim/README.opendkim.
#Domain                 example.com
#Selector               2020
#KeyFile                /etc/dkimkeys/example.private

# In Debian, opendkim runs as user "opendkim". A umask of 007 is required when
# using a local socket with MTAs that access the socket as a non-privileged
# user (for example, Postfix). You may need to add user "postfix" to group
# "opendkim" in that case.
UserID                  opendkim
UMask                   007

# Map domains in From addresses to keys used to sign messages
KeyTable            refile:/etc/opendkim/key.table
SigningTable        refile:/etc/opendkim/signing.table

# Hosts to ignore when verifying signatures
ExternalIgnoreList  /etc/opendkim/trusted.hosts

# A set of internal hosts whose mail should be signed
InternalHosts       /etc/opendkim/trusted.hosts
```

Creamos la carpeta de las keys.

```
ubuntu@mail:~$ sudo mkdir /etc/opendkim
ubuntu@mail:~$ sudo mkdir /etc/opendkim/keys
ubuntu@mail:~$ sudo chown -R opendkim:opendkim /etc/opendkim
ubuntu@mail:~$ sudo chmod go-rw /etc/opendkim/keys/
ubuntu@mail:~$
```

Creamos las signing tables.

```
ubuntu@mail: ~
 GNU nano 7.2                                          /etc/opendkim/signing.table *
*@mfgh.work.gd     default._domainkey.mfgh.work.gd
*@*.mfgh.work.gd     default._domainkey.mfgh.work.gd
```

Ahora creo la key table.

```
ubuntu@mail: ~
 GNU nano 7.2                                  /etc/opendkim/key.table *
default._domainkey.mfgh.work.gd     mfgh.work.gd:default:/etc/opendkim/keys/mfgh.work.gd/default.private
```

Ahora los trusted hosts.

```
ubuntu@mail: ~
 GNU nano 7.2                                  /etc/opendkim/trusted.hosts *
127.0.0.1
localhost

.mfgh.work.gd
```

Creo la propia llave privada junto a su carpeta.

```
ubuntu@mail:~$ sudo mkdir /etc/opendkim/keys/mfgh.work.gd
ubuntu@mail:~$ sudo opendkim-genkey -b 2048 -d mfgh.work.gd -D /etc/opendkim/keys/mfgh.work.gd -s default -v
opendkim-genkey: generating private key
opendkim-genkey: private key written to default.private
opendkim-genkey: extracting public key
opendkim-genkey: DNS TXT record written to default.txt
```

```
ubuntu@mail:~$ sudo chown opendkim:opendkim /etc/opendkim/keys/mfgh.work.gd/default.private
ubuntu@mail:~$ sudo chmod 600 /etc/opendkim/keys/mfgh.work.gd/default.private
ubuntu@mail:~$
```

Comprobamos que se ha creado bien.

```
ubuntu@mail:~$ sudo cat /etc/opendkim/keys/mfgh.work.gd/default.txt
default._domainkey      IN      TXT     ( "v=DKIM1; h=sha256; k=rsa; "
          "p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA5IialHOCPusJNA8FXFntvMeckT2afVL4LJyPLtMXI9q9PoyFEhyLQwj4nm+SK32
uujOOOEMbpv4nN74YXLthw5DVt/LUDaH4V8wZ81deB9ijgZVZ90QOyH1SeC//VLLs2OMJ+jp0cLRH1vnjE+ERHvwaUVewbW3lOCSKFe4ueo77zMLJf+f6Sf8
LYbf/CchmrsQP5NpWPIz02G"
          "n9udMrk53GYAn7k92/cFyYuVpzhjnREw/+Nj/SIV9K/uaKt2FZbS2xc7HmQZTR4pRdQFgEztI9xZo3QAHgqGVDxm/sk+Y2LbvggD2YYD2ZwTi
ioJwYtdsd5XgpsiCQkPrBGcFD0QIDAQAB" )  ; ----- DKIM key default for mfgh.work.gd
ubuntu@mail:~$
```

Esta clave debemos añadirla ahora en otro txt en nuestro freedns:

Comprobamos, aunque a nosotros nos sale not secure.

```
ubuntu@mail:~$ sudo opendkim-testkey -d mfgh.work.gd -s default -vvv
opendkim-testkey: using default configfile /etc/opendkim.conf
opendkim-testkey: checking key 'default._domainkey.mfgh.work.gd'
opendkim-testkey: key not secure
opendkim-testkey: key OK
```

Creamos la siguiente carpeta.

```
ubuntu@mail:~$ sudo mkdir /var/spool/postfix/opendkim

sudo chown opendkim:postfix /var/spool/postfix/opendkim
ubuntu@mail:~$ sudo chown opendkim:postfix /var/spool/postfix/opendkim
```

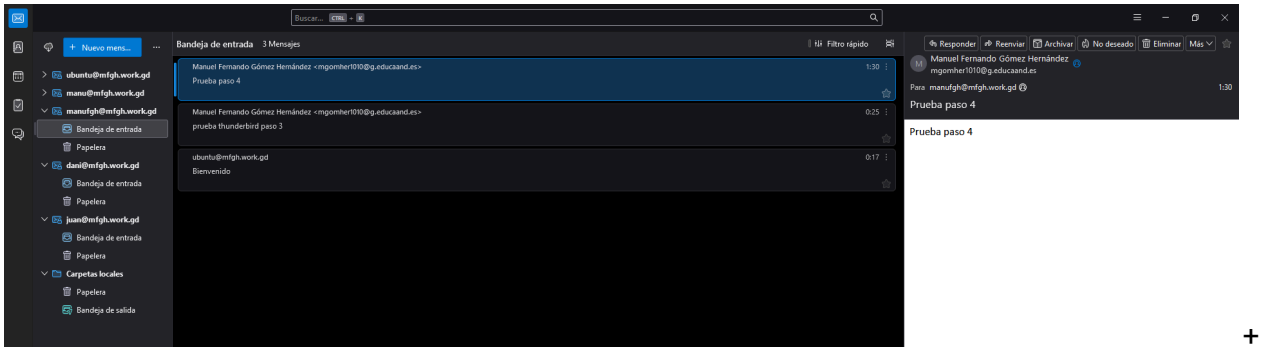
Entramos a /etc/opendkim.conf y comentamos la primera línea de socket y descomentamos la última. ()

```
ubuntu@mail: ~
  GNU nano 7.2                                        /etc/opendkim.conf *
SigningTable       refile:/etc/opendkim/signing.table

# Hosts to ignore when verifying signatures
ExternalIgnoreList  /etc/opendkim/trusted.hosts

# A set of internal hosts whose mail should be signed
InternalHosts       /etc/opendkim/trusted.hosts

# Socket for the MTA connection (required). If the MTA is inside a chroot jail,
# it must be ensured that the socket is accessible. In Debian, Postfix runs in
# a chroot in /var/spool/postfix, therefore a Unix socket would have to be
# configured as shown on the last line below.
#Socket                 local:/run/opendkim/opendkim.sock
#Socket                 inet:8891@localhost
#Socket                 inet:8891
Socket                  local:/var/spool/postfix/opendkim/opendkim.sock
```

Ahora vamos a main.cf y añadimos lo siguiente de Milter al final.

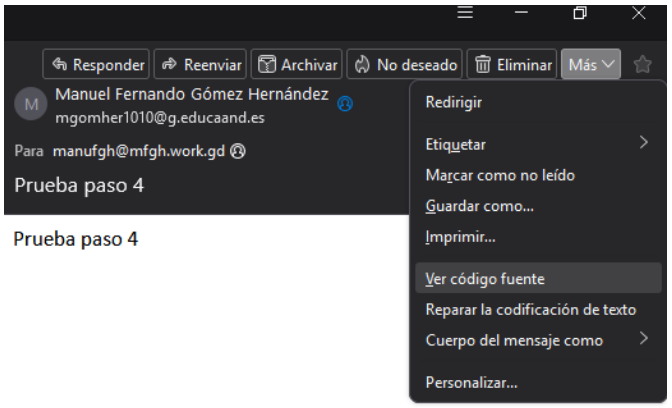
```
ubuntu@mail: ~
  GNU nano 7.2                                        /etc/postfix/main.cf *
virtual_uid_maps = static:2000
virtual_gid_maps = static:2000

policyd-spf_time_limit = 3600
smtpd_recipient_restrictions =
   permit_mynetworks,
   permit_sasl_authenticated,
   reject_unauth_destination,
   check_policy_service unix:private/policyd-spf

# Milter configuration
milter_default_action = accept
milter_protocol = 6
smtpd_milters = local:opendkim/opendkim.sock
non_smtpd_milters = $smtpd_milters
```

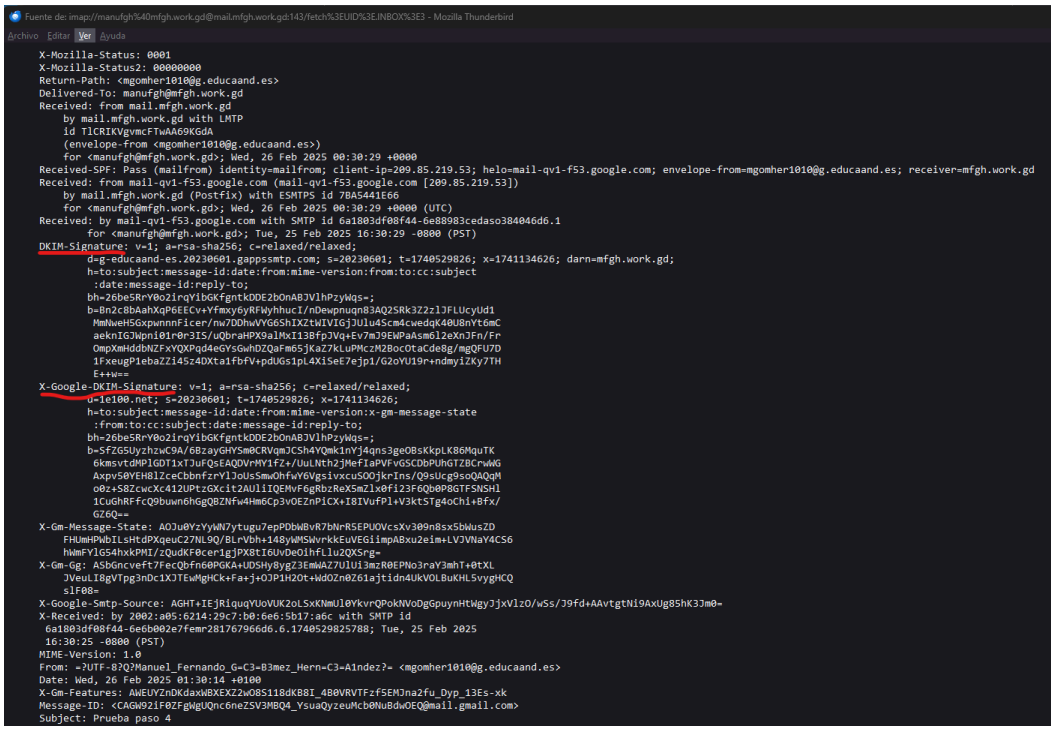
Vamos a enviar un mail de prueba.



+

Si pulsamos en más->Ver código fuente podemos observar lo siguiente.





```
Fuente de: imap://manufgh%40mfgh.work.gd@mail.mfgh.work.gd:143/fetch%3EUID%3E.INBOX%3E3 - Mozilla Thunderbird
Archivo  Editar  Ver  Ayuda

X-Mozilla-Status: 0001
X-Mozilla-Status2: 00000000
Return-Path: <mgomher1010@g.educaand.es>
Delivered-To: manufgh@mfgh.work.gd
Received: from mail.mfgh.work.gd
    by mail.mfgh.work.gd with LMTP
    id TlCRIKVgvmcFTwAA69KGdA
    (envelope-from <mgomher1010@g.educaand.es>)
    for <manufgh@mfgh.work.gd>; Wed, 26 Feb 2025 00:30:29 +0000
Received-SPF: Pass (mailfrom) identity=mailfrom; client-ip=209.85.219.53; helo=mail-qv1-f53.google.com; envelope-from=mgomher1010@g.educaand.es; receiver=mfgh.work.gd
Received: from mail-qv1-f53.google.com (mail-qv1-f53.google.com [209.85.219.53])
    by mail.mfgh.work.gd (Postfix) with ESMTPS id 7BA5441E66
    for <manufgh@mfgh.work.gd>; Wed, 26 Feb 2025 00:30:29 +0000 (UTC)
Received: by mail-qv1-f53.google.com with SMTP id 6a1803df08f44-6e88983cedaso384046d6.1
        for <manufgh@mfgh.work.gd>; Tue, 25 Feb 2025 16:30:29 -0800 (PST)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;
        d=g-educaand-es.20230601.gappssmtp.com; s=20230601; t=1740529826; x=1741134626; darn=mfgh.work.gd;
        h=to:subject:message-id:date:from:mime-version:from:to:cc:subject
         :date:message-id:reply-to;
        bh=26be5RrY0o2irqYibGKFgntkDDE2bOnABJVlhPzyWqs=;
        b=Bn2c8bAahXqP6EECv+Yfmxy6yRFWyhhucI/nDewpnuqn83AQ2SRk3Z2zlJFLUcyUd1
         MmNweH5GxpwnnnFicer/nw7DDhwVYG6ShIXZtWIVIGjJUlu4Scm4cwedqK40U8nYt6mC
         aeknIGJWpni01r0r3IS/uQbraHPX9alMxI13BfpJVq+Ev7mJ9EWPaAsm6l2eXnJFn/Fr
         OmpXmHddbNZFxYQXPqd4eGYsGwhDZQaFm65jKaZ7kLuPMczM2BocOtaCde8g/mgQFU7D
         1FxeugP1ebaZZi45z4DXta1FbfV+pdUGs1pL4XiSeE7ejp1/G2oYU19r+ndmyiZKy7TH
         E++w==
X-Google-DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;
        d=1e100.net; s=20230601; t=1740529826; x=1741134626;
        h=to:subject:message-id:date:from:mime-version:x-gm-message-state
         :from:to:cc:subject:date:message-id:reply-to;
        bh=26be5RrY0o2irqYibGKFgntkDDE2bOnABJVlhPzyWqs=;
        b=SfZG5UyzhzwC9A/6BzayGHYSm0CRVqmJCSh4YQmk1nYj4qns3geOBsKkpLK86MquTK
         6kmsvtdMPlGDT1xTJuFQsEAQDVrMY1fZ+/UuLNth2jMefIaPVFvGSCDbPUhGTZBCrwWG
         Axpv50YEH8lZceCbbnfzrYlJoUsSmwOhfwY6VgsivxcuSOOjkrIns/Q9sUcg9soQAQqM
         o02+S8ZcwcXc412UPtzGXcit2AUliIQEMvF6gRbzReXSmZlx0fi23F6Qb0P8GTFSNSH1
         1CuGhRFfcQ9buwn6hGgQBZNfw4Hm6Cp3vOEZnPiCX+I8IVufPl+V3ktSTg4oChi+Bfx/
         GZ6Q==
X-Gm-Message-State: AOJu0YzYyWN7ytugu7epPDbWBvR7bNrR5EPUOVcsXv309n8sx5bWusZD
    FHUmHPNbILsHtdPXqeuC27NL9Q/BLrVbh+148yWMSWvrkkEuVFGiimpA8xu2eim+LVJVNaY4CS6
    hWmFYlG54hxkPMI/zQuDKF8cer1gjPX8tI6UvOeOihfiLu2QX5rg=
X-Gm-Gg: ASbGncveft7FecQbfn60PGKA+UDSHy8ygZ3EmWAZ7UlUi3mzR0EPNo3raY3mhT+0tXL
    JVeuLI8gVTpg3nDc1XJTEwMgHCk+Fa+j+OJP1H2Ot+WdOZn0Z61ajtidn4UkVOLBuKHL5vygHCQ
    slF08=
X-Google-Smtp-Source: AGHT+IEjRiquqYUoVUK2oLSxKNmUl0YkvrQPokNVoDgGpuynHtWgyJjxVlzO/wSs/J9fd+AAvtgtNi9AxUg85hK3Jm0=
X-Received: by 2002:a05:6214:29c7:b0:6e6:5b17:a6c with SMTP id
 6a1803df08f44-6e6b002e7femr281767966d6.6.1740529825788; Tue, 25 Feb 2025
 16:30:25 -0800 (PST)
MIME-Version: 1.0
From: =?UTF-8?Q?Manuel_Fernando_G=C3=B3mez_Hern=C3=A1ndez?= <mgomher1010@g.educaand.es>
Date: Wed, 26 Feb 2025 01:30:14 +0100
X-Gm-Features: AWEUYZnDKdaxWBXEXZ2w08S118dKB8I_4B0VRVTFzf5EMJna2fu_Dyp_13Es-xk
Message-ID: <CAGW92if0ZFgWgUQnc6neZSV3MBQ4_YsuaQyzeuMcb0NuBdwOEQ@mail.gmail.com>
Subject: Prueba paso 4
```

El resto de tutoriales de esta serie también funcionan en Ubuntu 22.04 perfectamente y son de mucha utilidad para un servidor de correo electrónico en producción. No se pide su realización por motivos de tiempo.

Esta práctica debe ser documentada pero también será mostrada y explicada al profesor en sus objetivos principales.

No se evaluará nada fuera de plazo.

Enlaces de ayuda:

- Plataforma Pledin Servicio de correo electrónico
- Postfix Home Page
- Dovecot Home Page
- **Referencias Base utilizadas para la práctica**
    - Paso 1 Build Your Own Email Server on Ubuntu: Basic Postfix Setup - LinuxBabe
    - Paso 2 Part 2: Install Dovecot IMAP server on Ubuntu & Enable TLS Encryption
    - Paso 3 Part 3: PostfixAdmin – Create Virtual Mailboxes on Ubuntu Mail Server
    - Paso 4 Part 4: How to Set up SPF and DKIM with Postfix on Ubuntu Server
- Otra referencia útiles
    - A Mailserver on Ubuntu 16.04: Postfix, Dovecot, MySQL – Ex Ratione
    - How to set up a mail server on a GNU / Linux system
    - Correo electrónico con Postfix, Dovecot y Thunderbird en Ubuntu 20.04 (actualizado a Ubuntu 22.04 :-)
    - Servidor de correo: Postfix y Dovecot