

BOOTCAMP ESPECIALIDAD GNU/LINUX (2023)

Lab 09 - Servidor de Correo

Pedro Antonio Mayorgas Parejo

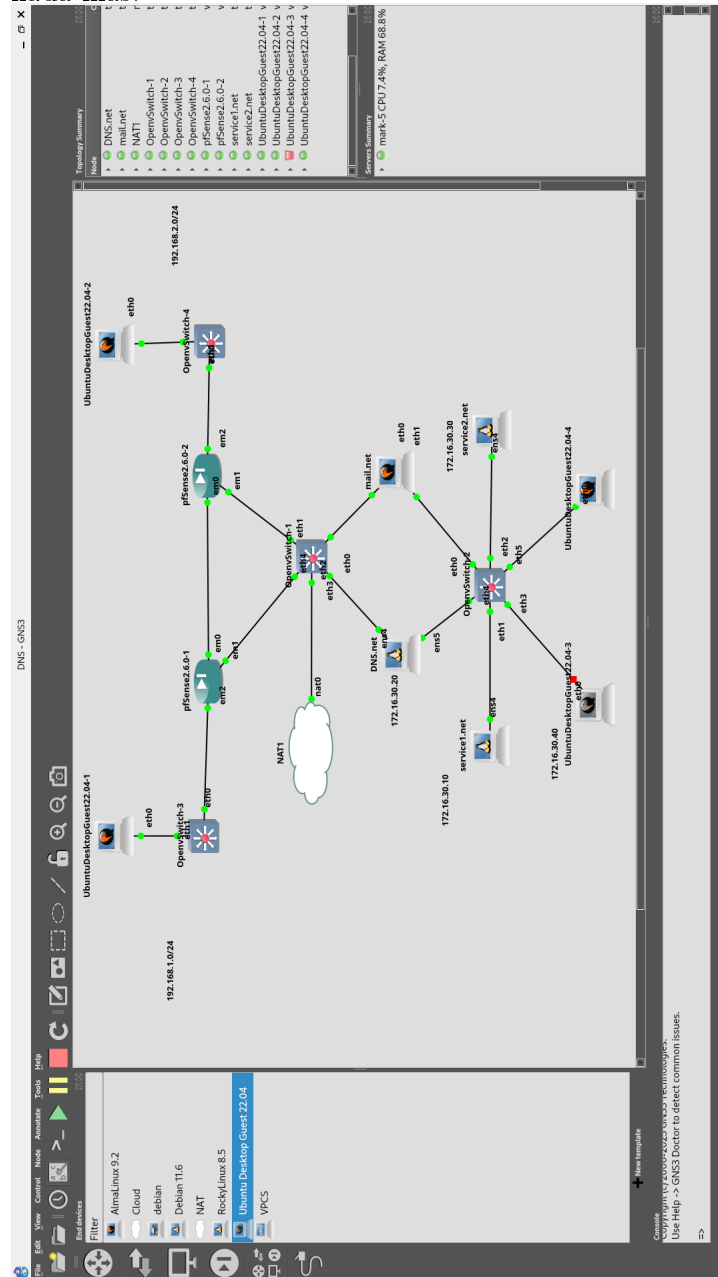
3 de septiembre de 2023

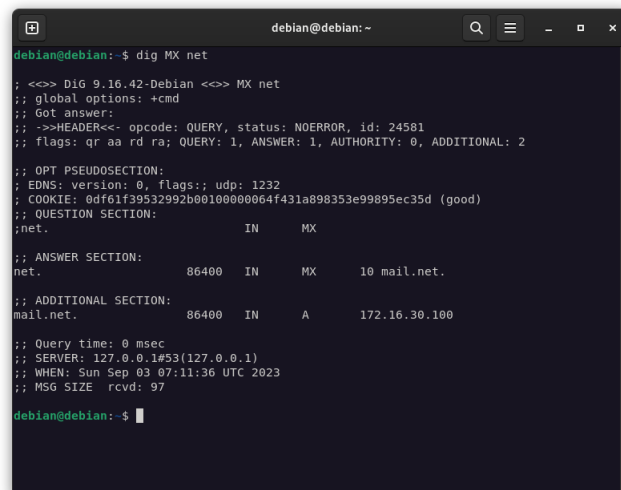
Índice

1	Topología de red	3
2	Preparación de DNS	4
3	Instalación de Postfix MTA	5
3.1	TLS en Postfix	9
4	Instalación y configuración de Dovecot	13
5	Instalación de un webmail - squirrelmail	20

1. Topología de red

Se ha seguido una topología de red que nos ha permitido conectar por capa 2 los dispositivos y que estos puedan acceder a servicios de DNS compartidos sin tener que modificar nada más.





```
debian@debian:~$ dig MX net

;<<> Dig 9.16.42-Debian <<> MX net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 24581
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 0df61f39532992b00100000064f431a898353e99895ec35d (good)
;; QUESTION SECTION:
;net.                IN      MX

;; ANSWER SECTION:
net.                 86400   IN      MX      10 mail.net.

;; ADDITIONAL SECTION:
mail.net.            86400   IN      A       172.16.30.100

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Sep 03 07:11:36 UTC 2023
;; MSG SIZE rcvd: 97

debian@debian:~$
```

Figura 2: Pruebas del servicio DNS.

Todos los comandos que han sido utilizados son:

- 1 `sudo systemctl restart bind9.service`
- 2 `sudo systemctl status bind9.service`

Una vez que se han reiniciado los servicios, debemos también comprobar que se hayan cargado las zonas de DNS correctamente, para ello con el último comando que se ha indicado anteriormente, se debe comprobar que aparezca un **all zones loaded**.

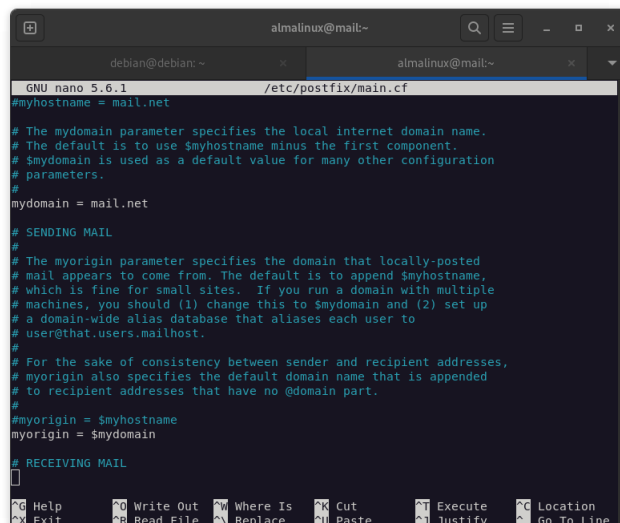
3. Instalación de Postfix MTA

Postfix es un MTA Mail Transfer Agent, que te permite el envío de correos electrónicos por el protocolo SMTP (Simple Mail Transfer Protocol). Es un MTA muy robusto que permite una amplia configuración.

Para la instalación de Postfix, hemos utilizado una distribución basada en Red Hat, para poder seguir un manual profesional de configuración del MTA. La mayoría de los cambios se han concentrado en el fichero **main.cf**, dicho fichero se localiza en **/etc/postfix/main.cf**. En la documentación de Red Hat se indica que hay que concentrarse en la configuración inicial de los siguientes parámetros del fichero de configuración, para que pueda configurarse de manera básica el MTA.

Comando para instalar Postfix en las distribuciones Red Hat:

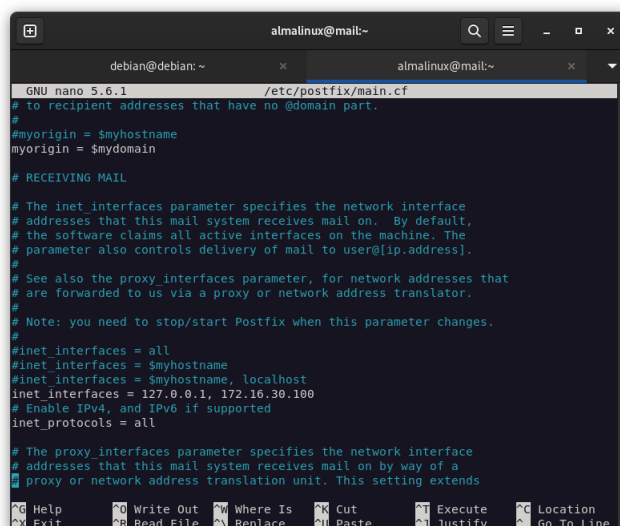
- 1 `sudo dnf update`
- 2 `sudo dnf install postfix`



```
almalinux@mail~
debian@debian: ~
GNU nano 5.6.1 /etc/postfix/main.cf
#myhostname = mail.net
# The mydomain parameter specifies the local internet domain name.
# The default is to use $myhostname minus the first component.
# $mydomain is used as a default value for many other configuration
# parameters.
#
mydomain = mail.net
# SENDING MAIL
#
# The myorigin parameter specifies the domain that locally-posted
# mail appears to come from. The default is to append $myhostname,
# which is fine for small sites. If you run a domain with multiple
# machines, you should (1) change this to $mydomain and (2) set up
# a domain-wide alias database that aliases each user to
# user@that.users.mailhost.
#
# For the sake of consistency between sender and recipient addresses,
# myorigin also specifies the default domain name that is appended
# to recipient addresses that have no @domain part.
#
myorigin = $myhostname
myorigin = $mydomain
# RECEIVING MAIL
#
# The inet_interfaces parameter specifies the network interface
# addresses that this mail system receives mail on. By default,
# the software claims all active interfaces on the machine. The
# parameter also controls delivery of mail to user[ip.address].
#
# See also the proxy_interfaces parameter, for network addresses that
# are forwarded to us via a proxy or network address translator.
#
# Note: you need to stop/start Postfix when this parameter changes.
#
inet_interfaces = all
inet_interfaces = $myhostname
inet_interfaces = $myhostname, localhost
inet_interfaces = 127.0.0.1, 172.16.30.100
# Enable IPv4, and IPv6 if supported
inet_protocols = all
# The proxy_interfaces parameter specifies the network interface
# addresses that this mail system receives mail on by way of a
# proxy or network address translation unit. This setting extends
```

Figura 3: Parámetros de main.cf - Configuración del dominio y origen.

En la captura anterior, se tiene que configurar el dominio local del servicio de correo. Puede ser mx.dominio.tld, o dominio.tld indistintamente. Luego el siguiente parámetro que hemos configurado, es el de origen, que es el parámetro utilizado para indicar el origen de mensajes enviados desde este MTA o sus usuarios.

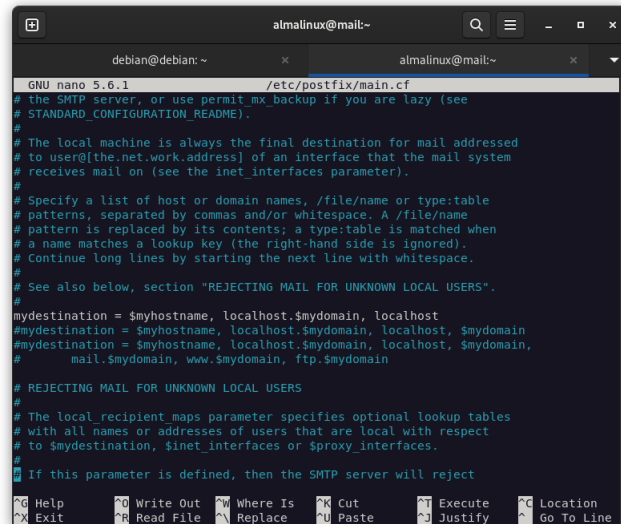


```
almalinux@mail~
debian@debian: ~
GNU nano 5.6.1 /etc/postfix/main.cf
# to recipient addresses that have no @domain part.
#
myorigin = $myhostname
myorigin = $mydomain
# RECEIVING MAIL
#
# The inet_interfaces parameter specifies the network interface
# addresses that this mail system receives mail on. By default,
# the software claims all active interfaces on the machine. The
# parameter also controls delivery of mail to user[ip.address].
#
# See also the proxy_interfaces parameter, for network addresses that
# are forwarded to us via a proxy or network address translator.
#
# Note: you need to stop/start Postfix when this parameter changes.
#
inet_interfaces = all
inet_interfaces = $myhostname
inet_interfaces = $myhostname, localhost
inet_interfaces = 127.0.0.1, 172.16.30.100
# Enable IPv4, and IPv6 if supported
inet_protocols = all
# The proxy_interfaces parameter specifies the network interface
# addresses that this mail system receives mail on by way of a
# proxy or network address translation unit. This setting extends
```

Figura 4: Parámetros de main.cf - Configuración de las interfaces de recepción de email.

Ahora tenemos que configurar las interfaces por las que Postfix recibe los correos, en el manual de administración se puede indicar un parámetro especial **all**, pero esto implica

que se ponga el correo en escucha en todas las interfaces de red indistintamente, lo cual no suele ser bueno. En su lugar, he configurado dos interfaces, en la cual una es la dirección IP externa y otra que debería de estar de manera obligatoria que es la dirección de loopback o localhost.



```
almalinux@mail~
debian@debian: ~
GNU nano 5.6.1 /etc/postfix/main.cf
# the SMTP server, or use permit_mx_backup if you are lazy (see
# STANDARD_CONFIGURATION_README).
#
# The local machine is always the final destination for mail addressed
# to user@[the.net.work.address] of an interface that the mail system
# receives mail on (see the inet_interfaces parameter).
#
# Specify a list of host or domain names, /file/name or type:table
# patterns, separated by commas and/or whitespace. A /file/name
# pattern is replaced by its contents; a type:table is matched when
# a name matches a lookup key (the right-hand side is ignored).
# Continue long lines by starting the next line with whitespace.
#
# See also below, section "REJECTING MAIL FOR UNKNOWN LOCAL USERS".
#
mydestination = $myhostname, localhost.$mydomain, localhost
mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain,
# mail.$mydomain, www.$mydomain, ftp.$mydomain
#
# REJECTING MAIL FOR UNKNOWN LOCAL USERS
#
# The local_recipient_maps parameter specifies optional lookup tables
# with all names or addresses of users that are local with respect
# to $mydestination, $inet_interfaces or $proxy_interfaces.
#
# If this parameter is defined, then the SMTP server will reject
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

Figura 5: Parámetros de main.cf - Configuración de destinatarios en el MTA.

Luego tenemos que configurar el parámetro de **mydestination**, que consiste en cómo acepta de manera interna el MTA que es un destinatario que se envía de manera local, en vez de hacer reenvío. La configuración indicada en el manual de postfix es el caso 1, donde es el caso por defecto.

Una vez terminada la configuración anterior, debemos ejecutar el comando siguiente para comprobar si la configuración del main.cf de Postfix contiene algún error.

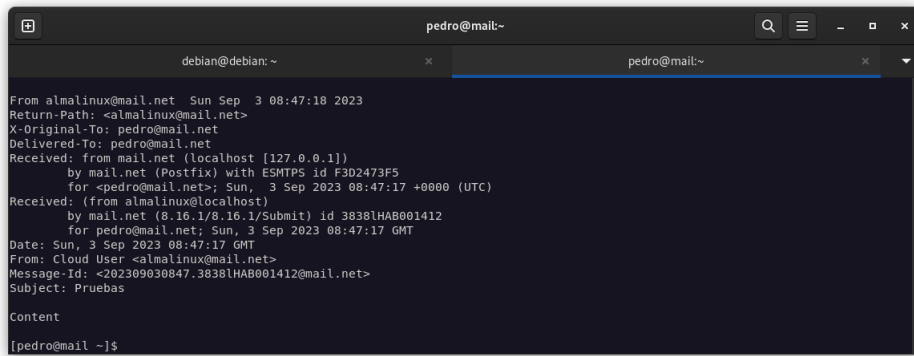
```
1 sudo postfix check
```

Luego de configurar Postfix, debemos activar y habilitar el servicio con el siguiente comando:

```
1 sudo systemctl enable --now postfix.service
2 # Verify
3 sudo systemctl status postfix
```

Luego tenemos que probar el servicio de correo, para ello instalar la siguiente utilidad que nos permite usar el MTA para enviar correos a los usuarios locales, claro que tenemos que crear previamente al usuario para poder enviarle el mail, así como para poder autenticarnos y leer su correo final en /var/log/maillog o en

```
1 sudo dnf install sendmail
2 nano mail.txt
3 sendmail pedro@mail.net < mail.txt
```

A terminal window titled 'pedro@mail~' with two tabs: 'debian@debian: ~' and 'pedro@mail~'. The active tab shows the content of an email received from 'almalinux@mail.net' on 'Sun Sep 3 08:47:18 2023'. The email headers include 'Return-Path: <almalinux@mail.net>', 'X-Original-To: pedro@mail.net', and 'Delivered-To: pedro@mail.net'. The body of the email is 'Content'.

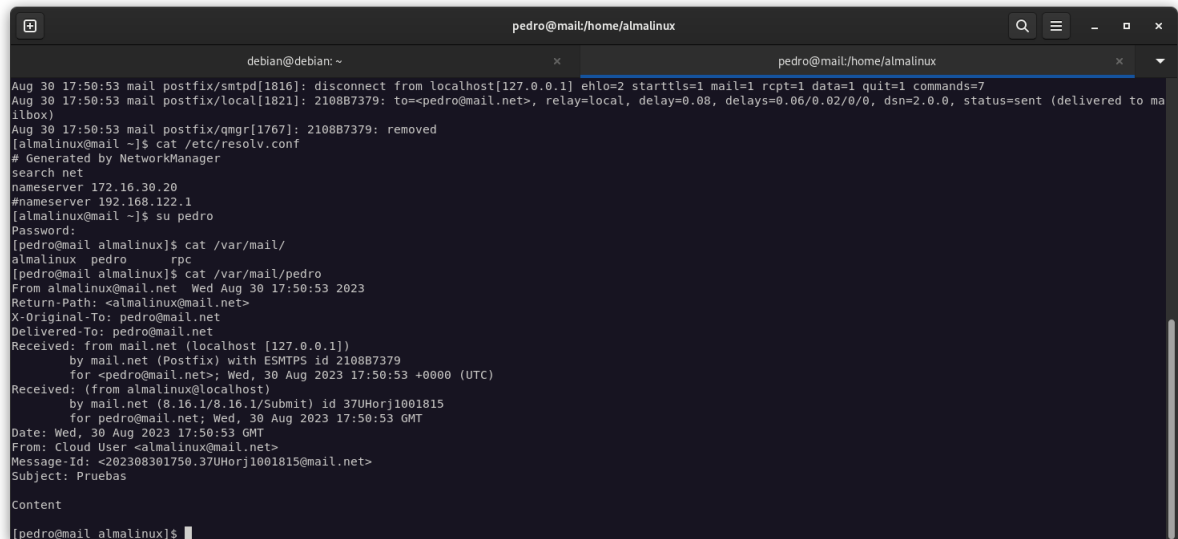
```
From almalinux@mail.net Sun Sep 3 08:47:18 2023
Return-Path: <almalinux@mail.net>
X-Original-To: pedro@mail.net
Delivered-To: pedro@mail.net
Received: from mail.net (localhost [127.0.0.1])
    by mail.net (Postfix) with ESMTPS id F3D2473F5
    for <pedro@mail.net>; Sun, 3 Sep 2023 08:47:17 +0000 (UTC)
Received: (from almalinux@localhost)
    by mail.net (8.16.1/8.16.1/Submit) id 3838LHAB001412
    for pedro@mail.net; Sun, 3 Sep 2023 08:47:17 GMT
Date: Sun, 3 Sep 2023 08:47:17 GMT
From: Cloud User <almalinux@mail.net>
Message-Id: <202309030847.3838LHAB001412@mail.net>
Subject: Pruebas

Content

[pedro@mail ~]$
```

Figura 6: Recepción del correo electrónico.

Para poder ver los correos, vamos a `/var/log/mail/$USER`.

A terminal window titled 'pedro@mail:/home/almalinux' with two tabs: 'debian@debian: ~' and 'pedro@mail:/home/almalinux'. The active tab shows the content of the file '/var/log/mail/pedro'. The log contains information about mail delivery, including a disconnect from localhost and a successful delivery to the mailbox. It also shows the user switching to 'pedro' and viewing the email content, which matches the one in Figure 6.

```
Aug 30 17:50:53 mail postfix/smtpd[1816]: disconnect from localhost[127.0.0.1] ehlo=2 starttls=1 mail=1 rcpt=1 data=1 quit=1 commands=7
Aug 30 17:50:53 mail postfix/local[1821]: 2108B7379: to=<pedro@mail.net>, relay=local, delay=0.08, delays=0.06/0.02/0/0, dsn=2.0.0, status=sent (delivered to mailbox)
Aug 30 17:50:53 mail postfix/qmgr[1767]: 2108B7379: removed
[almalinux@mail ~]$ cat /etc/resolv.conf
# Generated by NetworkManager
search net
nameserver 172.16.30.20
#nameserver 192.168.122.1
[almalinux@mail ~]$ su pedro
Password:
[pedro@mail almalinux]$ cat /var/mail/almalinux_pedro
[pedro@mail almalinux]$ cat /var/mail/pedro
From almalinux@mail.net Wed Aug 30 17:50:53 2023
Return-Path: <almalinux@mail.net>
X-Original-To: pedro@mail.net
Delivered-To: pedro@mail.net
Received: from mail.net (localhost [127.0.0.1])
    by mail.net (Postfix) with ESMTPS id 2108B7379
    for <pedro@mail.net>; Wed, 30 Aug 2023 17:50:53 +0000 (UTC)
Received: (from almalinux@localhost)
    by mail.net (8.16.1/8.16.1/Submit) id 37UH0rj1001815
    for pedro@mail.net; Wed, 30 Aug 2023 17:50:53 GMT
Date: Wed, 30 Aug 2023 17:50:53 GMT
From: Cloud User <almalinux@mail.net>
Message-Id: <202308301750.37UH0rj1001815@mail.net>
Subject: Pruebas

Content

[pedro@mail almalinux]$
```

Figura 7: Correos en `/var/log/mail/`.

Por último tenemos que configurar el **home_mailbox**, para que podamos tenerlo en el `$HOME` del usuario, sobre todo a la hora de que Dovecot pueda leer los mensajes para utilizarlos con los protocolos clientes de IMAP o POP3.

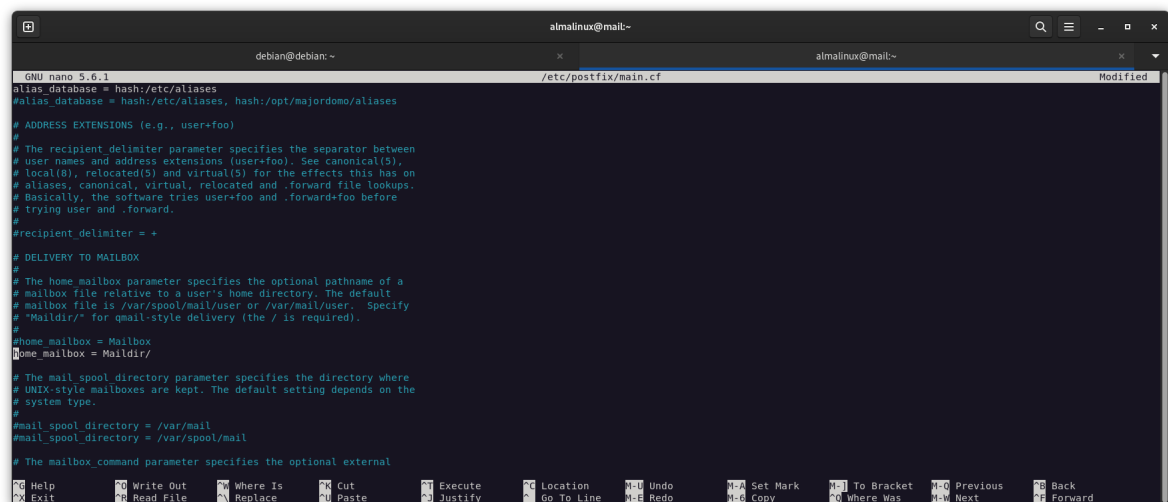


Figura 8: Configuración del home de los correos.

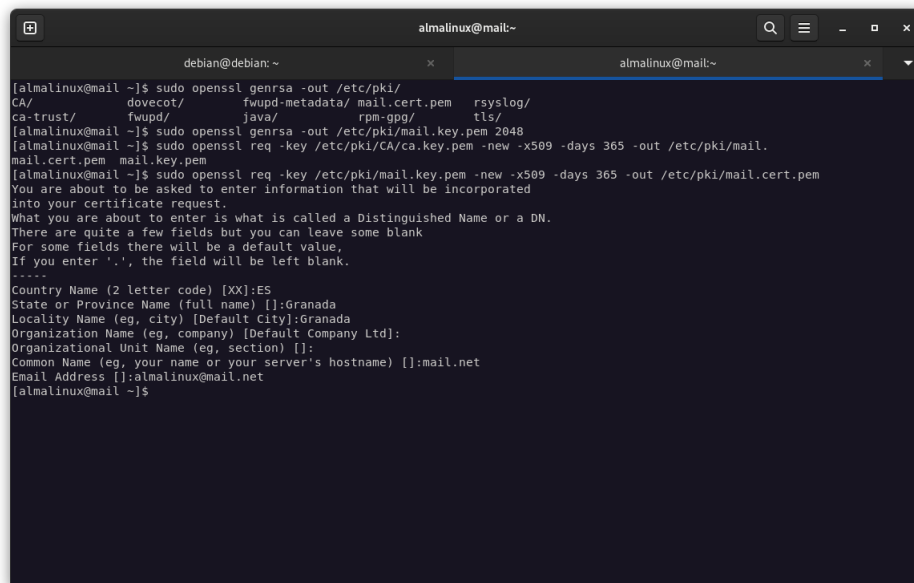
Ahora inicializamos el servicio y comprobamos si hay errores.

- 1 `sudo postfix check`
- 2 **# Habilitamos el arranque del servicio**
- 3 `sudo systemctl enable postfix.service`
- 4 `sudo systemctl start postfix.service`
- 5 **# Comprobamos si hay errores**
- 6 `sudo journalctl -xeu dovecot.service`

3.1. TLS en Postfix

Ahora tenemos que configurar TLS en el Postfix para hacer el envío y recepción seguros con TLS. Primero tenemos que generar una clave privada que nos permita el cifrado asimétrico clásico de RSA que se usa en TLS, luego tenemos que generar a partir de esa clave privada el certificado público de tipo x509 que el servidor provee a los clientes.

- 1 **# PRIVATE KEY GEN**
- 2 `sudo openssl genrsa -out /etc/pki/mail.key.pem 2048`
- 3 **# PUBLIC AND CERT x509 GEN**
- 4 `sudo openssl req -key /etc/pki/mail.key.pem -new -x509 -days 365 -out /etc/pki/mail.cert.pem`

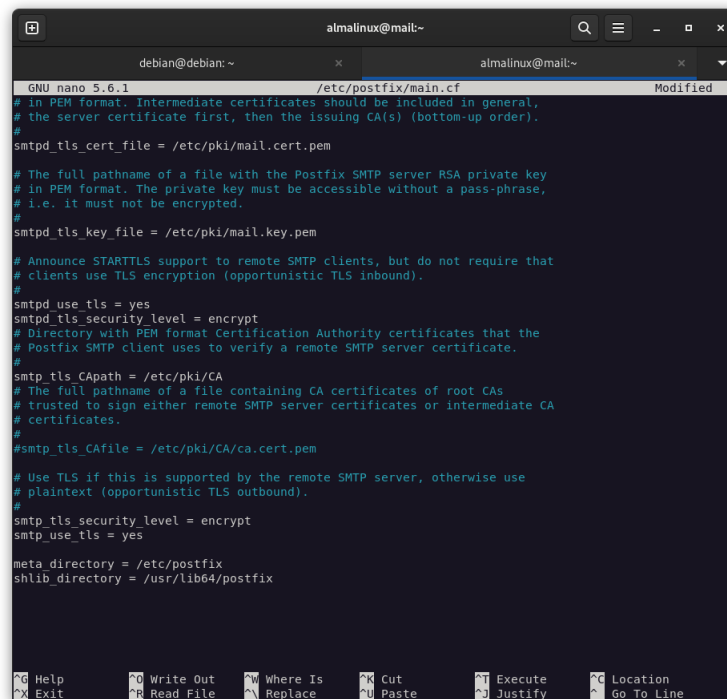
A terminal window titled 'almalinux@mail:~' with a dark background. It shows a series of commands and their outputs for generating an SSL certificate. The commands include 'openssl genrsa' to create a private key, 'openssl req' to create a certificate request, and 'openssl x509' to sign the request with a CA key. The output shows the creation of files in /etc/pki/ and prompts for DN information like Country, State, Locality, Organization, and Email Address, which are filled with default values like 'ES', 'Granada', and 'mail.net'.

```
[almalinux@mail ~]$ sudo openssl genrsa -out /etc/pki/CA/ca-trust/ dovecot/ fwupd-metadata/ mail.cert.pem rsyslog/ ca-trust/ fwupd/ java/ rpm-gpg/ tls/ [almalinux@mail ~]$ sudo openssl genrsa -out /etc/pki/mail.key.pem 2048 [almalinux@mail ~]$ sudo openssl req -key /etc/pki/CA/ca.key.pem -new -x509 -days 365 -out /etc/pki/mail.mail.cert.pem mail.key.pem [almalinux@mail ~]$ sudo openssl req -key /etc/pki/mail.key.pem -new -x509 -days 365 -out /etc/pki/mail.cert.pem You are about to be asked to enter information that will be incorporated into your certificate request. What you are about to enter is what is called a Distinguished Name or a DN. There are quite a few fields but you can leave some blank For some fields there will be a default value, If you enter '.', the field will be left blank. ----- Country Name (2 letter code) [XX]:ES State or Province Name (full name) []:Granada Locality Name (eg, city) [Default City]:Granada Organization Name (eg, company) [Default Company Ltd]: Organizational Unit Name (eg, section) []: Common Name (eg, your name or your server's hostname) []:mail.net Email Address []:almalinux@mail.net [almalinux@mail ~]$
```

Figura 9: Generación del certificado SSL para mail.net

Ahora tenemos que configurar el servidor Postfix, para que pueda utilizar correctamente el certificado, por defecto en las últimas actualizaciones al parecer usa un certificado flojo, para poder ofrecer conexiones un poco más seguras que en texto plano. Para configurar dicho certificado debemos ir a `/etc/postfix/main.cf`. Tenemos que modificar los siguientes parámetros.

Nota: Los parámetros los tenemos que indicar dos veces, debido a que uno es para la recepción y otro es para el envío. Entonces si uno de ellos se omite, ocurrirá que se envíen los datos en texto plano.



```
GNU nano 5.6.1 /etc/postfix/main.cf Modified
# in PEM format. Intermediate certificates should be included in general,
# the server certificate first, then the issuing CA(s) (bottom-up order).
#
smtpd_tls_cert_file = /etc/pki/mail.cert.pem
#
# The full pathname of a file with the Postfix SMTP server RSA private key
# in PEM format. The private key must be accessible without a pass-phrase,
# i.e. it must not be encrypted.
#
smtpd_tls_key_file = /etc/pki/mail.key.pem
#
# Announce STARTTLS support to remote SMTP clients, but do not require that
# clients use TLS encryption (opportunistic TLS inbound).
#
smtpd_use_tls = yes
smtpd_tls_security_level = encrypt
# Directory with PEM format Certification Authority certificates that the
# Postfix SMTP client uses to verify a remote SMTP server certificate.
#
smtp_tls_CApath = /etc/pki/CA
# The full pathname of a file containing CA certificates of root CAs
# trusted to sign either remote SMTP server certificates or intermediate CA
# certificates.
#
smtp_tls_CAfile = /etc/pki/CA/ca.cert.pem
#
# Use TLS if this is supported by the remote SMTP server, otherwise use
# plaintext (opportunistic TLS outbound).
#
smtp_tls_security_level = encrypt
smtp_use_tls = yes

meta_directory = /etc/postfix
shlib_directory = /usr/lib64/postfix
```

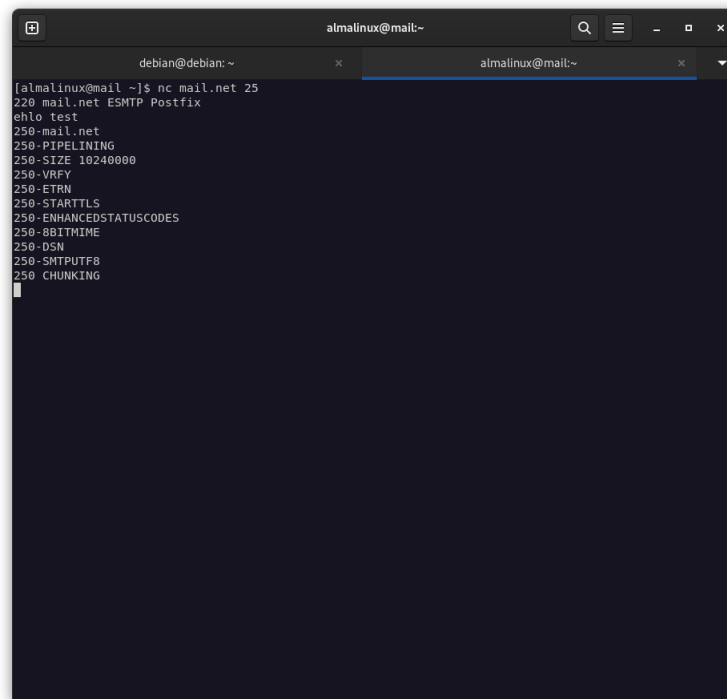
Figura 10: Configuración utilizada para STARTTLS

Ahora inicializamos el servicio y comprobamos si hay errores.

- 1 **# Comprobador de si hay fallos en los ficheros de configuracion**
- 2 `sudo postfix check`
- 3 `sudo systemctl restart postfix.service`
- 4 **# Comprobamos si hay errores adicionales**
- 5 `sudo journalctl -xeu postfix.service`

Finalmente para probar el STARTTLS, podemos instalar netcat y interactuar con el servidor para que nos pase los parámetros. Dentro de ellos, debe poner STARTTLS.

- 1 `nc mail.net 25`
- 2 `ehlo test`



The image shows a terminal window with two tabs. The active tab is titled 'almalinux@mail:~'. The terminal content shows an SMTP session initiated with 'nc mail.net 25'. The server responds with '220 mail.net ESMTP Postfix'. The client sends 'ehlo test', and the server responds with a list of supported extensions: '250-mail.net', '250-PIPELINING', '250-SIZE 10240000', '250-VERFY', '250-ETRN', '250-STARTTLS', '250-ENHANCEDSTATUSCODES', '250-8BITMIME', '250-DSN', '250-SMTPUTF8', and '250-CHUNKING'. The cursor is positioned at the end of the last line.

```
[almalinux@mail ~]$ nc mail.net 25
220 mail.net ESMTP Postfix
ehlo test
250-mail.net
250-PIPELINING
250-SIZE 10240000
250-VERFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-DSN
250-SMTPUTF8
250-CHUNKING
```

Figura 11: Pruebas de STARTTLS

4. Instalación y configuración de Dovecot

Ahora tenemos que realizar la instalación de Dovecot con el siguiente comando:

```
1 sudo dnf install dovecot
```

Los ficheros de configuración de Dovecot, se encuentran en `/etc/dovecot/dovecot.conf`, lo primero que debemos configurar son los protocolos que va a soportar. Así como las direcciones IP que debe escuchar.

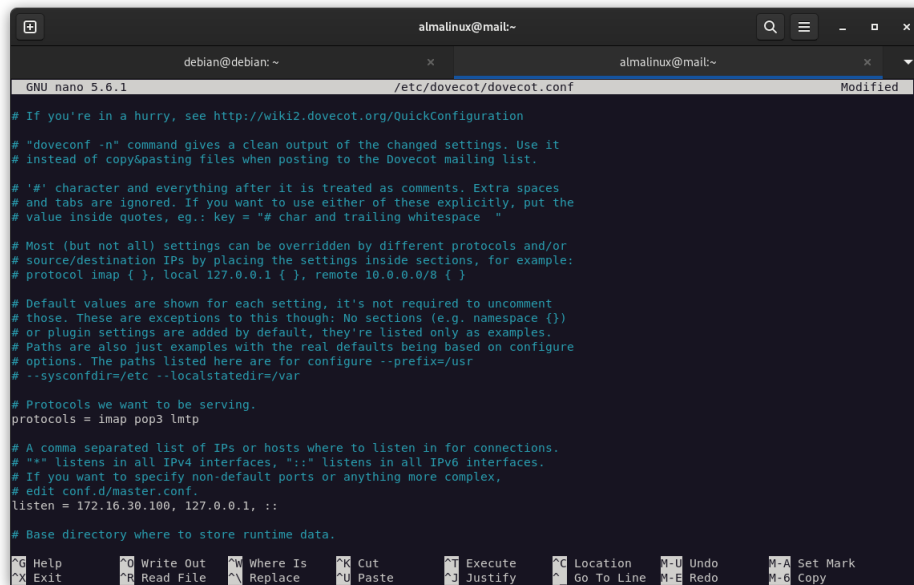
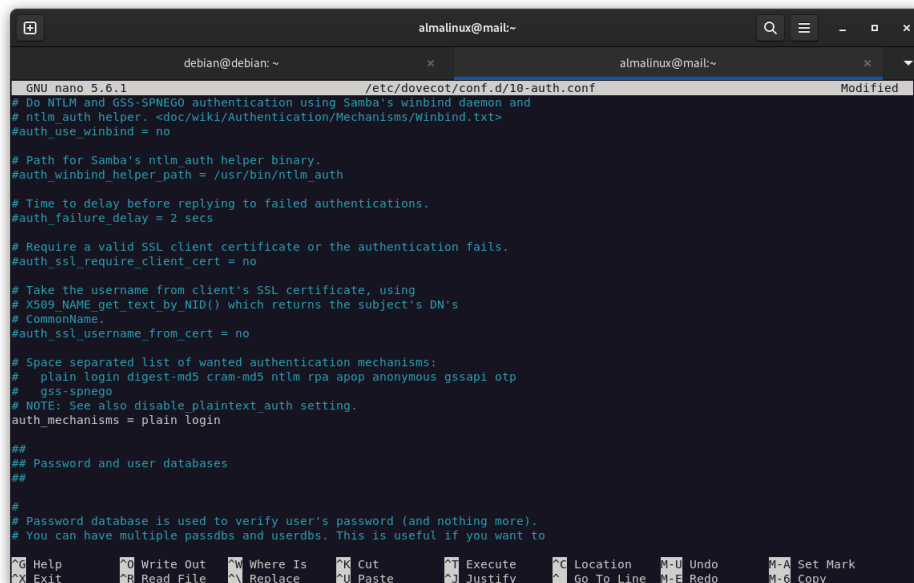


Figura 12: Configuración de protocolos de Dovecot

Luego debemos indicar el mecanismo de autenticación de login adicional al de plain. Todo eso está localizado en `/etc/dovecot/10-auth.conf`.



```
GNU nano 5.6.1 /etc/dovecot/conf.d/10-auth.conf Modified
# Do NTLM and GSS-SPNEGO authentication using Samba's winbind daemon and
# ntlm_auth helper. <doc/wiki/Authentication/Mechanisms/Winbind.txt>
#auth_use_winbind = no

# Path for Samba's ntlm auth helper binary.
#auth_winbind_helper_path = /usr/bin/ntlm_auth

# Time to delay before replying to failed authentications.
#auth_failure_delay = 2 secs

# Require a valid SSL client certificate or the authentication fails.
#auth_ssl_require_client_cert = no

# Take the username from client's SSL certificate, using
# X509_NAME_get_text_by_NID() which returns the subject's DN's
# CommonName.
#auth_ssl_username_from_cert = no

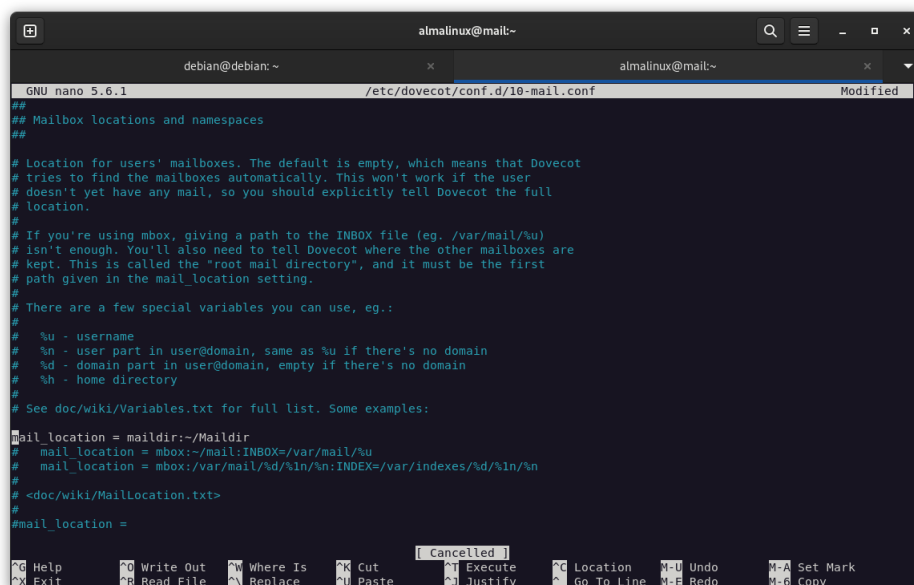
# Space separated list of wanted authentication mechanisms:
# plain login digest-md5 cram-md5 ntlm rpa apop anonymous gssapi otp
# gss-spnego
# NOTE: See also disable_plaintext_auth setting.
auth_mechanisms = plain login

##
## Password and user databases
##

# Password database is used to verify user's password (and nothing more).
# You can have multiple passdbs and userdbs. This is useful if you want to
```

Figura 13: Configuración de autenticación de Dovecot

Luego debemos configurar el directorio de la bandeja de entrada y salida del usuario. Puede estar ubicado en el **\$HOME** o en **/var/mail/\$USER**.



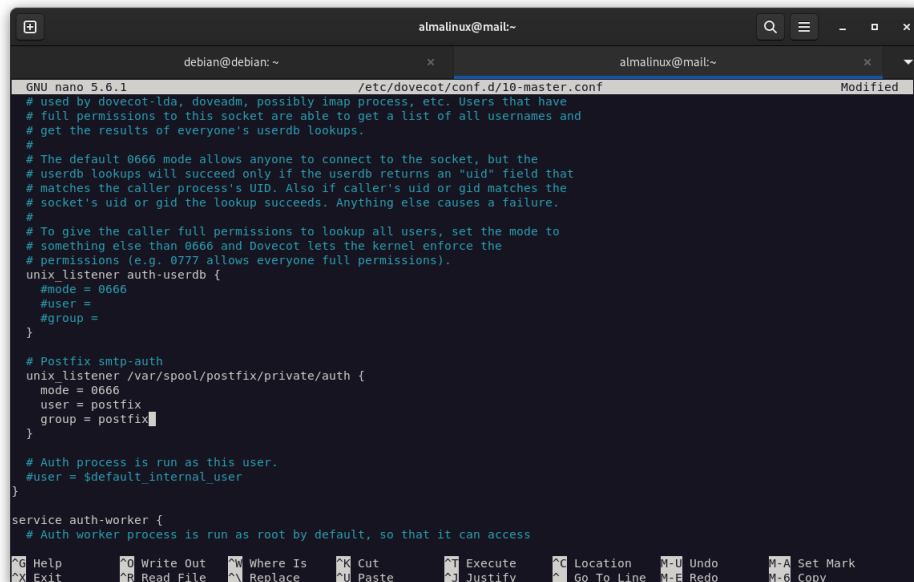
```
GNU nano 5.6.1 /etc/dovecot/conf.d/10-mail.conf Modified
## Mailbox locations and namespaces
##

# Location for users' mailboxes. The default is empty, which means that Dovecot
# tries to find the mailboxes automatically. This won't work if the user
# doesn't yet have any mail, so you should explicitly tell Dovecot the full
# location.
#
# If you're using mbox, giving a path to the INBOX file (eg. /var/mail/%u)
# isn't enough. You'll also need to tell Dovecot where the other mailboxes are
# kept. This is called the "root mail directory", and it must be the first
# path given in the mail_location setting.
#
# There are a few special variables you can use, eg.:
#
# %u - username
# %n - user part in user@domain, same as %u if there's no domain
# %d - domain part in user@domain, empty if there's no domain
# %h - home directory
#
# See doc/wiki/Variables.txt for full list. Some examples:
mail_location = maildir:~/Maildir
# mail_location = mbox:~/mail:INBOX=/var/mail/%u
# mail_location = mbox:/var/mail/%d/%n/INDEX=/var/indexes/%d/%n/%n
# <doc/wiki/MailLocation.txt>
#mail_location =
```

Figura 14: Directorio del mail

Ahora queremos configurar que Postfix, pueda usar Dovecot para la recepción de los

mensajes y que este los catalogue en los subdirectorios correspondientes. Para ello debemos contactarlos con el Socket correspondiente. El protocolo que usan es lmtmp entre ellos, se ha configurado anteriormente



```
GNU nano 5.6.1 /etc/dovecot/conf.d/10-master.conf Modified
# used by dovecot-lda, doveadm, possibly imap process, etc. Users that have
# full permissions to this socket are able to get a list of all usernames and
# get the results of everyone's userdb lookups.
#
# The default 0666 mode allows anyone to connect to the socket, but the
# userdb lookups will succeed only if the userdb returns an "uid" field that
# matches the caller process's UID. Also if caller's uid or gid matches the
# socket's uid or gid the lookup succeeds. Anything else causes a failure.
#
# To give the caller full permissions to lookup all users, set the mode to
# something else than 0666 and Dovecot lets the kernel enforce the
# permissions (e.g. 0777 allows everyone full permissions).
unix_listener auth-userdb {
  #mode = 0666
  #user =
  #group =
}

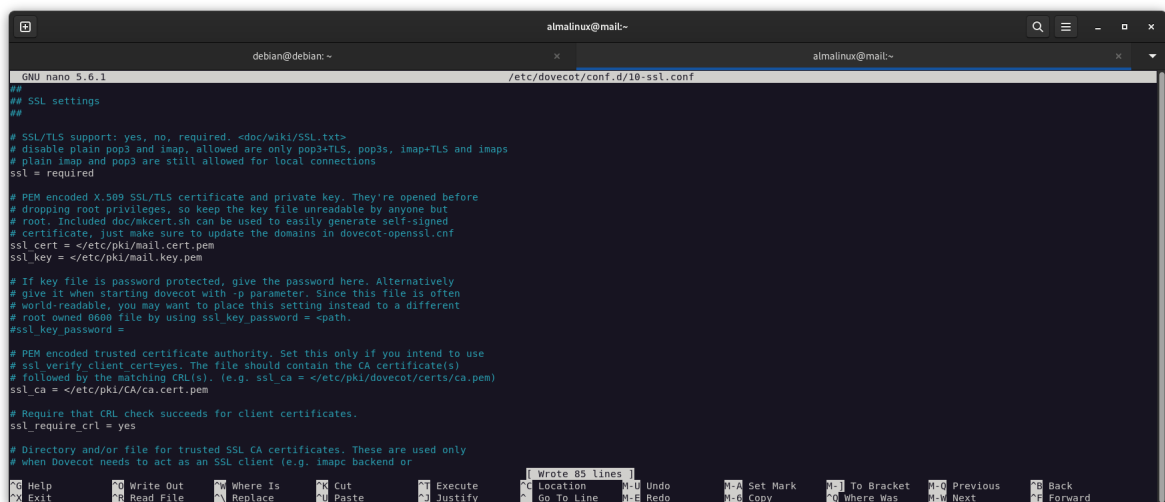
# Postfix smtp-auth
unix_listener /var/spool/postfix/private/auth {
  mode = 0666
  user = postfix
  group = postfix
}

# Auth process is run as this user.
#user = $default_internal_user
}

service auth-worker {
  # Auth worker process is run as root by default, so that it can access
```

Figura 15: Configuración del Socket

Por último configuramos los certificados compartidos con Postfix, para el uso de conexiones seguras en la ruta `/etc/dovecot/10-ssl.conf`



```
GNU nano 5.6.1 /etc/dovecot/conf.d/10-ssl.conf
##
## SSL settings
##
# SSL/TLS support: yes, no, required. <doc/wiki/SSL.txt>
# disable plain pop3 and imap, allowed are only pop3+TLS, pop3s, imap+TLS and imaps
# plain imap and pop3 are still allowed for local connections
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/pki/mail.cert.pem
ssl_key = </etc/pki/mail.key.pem

# If key file is password protected, give the password here. Alternatively
# give it when starting dovecot with -p parameter. Since this file is often
# world-readable, you may want to place this setting instead to a different
# root owned 0600 file by using ssl_key_password = <path>.
ssl_key_password =

# PEM encoded trusted certificate authority. Set this only if you intend to use
# ssl_verify_client_cert=yes. The file should contain the CA certificate(s)
# followed by the matching CRL(s). (e.g. ssl_ca = </etc/pki/dovecot/certs/ca.pem)
ssl_ca = </etc/pki/CA/ca.cert.pem

# Require that CRL check succeeds for client certificates.
ssl_require_crl = yes

# Directory and/or file for trusted SSL CA certificates. These are used only
# when Dovecot needs to act as an SSL client (e.g. imapc backend or
```

Figura 16: Configuración SSL

Ahora inicializamos el servicio y comprobamos si hay errores.

```
1 sudo systemctl enable dovecot.service
2 sudo systemctl start dovecot.service
3 # Comprobamos si hay errores
4 sudo journalctl -xeu dovecot.service
```

Ahora nos vamos a un cliente de correo (Thunderbird) y usamos IMAP siguiendo los siguientes pasos.

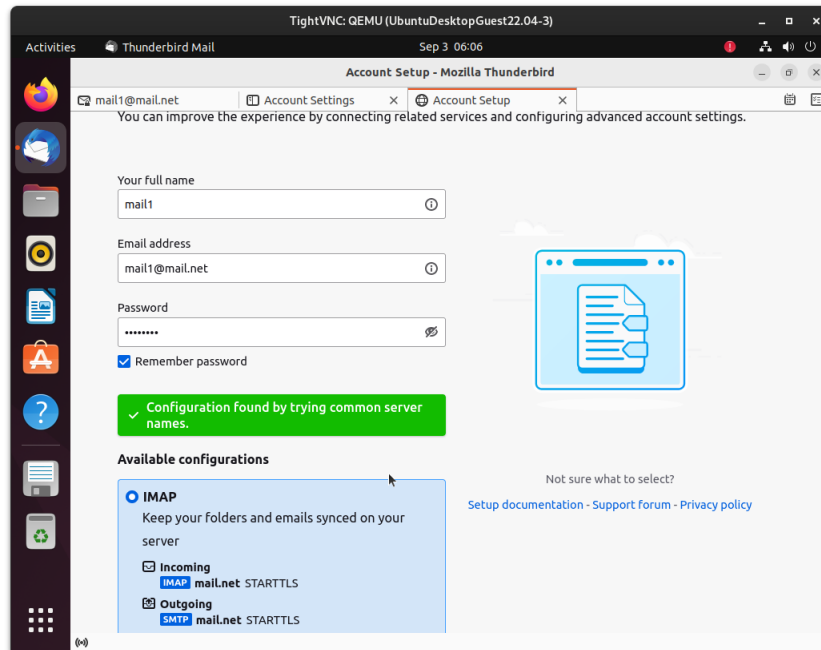


Figura 17: Introduciendo las credenciales

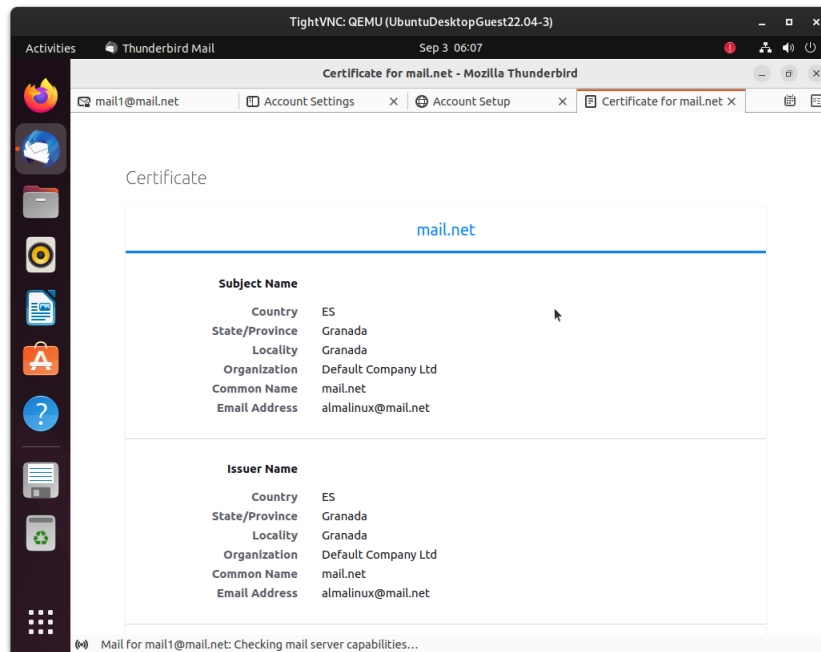


Figura 18: Verificando el certificado.

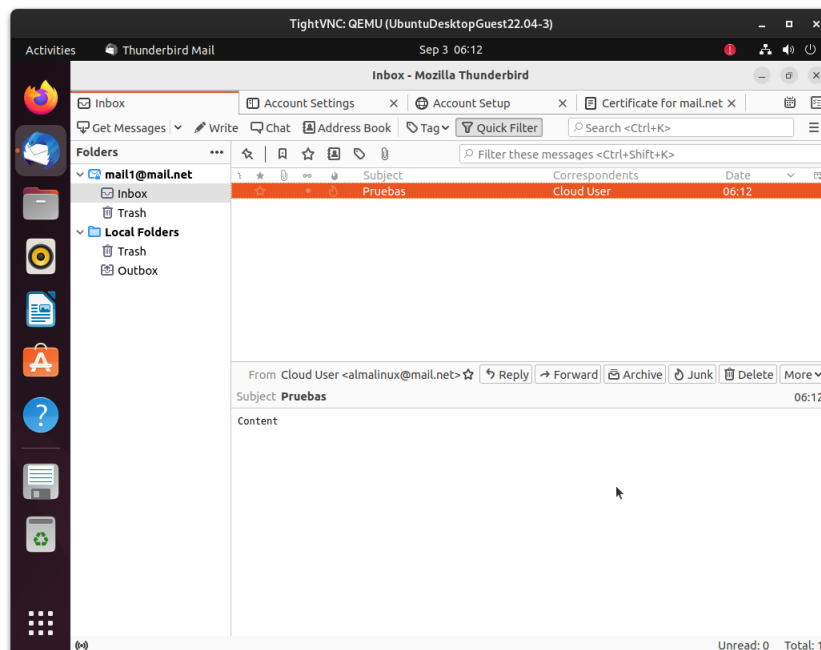


Figura 19: Recepción de mensajes.

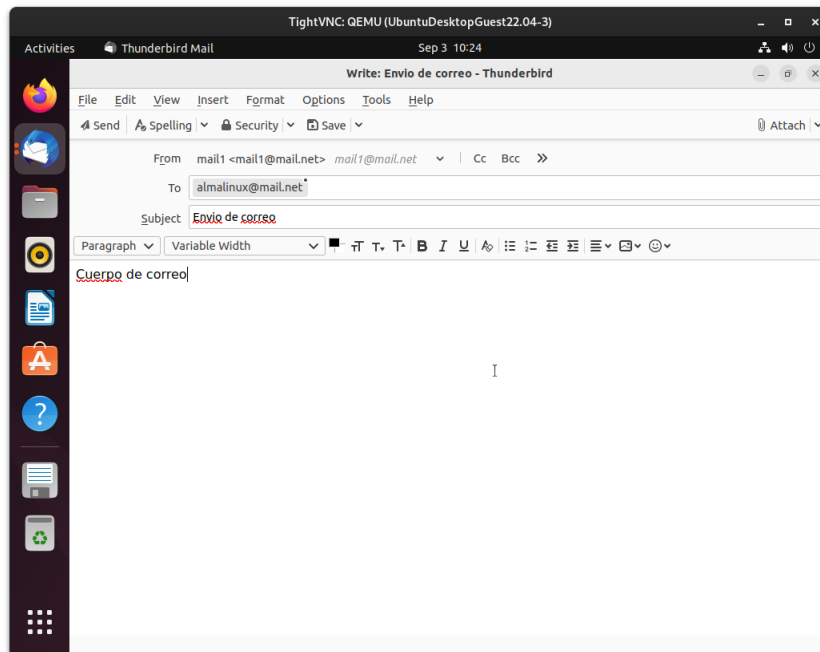


Figura 20: Envío de mensajes.

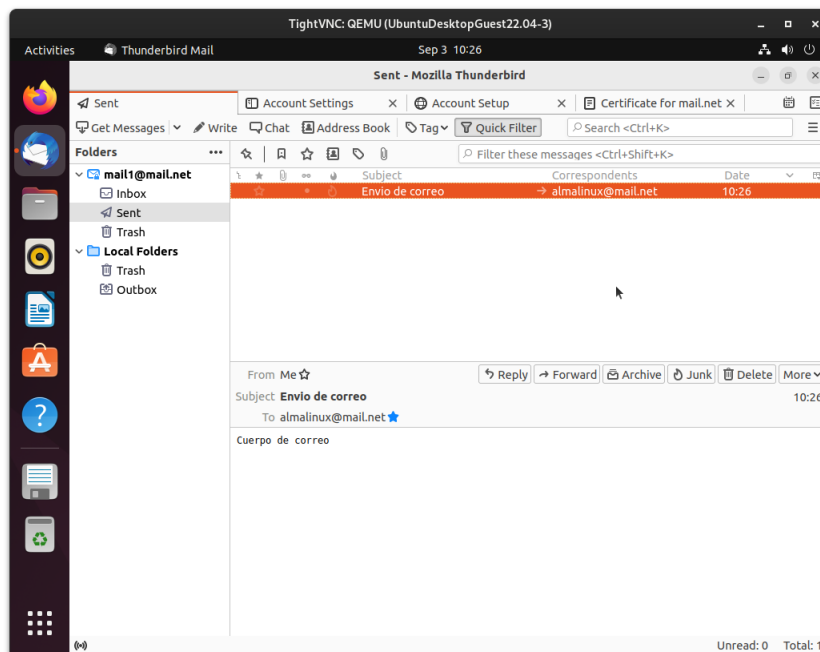


Figura 21: Envío de mensajes - parte 2.

```
alinux@mail:~$ ls
Maildir  squirrelmail-webmail-1.4.22.tar.gz  u
mail.txt squirrelmail-webmail-1.4.22.tar.gz.sha1
alinux@mail:~$ ls Maildir/
cur  new  tmp
alinux@mail:~$ ls
Maildir  squirrelmail-webmail-1.4.22.tar.gz  u
mail.txt squirrelmail-webmail-1.4.22.tar.gz.sha1
alinux@mail:~$ ls Maildir/new/
1693751167.V004110139d1M26297.mail.net
alinux@mail:~$ cat Maildir/new/
cat: Maildir/new/: Is a directory
alinux@mail:~$ cat Maildir/new/1693751167.V004110139d1M26297.mail.net
Return-Path: <mail@mail.net>
X-Original-To: alinux@mail.net
Delivered-To: alinux@mail.net
Received: from [172.16.30.40] (unknown [172.16.30.40])
    by mail.net (Postfix) with ESMTPS id E2440BFE8
    for <alinux@mail.net>; Sun,  3 Sep 2023 14:26:06 +0000 (UTC)
Message-ID: <72af37d4-1978-6647-797c-ecb67c9a6dc4@mail.net>
Date: Sun, 3 Sep 2023 10:26:03 -0400
MIME-Version: 1.0
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101
    Thunderbird/91.8.0
Content-Language: en-US
To: alinux@mail.net
From: mail1 <mail@mail.net>
Subject: Envio de correo
Content-Type: text/plain; charset=UTF-8; format=flowed
Content-Transfer-Encoding: 7bit

Cuerpo de correo

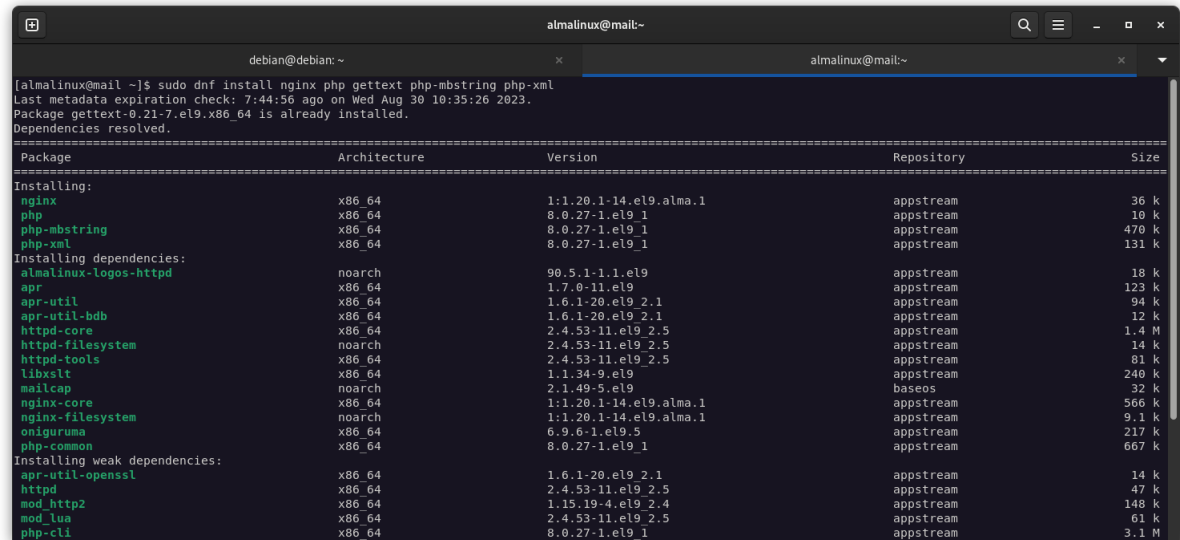
alinux@mail:~$
```

Figura 22: Envío de mensajes - parte 3.

5. Instalación de un webmail - squirrelmail

Para la instalación de Squirrelmail hemos utilizado de base Nginx con php-fpm, los paquetes que he utilizado para preparar el entorno son:

```
1 sudo dnf install nginx php gettext php-mbstring php-xml
```



```
[almalinux@mail~]$ sudo dnf install nginx php gettext php-mbstring php-xml
Last metadata expiration check: 7:44:56 ago on Wed Aug 30 10:35:26 2023.
Package gettext-0.21-7.el9.x86_64 is already installed.
Dependencies resolved.
=====
Package                                Architecture      Version            Repository          Size
=====
Installing:
nginx                                   x86_64            1:1.20.1-14.el9.almalinux-1  appstream           36 k
php                                     x86_64            8.0.27-1.el9_1         appstream           10 k
php-mbstring                           x86_64            8.0.27-1.el9_1         appstream          470 k
php-xml                                 x86_64            8.0.27-1.el9_1         appstream          131 k
Installing dependencies:
almalinux-logos-httpd                  noarch            90.5.1-1.1.el9         appstream           18 k
apr                                     x86_64            1.7.0-11.el9           appstream          123 k
apr-util                               x86_64            1.6.1-20.el9_2.1       appstream           94 k
apr-util-bdb                           x86_64            1.6.1-20.el9_2.1       appstream           12 k
httpd-core                             x86_64            2.4.53-11.el9_2.5      appstream          1.4 M
httpd-filesystem                       noarch            2.4.53-11.el9_2.5      appstream           14 k
httpd-tools                            x86_64            2.4.53-11.el9_2.5      appstream           81 k
libxslt                                x86_64            1.1.34-9.el9           appstream          240 k
mailcap                                 noarch            2.1.49-5.el9           baseos              32 k
nginx-core                             x86_64            1:1.20.1-14.el9.almalinux-1  appstream          566 k
nginx-filesystem                       noarch            1:1.20.1-14.el9.almalinux-1  appstream           9.1 k
oniguruma                              x86_64            6.9.6-1.el9_5          appstream          217 k
php-common                             x86_64            8.0.27-1.el9_1         appstream          667 k
Installing weak dependencies:
apr-util-openssl                       x86_64            1.6.1-20.el9_2.1       appstream           14 k
httpd                                   x86_64            2.4.53-11.el9_2.5      appstream           47 k
mod_http2                              x86_64            1.15.19-4.el9_2.4      appstream          148 k
mod_lua                                x86_64            2.4.53-11.el9_2.5      appstream           61 k
php-cli                                 x86_64            8.0.27-1.el9_1         appstream          3.1 M
```

Figura 23: Instalación del servicio web.

Ahora configuramos la ruta de la carpeta junto al nombre de dominio que va a tener el servicio web. Dicho nombre de dominio es el mismo que va a tener el servidor de correo.

```

GNU nano 5.6.1 /etc/nginx/nginx.conf

# Load modular configuration files from the /etc/nginx/conf.d directory.
# See http://nginx.org/en/docs/nginx_core_module.html#include
# for more information.
include /etc/nginx/conf.d/*.conf;

server {
    listen      80;
    listen      [::]:80;
    server_name mail.net;
    root        /var/www/squirrelmail;

    # Load configuration files for the default server block.
    include /etc/nginx/default.d/*.conf;

    error_page 404 /404.html;
    location = /404.html {
    }

    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
    }
}

# Settings for a TLS enabled server.
#
# server {
#     listen      443 ssl http2;

```

Figura 24: Configuración de NGINX.

Ahora para configurar squirrelmail, aparte de descargarnos los ficheros desde su web, tenemos que instalar perl, porque tiene un script que te ayuda a crear el fichero de configuración que necesita el servicio.

```

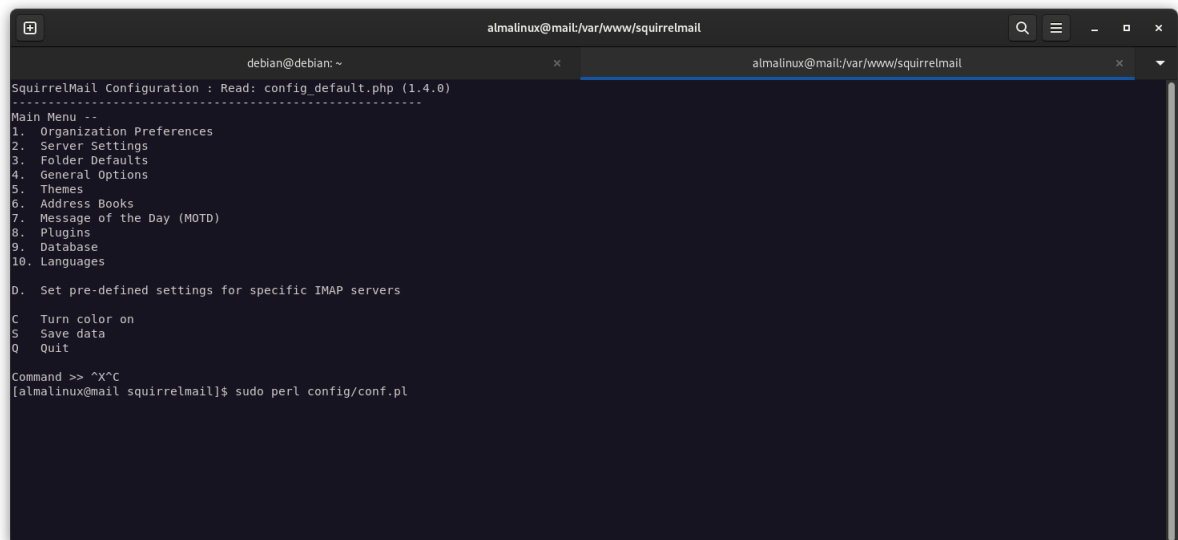
[almalinux@mail squirrelmail]$ sudo dnf install perl
Last metadata expiration check: 7:57:44 ago on Wed Aug 30 10:35:26 2023.
Dependencies resolved.
=====
Package                                Architecture      Version           Repository        Size
=====
Installing:
perl                                   x86_64            4:5.32.1-480.el9 appstream         13 k
Installing dependencies:
annobin                               x86_64            11.05-1.el9       appstream         972 k
binutils                              x86_64            2.35.2-37.el9     baseos            4.5 M
binutils-gold                         x86_64            2.35.2-37.el9     baseos            734 k
cpp                                    x86_64            11.3.1-4.3.el9.alma appstream         11 M
dwz                                    x86_64            0.14-3.el9        appstream         127 k
elfi-srpm-macros                      noarch            6-2.el9_0.0.1     appstream         21 k
elfutils-debuginfod-client            x86_64            0.188-3.el9       baseos            37 k
fonts-srpm-macros                     noarch            1:2.0.5-7.el9.1   appstream         27 k
gcc                                    x86_64            11.3.1-4.3.el9.alma appstream         32 M
gcc-c++                              x86_64            11.3.1-4.3.el9.alma appstream         13 M
gcc-plugin-annobin                    x86_64            11.3.1-4.3.el9.alma appstream         50 k
glibc-devel                           x86_64            1.5.0-6.el9       appstream         7.8 k
glibc-headers                         x86_64            2.34-60.el9       appstream         48 k
go-srpm-macros                        noarch            3.2.0-1.el9       appstream         26 k
kernel-headers                        x86_64            5.14.0-284.25.1.el9_2 appstream         4.8 M
kernel-srpm-macros                    noarch            1.0-12.el9        appstream         15 k
libdatrie                             x86_64            0.2.13-4.el9      appstream         32 k
libmpc                                x86_64            1.2.1-4.el9       appstream         61 k
libpkgconf                            x86_64            1.7.3-10.el9      baseos            35 k
libstdc++-devel                       x86_64            11.3.1-4.3.el9.alma appstream         2.2 M
libthai                               x86_64            0.1.28-8.el9      appstream         288 k
libxcrypt-devel                       x86_64            4.4.18-3.el9      appstream         28 k
=====

```

Figura 25: Instalación de perl.

- 1 sudo dnf install perl
- 2 # Luego de instalar , en /var/www/squirrelmail
- 3 sudo perl config/conf.pl

Ahora lanzamos el menú de configuración.



```
almalinux@mail:/var/www/squirrelmail
SquirrelMail Configuration : Read: config_default.php (1.4.0)
-----
Main Menu --
1. Organization Preferences
2. Server Settings
3. Folder Defaults
4. General Options
5. Themes
6. Address Books
7. Message of the Day (MOTD)
8. Plugins
9. Database
10. Languages

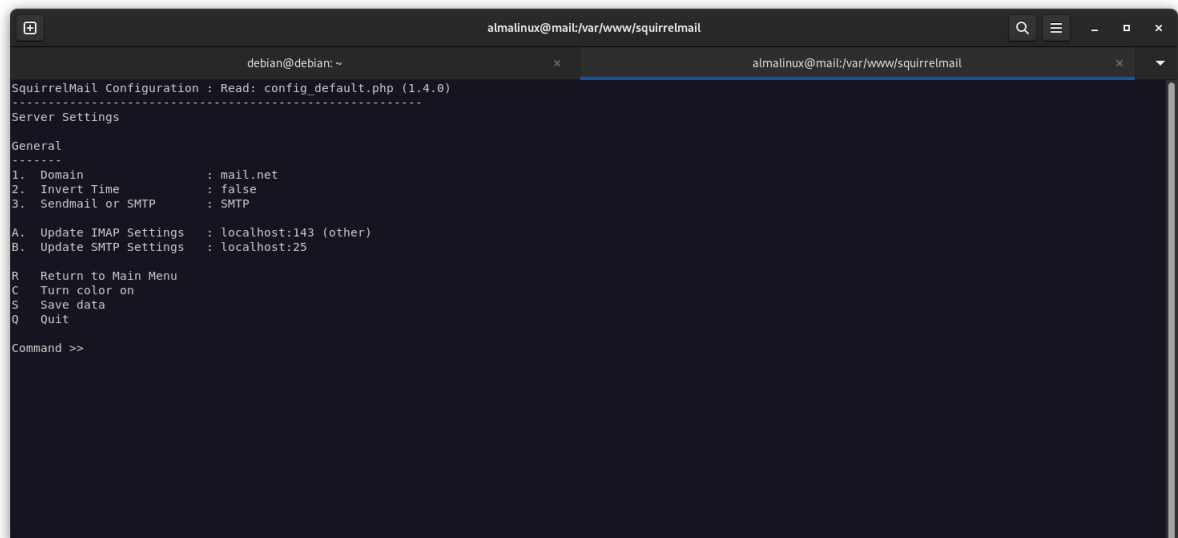
D. Set pre-defined settings for specific IMAP servers

C Turn color on
S Save data
Q Quit

Command >> ^X^C
[almalinux@mail squirrelmail]$ sudo perl config/conf.pl
```

Figura 26: Configurador de squirrelmail.

Nos vamos a la opción 2 para configurar el servidor web. Cambiamos el nombre de dominio y el resto lo dejamos en localhost.



```
almalinux@mail:/var/www/squirrelmail
SquirrelMail Configuration : Read: config_default.php (1.4.0)
-----
Server Settings

General
-----
1. Domain           : mail.net
2. Invert Time      : false
3. Sendmail or SMTP : SMTP

A. Update IMAP Settings : localhost:143 (other)
B. Update SMTP Settings : localhost:25

R Return to Main Menu
C Turn color on
S Save data
Q Quit

Command >>
```

Figura 27: Configurador de squirrelmail - Opción 2.

Finalmente ejecutamos los siguientes comandos al terminar de ejecutar el script configurador, esto es necesario para generar el correcto funcionamiento del servicio web que

necesita unos directorios especiales.

```
1 # Necesita que las carpetas las tenga el mismo usuario que el servicio web
2 sudo chown -R nginx:nginx /var/www/squirrelmail/
3 sudo mkdir -p /var/local/squirrelmail/data
4 sudo chown -R nginx:nginx /var/local/squirrelmail/
5 # Esto es necesario para permitir la conectividad con IMAP
6 sudo setsebool httpd_can_network_connect=1
```

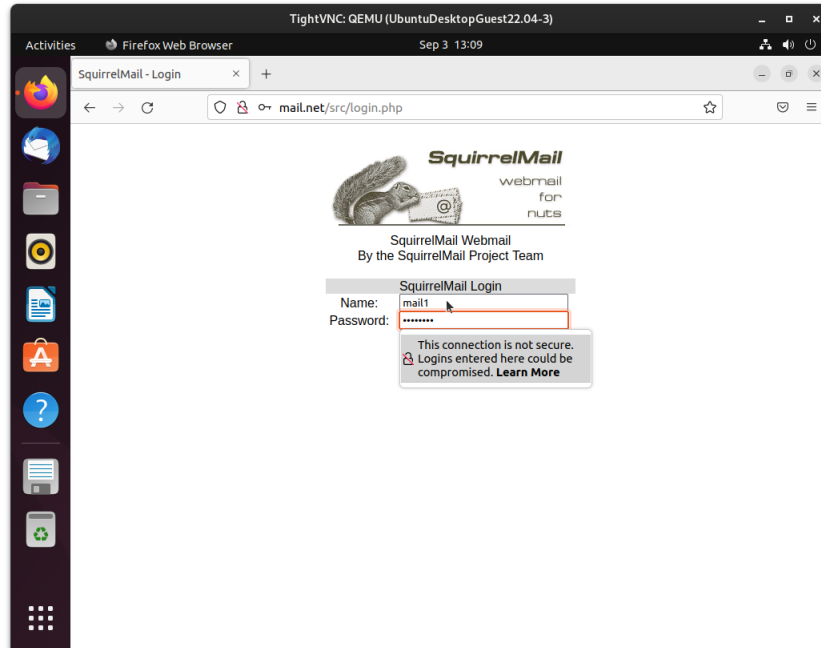


Figura 28: Servicio web activado.

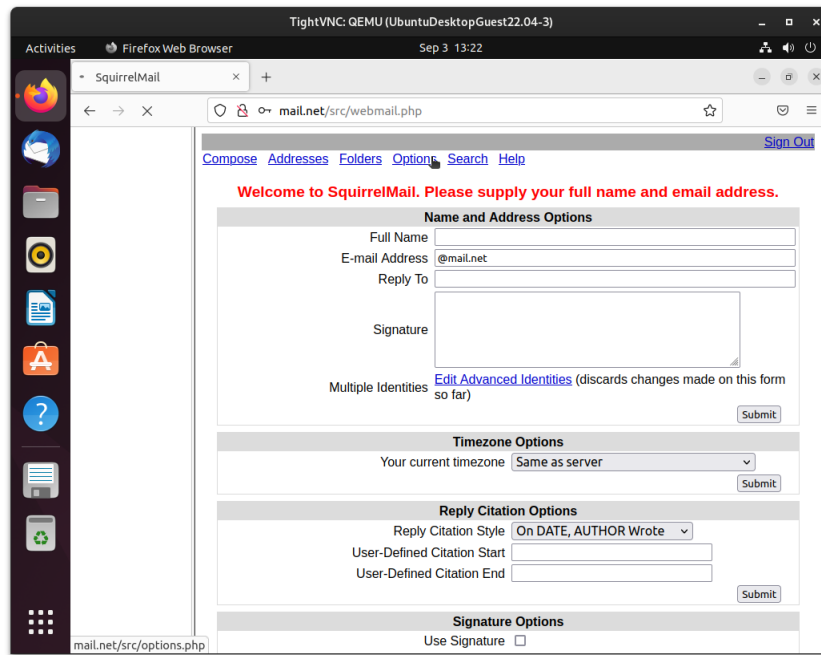
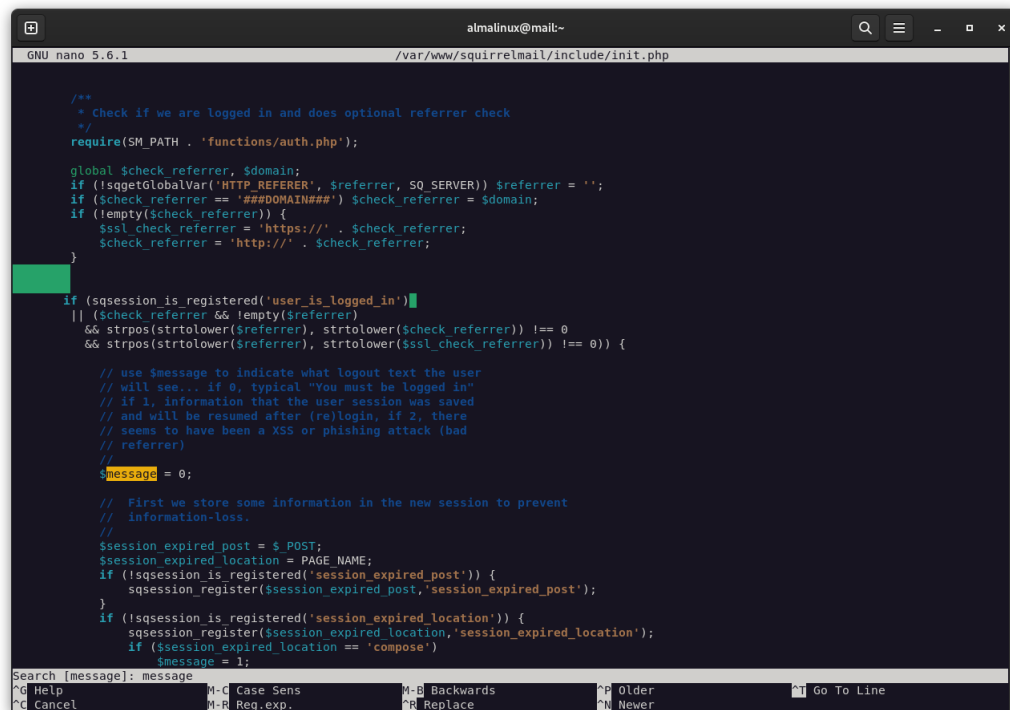


Figura 29: Servicio web activado - Parte 2.

Notas: Debido a la versión de PHP, algunas funciones no funcionan como deben debido a lo obsoleto del software, por lo que se ha modificado el siguiente archivo la parte del if (donde se ha quitado un !), para que pueda funcionar y verse el menú. Pero sigue teniendo un estado erróneo. En concreto no te deja autenticarte como usuario, quedándose en un bucle donde no puedes salir del menú de autenticación.



```
almailinux@mail~
GNU nano 5.6.1 /var/www/squirrelmail/include/init.php

/**
 * Check if we are logged in and does optional referrer check
 */
require(SM_PATH . 'functions/auth.php');

global $check_referrer, $domain;
if (!isset($_GET['HTTP_REFERER'], $referrer, $_SERVER)) $referrer = '';
if ($check_referrer == '###DOMAIN###') $check_referrer = $domain;
if (empty($check_referrer)) {
    $ssl_check_referrer = 'https://' . $check_referrer;
    $check_referrer = 'http://' . $check_referrer;
}

if (sqsession_is_registered('user_is_logged_in')) {
    if ($check_referrer && !empty($referrer)
        && strpos(strtolower($referrer), strtolower($check_referrer)) != 0
        && strpos(strtolower($referrer), strtolower($ssl_check_referrer)) != 0) {

        // use $message to indicate what logout text the user
        // will see... if 0, typical "You must be logged in"
        // if 1, information that the user session was saved
        // and will be resumed after (re)login, if 2, there
        // seems to have been a XSS or phishing attack (bad
        // referrer)
        //
        $message = 0;

        // First we store some information in the new session to prevent
        // information-loss.
        //
        $session_expired_post = $_POST;
        $session_expired_location = PAGE_NAME;
        if (!sqsession_is_registered('session_expired_post')) {
            sqsession_register($session_expired_post, 'session_expired_post');
        }
        if (!sqsession_is_registered('session_expired_location')) {
            sqsession_register($session_expired_location, 'session_expired_location');
            if ($session_expired_location == 'compose')
                $message = 1;
        }
    }
}

Search [message]: message
^C Help      ^M-C Case Sens  ^M-B Backwards  ^M-P Older      ^M Go To Line
^C Cancel    ^M-R Reg.exp.   ^M-R Replace    ^M-N Newer
```

Figura 30: Parte que da error.