SERVIDOR CORREO CENTOS 7

Un servidor de correo electrónico es una aplicación que permite enviar y recibir mensajes de correo entre usuarios a través de la red de internet.
Para la gestión de los correos y las conexiones hay definidos una serie de protocolos, cada uno con una finalidad concreta que se explican a continuación.

- SMTP (*simple mail transfer protocol*): es el protocolo que se emplea para que dos servidores de correo intercambien mensajes. También se emplea para que los clientes envíen correos al servidor.
- POP(*post office protocol*): se usa para obtener los mensajes almacenados en el servidor y que el destinatario los pueda consultar. Actualmente se usa la versión 3, POP3.
- IMAP(internet message access protocol): tiene la misma finalidad que la de POP, pero con un funcionamiento y unas funcionalidades un poco diferente.

Así pues, un servidor de correo suele ser en realidad una combinación de dos servicios. Un servicio SMTP, que es el encargado de enviar y recibir los mensajes, y un servicio POP/IMAP, que permite a los usuarios obtener los mensajes.
Estos protocolos de correo electrónico usan los puertos:

- SMTP utiliza el puerto 25/TCP (SMTP sobre TLS utiliza el puerto 465/TCP).
- POP utiliza el puerto 110/TCP (POP sobre SSL utiliza el puerto 995/TCP).
- IMAP utiliza el puerto 143/TCP (IMAP sobre SSL utiliza el puerto 993/TCP).

Los RFC donde se define el funcionamiento básico para el protocolo SMTP son los siguientes.

- RFC 821: *Simple mail transfer protocol*.
- RFC 822: *Standard for the format of Arpa Internet text messages.*
- RFC 1334: Implications of MIME for Internet mail gateways.
- RFC 1425: *SMTP service extensions*.
- RFC 1426: *SMTP service extensions for 8-bit-MIME Transport.*
- RFC 2487: *SMTP service extensions for secure SMTP over TLS.*
- RFC 3207 *SMTP service extensions for secure SMTP over Transport.*

Los RFC para el protocolo IMAP más importante son los siguientes.

- RFC 1734: *POP3 AUTHentication command.*
- RFC 1957: *Some observations on implementations of the post office protocol*.
- RFC 1939: *Post officce protocol-Version 3.*
- RFC 2449: *POP 3 extensions mechanism*.
- RFC 2595: *Using TLS with IMAP, POP3 and ACAP.*
- RFC 5034: *The post office protocol (POP3). Simple authentication and security layer (SASL) authetication mechanism.*

Los RFC para el protocolo IMAP más importantes son los siguientes:

- RFC 3516: *IMAP4 binary content extensión*.
- RFC3503: *Message disposition notification (MDN) profile for Internet message access protocol (IMAP).*
- RFC 3502: *IMAP MULTIAPPEND extensión*
- RFC 3501: *Internet message access protocol-Version 4rev1.*
- RFC 3348: *IMAP4 child mailbox extensión.*
- RFC 3028: *Sieve:a mail filtering language*.
- RFC 2971: *IMAP4 ID extensión.*

- RFC 2595: Using TLS with IMAP, POP3 and ACAP.
- RFC 2359: *IMAP4 UIDPLUS extensión*.

Cuando un mensaje de correo electrónico es enviado a través de Internet, el remitente hace una petición al DNS en que solicita el registro MX para el dominio de destino. La consulta devuelve una lista de nombres de dominios de servidores de intercambio de correo que aceptan correo entrante para este dominio, juntamente con un número de preferencia. A continuación, el agente emisor intenta establecer una conexión SMTP hacia uno de estos servidores, empezando por el que tiene el número de preferencia más bajo y envía el mensaje al primer servidor con el cual ha podido establecer conexión.

El servidor de correo electrónico suele situarse en la red DMZ (zona desmilitarizada) para que sea visible desde internet y poder recibir los correos electrónicos.

En este caso, vamos a configurar, una máquina para gestionar todo el correo. Un ejemplo de configuración DNS podrá ser:

```
grupo0.net.              IN   MX 10     correo.grupo0.net.
correo.grupo0.net.       IN   A         10.0.99.30
smtp.grupo0.net.         IN   CNAME     correo.grupo0.net.
pop.grupo0.net.          IN   CNAME     correo.grupo0.net.
imap.grupo0.net.         IN   CNAME     correo.grupo0.net.
```

En situaciones reales i dependiendo de la envergadura de la red, suele separarse la gestión del correo en diferentes máquinas. Por ejemplo, una o dos máquinas de entrada a la red para recibir correo (SMTP) destinada al dominio y aplicarle políticas anti-spam y antivirus, una segunda máquina para gestionar el acceso a los buzones de correo (POP-IMAP) solo accesible desde la red interna y una tercera máquina para enviar correo (SMTP) sólo accesible desde la red interna. En este caso, la configuración en el DNS podrá ser:

```
grupo0.net.              IN   MX 10     mx1.grupo0.net
grupo0.net.              IN   MX 20     mx2.grupo0.net

mx1.grupo0.net.          IN   A         10.0.99.30
mx2.grupo0.net.          IN   A         10.0.99.31
pop3.grupo0.net.         IN   A         10.0.99.32
imap.grupo0.net.         IN   CNAME     pop3.grupo0.net.
smtp.grupo0.net.         IN   A         10.0.99.33
```

1. Objetivos

Mostrar cómo se instala i configura un servidor de correo electrónico. Primero se hace una configuración básica del sistema y comprobamos si funciona con un cliente de correo. Se instala tanto el servicio de envío de correo SMTP como el servicio de gestión de buzones con el POP3 y el IMAP. Después se propone una configuración del sistema más segura usando protocolo SSL convirtiendo los protocolos de recepción de correo en

SMTP-TLS y gestión de buzones en POP3-SSL y IMAP-SSL. A continuación, mostramos como hacer una instalación de un servicio de acceso al correo a través de web conocido comúnmente como correo web. Al final, veremos las tareas de resolución de incidencias consultando los registros que crea la aplicación.

2. Preparativos

Partiendo de la configuración básica. Cree la máquina mailC con la IP 10.0.99.30.
Por otro lado, abriremos una máquina DNS, por ejemplo, dnsw (Windows Server 2012 r2).
Configura para poder acceder por terminal para realizar su configuración, sólo se puede acceder vía 10.0.100.30 (VLAN GESTIÓN).

Configure en el servidor DNS, las nuevas máquinas mailc, smtp, pop3, imap.

3. Instalación del servidor SMTP (Postfix) e IMAP y POP3(Dovecot)

Deshabilitar Selinux.

vi/etc/selinux/confing→SELINUX=disabled.

Habilitar el puerto 80 para permitir servicio web Apache
firewall-cmd --permanent -add-port=80/tcp

Habilite los puertos tcp, 25,110,143,465,993,995.
Habilte el servicio telnet (testeo conexión)

Reiniciar firewall
firewall-cmd –reload

- **Instalación POSTFIX**
  yum install postfix

Configuración postfix. Abrir fichero /etc/postfix/main.cf

Encontrar y editar las siguientes líneas

- myhostname = mailc.grupo0.net (línea 75)
- mydomain = grupo0.net (línea 83)
- myorigin = $mydomain (línea 99)
- inet_interfaces = all (113)
- inet_protocols = all (119)
- Comenta la línea 164: #mydestination = $myhostname, localhost.&mydomin, localhost
- Quita comentario línea 165: mydestination = $myhostname, localhost.&mydomin, localhost, $mydomian

- mynetworks = 10.0.99.0/24, 10.0.100.0/24, 127.0.0.0/8 (línea 264)

- home_mailbox = Maildir/ (línea 419)

Salvad y salid del fichero.
Iniciamos y habilitamos el servicio
- systemctl enable postfix
- systemctl restart postfix

Testeo del correcto funcionamiento del servicio postfix.
Cree dos usuarios con su nombre y el de otro miembro de su grupo

- useradd pau –s /sbin/nologin
- passwd pau
- Introduzca contraseña
- Creamos directorio mkdir –p /home/pau/Maildir
- Indicamos propietario: chown pau:pau –R /home/pau/Maildir

- useradd andreu –s /sbin/nologin
- passwd andreu
- Introduzca contraseña.
- Creamos directorio mkdir –p /home/pau/Maildir
- Indicamos propietario: chown andreu:andreu –R /home/andreu /Maildir

Acceda al servidor vía telnet. Instale el servicio telnet (yum install –y telnet)

- telnet mailc.grupo0.net smtp (ó telnet localhost 25)

[root@mailc /]# telnet mailc.grupo0.net smtp
Trying 10.0.99.30...
Connected to mailc.grupo0.net.
Escape character is '^]'.
220 mailc.grupo0.net ESMTP Postfix
ehlo mailc.grupo0.net → Interrogemos al servidor
250-mailc.grupo0.net
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
mail from:<pau> → Escribimos quién envía mensaje
250 2.1.0 Ok
rcpt to: <andreu> → Indicamos a quién va dirigido
250 2.1.5 Ok
data → Comando para introducir el cuerpo del mensaje
354 End data with <CR><LF>.<CR><LF>
Hola Andreu estás bien de regreso de tu viaje ... → Mensaje [enter]
. → Tras finalizar mensaje . para indicar hemos terminado
250 2.0.0 Ok: queued as F026A9B5806

quit → Salimos
221 2.0.0 Bye
Connection closed by foreign host.

Navegamos en la carpeta “new” del usuario Andreu
ls /home/andreu/Maildir/new/
1475828691.Vfd00I99acadM930005.mailc.group0.net

Observamos tenemos un nuevo mensaje, que podemos abrir con el **visor vi** o con **cat**

Verificado pues que Postfix está trabajando.

- **Instalación DOVECOT**
  Dovecot es un servidor de correo “open source” de POP3 e IMAP para sistemas Unix/Linux.

  yum install dovecot

Configuración dovecot. Abrir fichero /etc/dovecot/dovecot.conf

- Quitar comentario a la línea 24
- protocols = imap pop3 lmtp

Edite /etc/dovecot/conf.d/10-mail.conf

- Quitar comentario a la línea 24
- mail_location = maildir:~/Maildir

Edite /etc/dovecot/conf.d/10-auth.conf

- Quitar comentario a la línea 10
- disable_plaintext_auth = no (impresindible para no usar SSL)
- Quitar comentario línea 100 y añadir “login”
- auth_mechanisms = plain login

Edite /etc/dovecot/conf.d/10-ssl.conf

- ssl = yes (imprescindible para no usr SSL)

Edite /etc/dovecot/conf.d/10-master.conf

- Quitar comentario a las líneas 91 y 92
- mode = 0666
- user = postfix
- group = postfix

Arrancamos servicio Dovecot.

- systemctl enable dovecot
- systemctl restart dovecot

Testeo del correcto funcionamiento del servicio dovecot, desde la máquina cliente.

- telnet pop.grupo0.net pop3 (ó 110)

telnet pop.grupo0.net pop3
user anTrying 10.0.99.30...
Connected to pop.grupo0.net.
Escape character is '^]'.
+OK Dovecot ready.
user andreu ->Usuario a leer correo recibido
+OK
pass XXXX →contraseña usuario de correo
+OK Logged in.
retr 1 → Le solicitamos la lectura del 1 (más antiguo)
+OK 446 octets
Return-Path: <pau@grupo0.net>
X-Original-To: andreu
Delivered-To: andreu@grupo0.net
Received: from mailc.grupo0.net (mailc.grupo0.net [10.0.99.30])
by mailc.grupo0.net (Postfix) with ESMTP id F026A9B5806
for <andreu>; Fri, 7 Oct 2016 10:23:06 +0200 (CEST)
Message-Id: <20161007082415.F026A9B5806@mailc.grupo0.net>
Date: Fri, 7 Oct 2016 10:23:06 +0200 (CEST)
From: pau@grupo0.net

Hola Andreu estás bien de regreso de tu viaje ...
.
quit → Salimos
+OK Logging out.
Connection closed by foreign host.

Comprobado queda que DOVECOT está funcionando.

4. **Thunberbird o Evolution**
   Cree las cuentas de correo en cualquiera de los programas indicados.

5. **Instalación de Squirrelmail**

Enviar y recibir emails mediante comandos no es fácil todo el tiempo. Es mejor hacerlo usando una consola gráfica. De esta forma podemos enviar/recibir correo usando el denominado cliente Squirrelmail vía navegador.

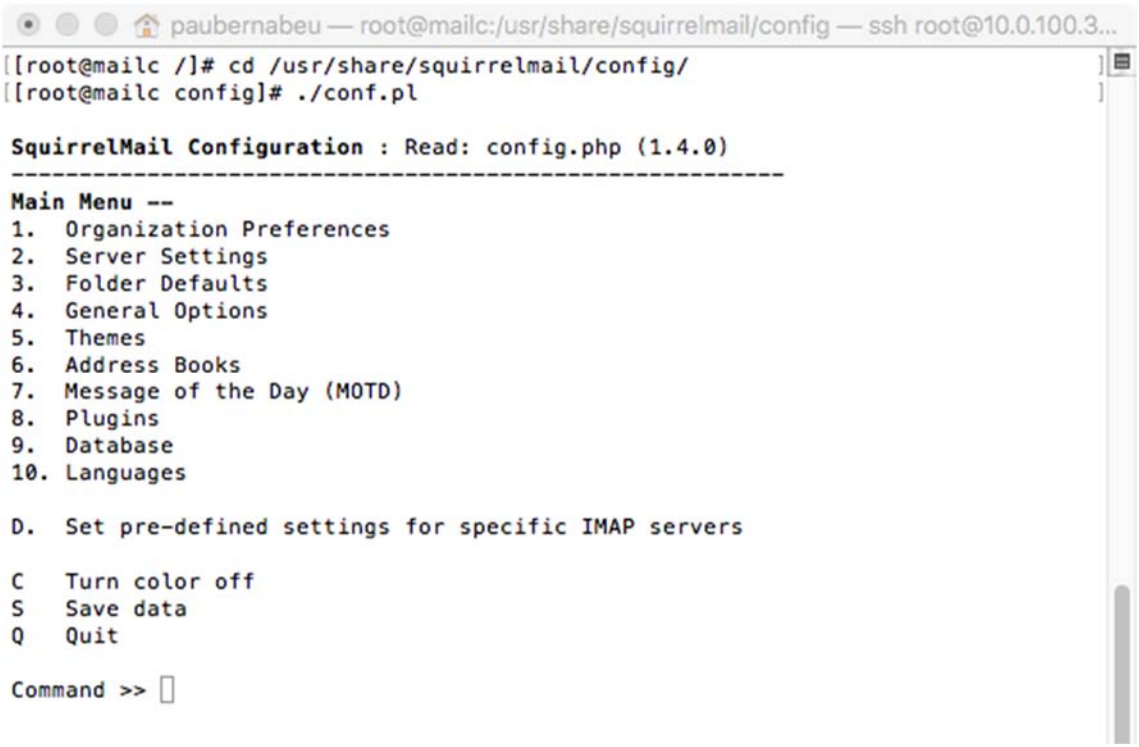
Antes de instalar, añadimos el siguiente repositorio para realizar la descarga de Squirrelmail

```
wget http://dl.fedoraproject.org/pub/epel/7/x86_64/e/epel-release-7-8.noarch.rpm

rpm -ivh epel-release-7-8.noarch.rpm
```

- yum install –y squirrelmail

Configuración: Navegar a la carpeta cd /usr/share/squirrelmail/config/

- Ejecutamos el comando: ./conf.pl



Seleccionamos comando 1

SquirrelMail Configuration : Read: config.php (1.4.0)

Organization Preferences
1. Organization Name : **SquirrelMail**
2. Organization Logo : **../images/sm_logo.png**
3. Org. Logo Width/Height : **(308/111)**
4. Organization Title : **SquirrelMail $version**
5. Signout Page :
6. Top Frame : **_top**
7. Provider link : **http://squirrelmail.org/**
8. Provider name : **SquirrelMail**

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

Command >> 1

(Cambiamos nombre organización por grupoXX)

SquirrelMail Configuration : Read: config.php (1.4.0)

Organization Preferences
1. Organization Name : **SquirrelMail**
2. Organization Logo : **../images/sm_logo.png**
3. Org. Logo Width/Height : **(308/111)**
4. Organization Title : **SquirrelMail $version**
5. Signout Page :
6. Top Frame : **_top**
7. Provider link : **http://squirrelmail.org/**
8. Provider name : **SquirrelMail**

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

Command >> **1**

We have tried to make the name SquirrelMail as transparent as
possible. If you set up an organization name, most places where
SquirrelMail would take credit will be credited to your organization.

If your Organization Name includes a '$', please precede it with a \.
Other '$' will be considered the beginning of a variable that
must be defined before the $org_name is printed.
$version, for example, is included by default, and will print the
string representing the current SquirrelMail version.

[**SquirrelMail**]: **grupo0**

Pulsamos ‘s’ para salvar y enter para continuar

Haced los cambios necesarios para que nos queden los siguientes parámetros.

SquirrelMail Configuration : Read: config.php (1.4.0)

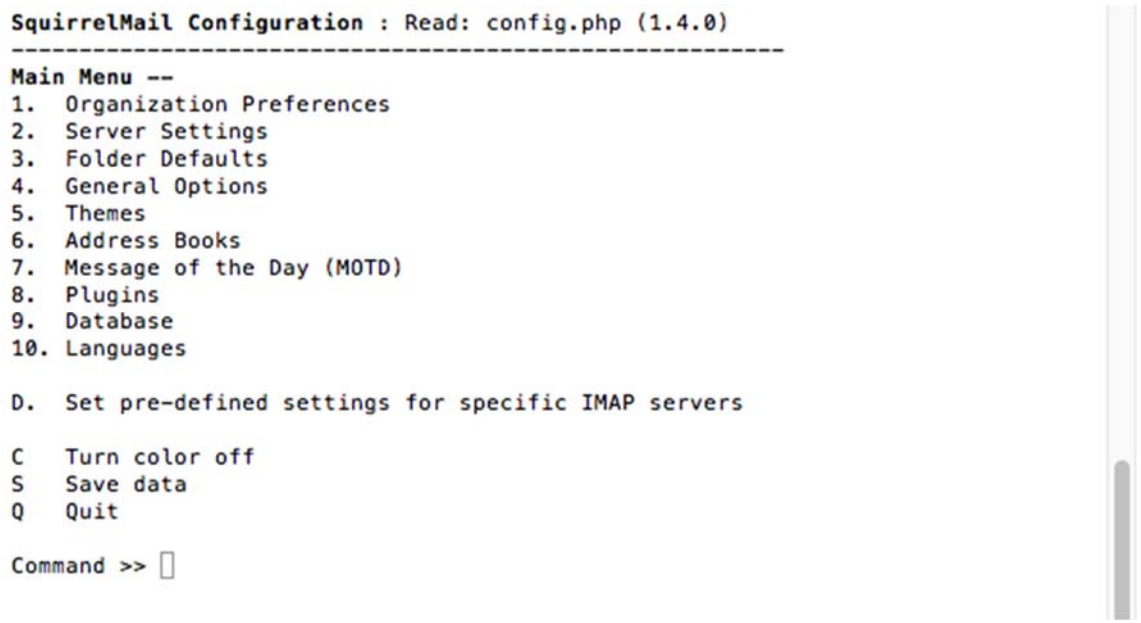
Organization Preferences
1. Organization Name : **grupo0**
2. Organization Logo : **../images/sm_logo.png**
3. Org. Logo Width/Height : **(308/111)**
4. Organization Title : **SistemasServiciosRed**
5. Signout Page :
6. Top Frame : **_top**

7. Provider link : **http://squirrelmail.org/**
8. Provider name : **Grupo0 Mail**

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

Command >>

Salvamos y salimos cuando estén hechos todos los cambios. Volvemos a la página inicial de la configuración.



Seleccionamos 2.

Proceda a realizar el cambio del dominio por grupoXX.net

SquirrelMail Configuration : Read: config.php (1.4.0)

Server Settings

General
---
1. Domain : **grupo0.net**
2. Invert Time : **false**
3. Sendmail or SMTP : **SMTP**

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

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

Command >>

Tras realizar los cambios indicados salve y salga.

Vamos ahora a crear un virtual host a squirrelmail en apache (instalado por defecto).

Para ello:

- vi /etc/httpd/conf/httpd.conf

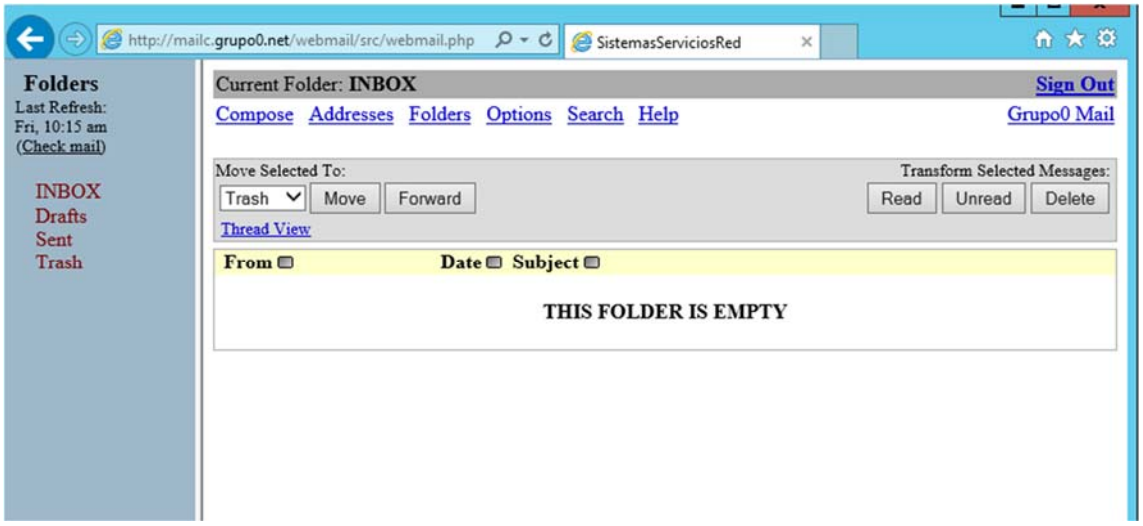
Y añadimos las siguientes líneas al final del fichero.

```
Alias /webmail /usr/share/squirrelmail

<Directory /usr/share/squirrelmail>

 Options Indexes FollowSymLinks

 RewriteEngine On

 AllowOverride All

 DirectoryIndex index.php

 Order allow,deny

 Allow from all

</Directory>
```

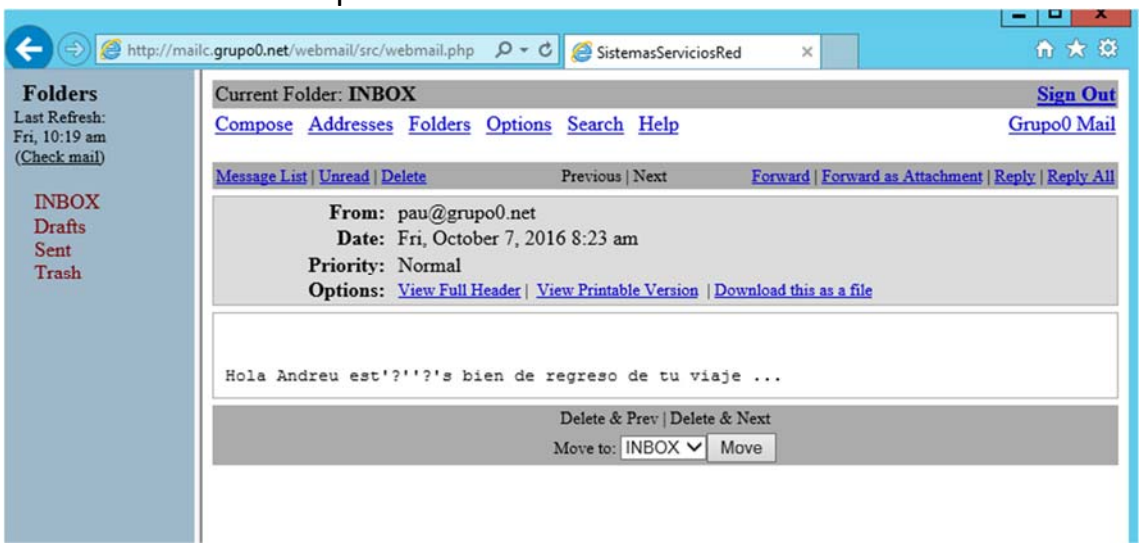
Reiniciamos el servicio Apache:

- systemctl restart httpd

Accedemos vía:: web, introducimos usuario y contraseña



Si vemos correo de Andreu, podemos leer el mensaje hemos enviado usando comandos de postfix.



6. Configuración de los servicios SMTP-TLS/POP-IMAP-SSL

La configuración de los servicios SMTP-TLS/POP-IMAP-SSL se lleva a término mediante certificados: uno para el servicio SMTP-TLS y otro par al POP-IMAP-SSL (el hecho de tener un certificado diferente para cada servicio con un nombre de máquina diferente facilita futuras migraciones de equipos).

6.1. Configuración SSL sobre dovecot (POP3-IMAP).

Editamos el fichero de configuración /etc/dovecot/conf.d/10-ssl.conf para estar seguro que la variable ssl_cipher_list está sin comentar y añadimos !SSLv3.

ssl_cipher_list = ALL:!LOW:!SSLv2:!EXP:!aNULL:!SSLv3

Este valor asegura que dovecot evite las versiones SSL 2 y 3, las cuales son consideradas inseguras. Ya que se encontraron vulnerabilidades en las mismas.

Analiza el valor a colocar del parámetro ssl en dichero fichero, analizando la siguiente argumentación o accediendo a la dirección URL http://wiki.dovecot.org/SSL/DovecotConfiguration

There are a couple of different ways to specify when SSL/TLS is required:

- `ssl=no`: SSL/TLS is completely disabled.
- `ssl=yes` and `disable_plaintext_auth=no`: SSL/TLS is offered to the client, but the client isn't required to use it. The client is allowed to login with plaintext authentication even when SSL/TLS isn't enabled on the connection. This is insecure, because the plaintext password is exposed to the internet.
- `ssl=yes` and `disable_plaintext_auth=yes`: SSL/TLS is offered to the client, but the client isn't required to use it. The client isn't allowed to use plaintext authentication, unless SSL/TLS is enabled first. However, if non-plaintext authentication mechanisms are enabled they are still allowed even without SSL/TLS. Depending on how secure they are, the authentication is either fully secure or it could have some ways for it to be attacked.
- `ssl=required`: SSL/TLS is always required, even if non-plaintext authentication mechanisms are used. Any attempt to authenticate before SSL/TLS is enabled will cause an authentication failure.
- NOTE: If you have only plaintext mechanisms enabled (e.g. `auth { mechanisms = plain login }`), `ssl=yes` and `ssl=required` are completely equivalent because in either case the authentication will fail unless SSL/TLS is enabled first.
- NOTE2: With both `ssl=yes` and `ssl=required` it's still possible that the client attempts to do a plaintext authentication before enabling SSL/TLS, which exposes the plaintext password to the internet. Dovecot attempts to indicate this to the IMAP clients via the LOGINDISABLED capability, but many clients still ignore it and send the password anyway. There is unfortunately no way for Dovecot to prevent this behavior. The POP3 standard doesn't have an equivalent capability at all, so the POP3 clients can't even know if the server would accept a plaintext authentication.
- The main difference between `ssl=required` and `disable_plaintext_auth=yes` is that if `ssl=required`, it guarantees that the entire connection is protected against eavesdropping (SSL/TLS encrypts the rest of the connection), while `disable_plaintext_auth=yes` only guarantees that the password is protected against eavesdropping (SASL mechanism is encrypted, but no SSL/TLS is necessarily used). Nowadays you most likely should be using SSL/TLS anyway for the entire connection, since the cost of SSL/TLS is cheap enough. Using both SSL/TLS and non-plaintext authentication would be the ideal situation since it protects the plaintext password even against man-in-the-middle attacks.

Note that plaintext authentication is always allowed (and SSL not required) for connections from localhost, as they're assumed to be secure anyway. This applies to all connections where the local and the remote IP addresses are equal. Also IP ranges specified by `login_trusted_networks` setting are assumed to be secure.

El siguiente paso va a ser la creación del certificado SSL para dovecot.

- Para ello editamos el fichero de configuración /etc/pki/dovecot/dovecot-openssl.cnf para colocar nuestras preferencias (valores por defecto). Sin embargo, en una instalación típica no requiere modificación.

Añadimos los campos. Usamos el * en Commom Name, para que nos sirva para las máquinas pop.grupo0.net ó imap.grupo0.net o mailc.grupo0.net.

- Renombramos, movemos o borramos los ficheros, /etc/pki/dovecot/private/dovecot.pem y /etc/pki/dovecot/certs/dovecot.pem

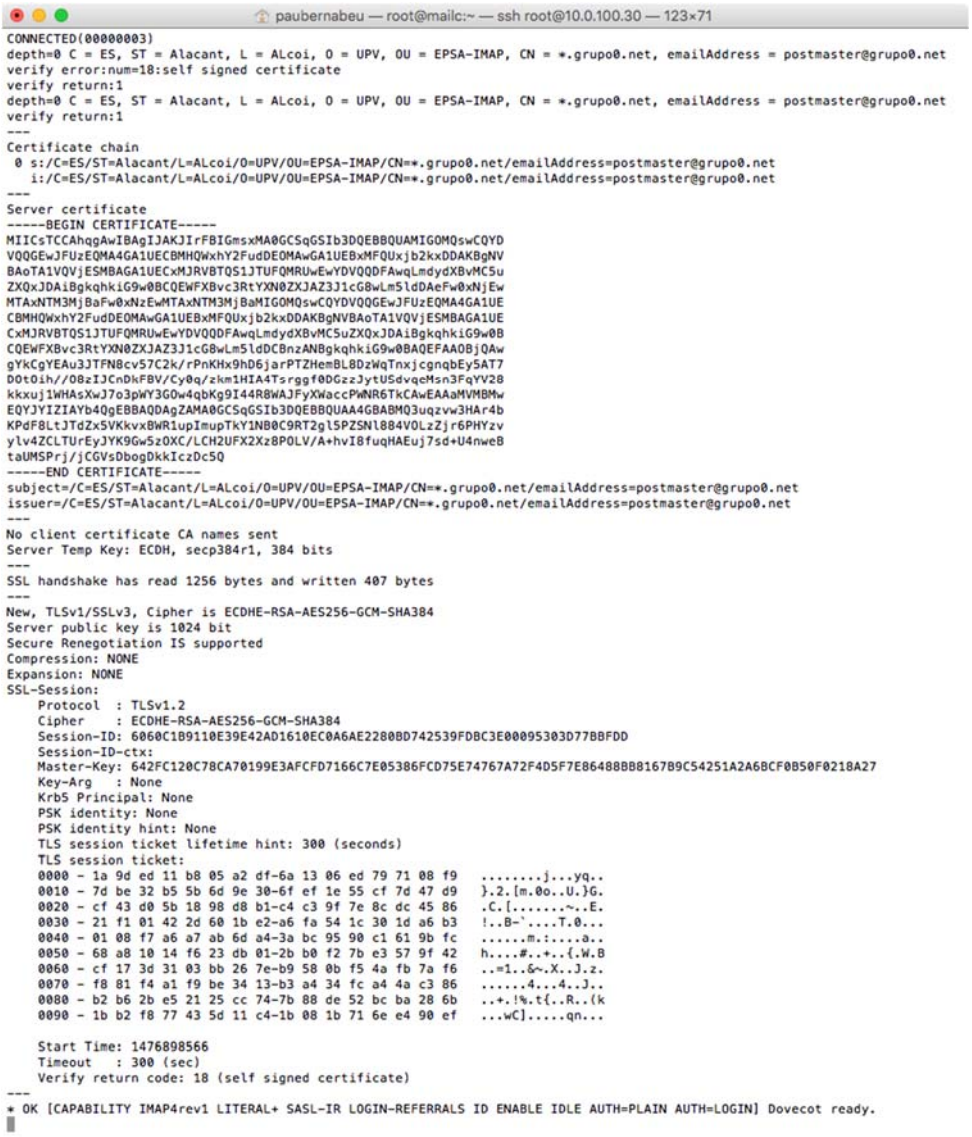
- Ejecutamos el script /usr/libexec/dovecot/mkcert.sh el cual crea el dovecot certificados autofirmados (self signed certificates). Estos certificados son copiados en : /etc/pki/dovecot/private/dovecot.pem y /etc/pki/dovecot/certs/dovecot.pem

Reiniciamos el servicio dovecot

Testeo funcionamiento, introducimos:

openssl s_client -connect imap.grupo0.net:imaps

Deberéis obtener algo así,



openssl s_client -connect imap.grupo0.net:pop3s

Deberéis obtener algo así,

6.2. Configuración TLS sobre SMTP

Creación del certificado SMTP-TLS. Conviene seguir el procedimiento seguro. Si no hacemos este paso, se usarán los certificados que se crean por defecto durante la instalación de postfix.

- mkdir –p /etc/postfix/ssl
- cd /etc/postfix/ssl

Creamos la petición de certificado dentro de este nuevo directorio:

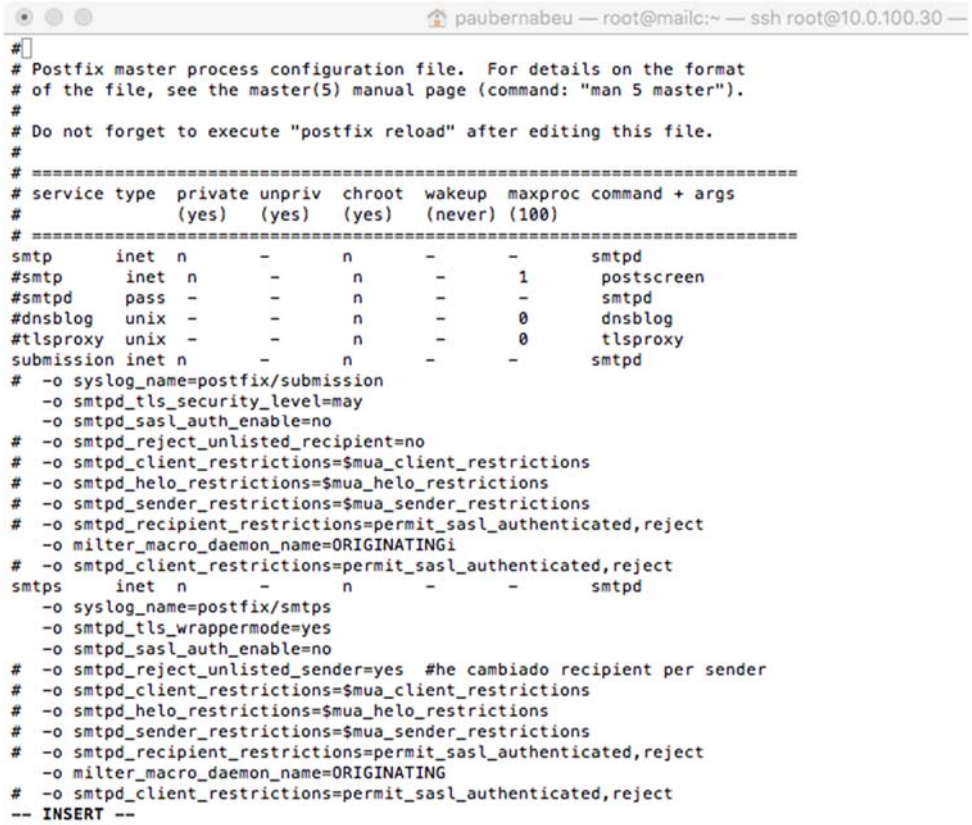
- openssl req –new –out smtp.csr –keyout smtp.key
- Rellenamos los campos. Importante Common Name **smtp.grupoXX.net**
- openssl req –x509 –in smtp.csr –key smtp.key –out smtp.csr
- openssl rsa –in smtp.key –out smtp.key
- Per seguritat chmod 400 smtp.key

Editamos fichero /etc/postfix/main.cf y añadimos al final las siguientes líneas.

```
smtpd_tls_key_file = /etc/postfix/ssl/smtp.key
smtpd_tls_cert_file = /etc/postfix/ssl/smtp.crt
smtpd_tls_security_level = may
# Sólo se utiliza si se adquiere un certificado con un CA
# smtpd_tls_CAfile = /etc/pki/tls/certs/ca-bundle.crt
# Las rutas deben corresponder a las del certificado y firma digital creados.
smtpd_tls_auth_only = yes
smtp_use_tls = yes
smtpd_use_tls = yes
smtp_tls_note_starttls_offer = yes
smtpd_tls_loglevel = 1
smtpd_tls_received_header = yes
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev:/dev/urandom
smtpd_tls_protocols = !SSLv2,!SSLv3
smtpd_tls_mandatory_protocols = !SSLv2,!SSLv3
smtp_enforce_tls = yes
smtpd_tls_session_cache_database = btree:/etc/postfix/smtpd_scache
```

Analice y entienda que es cada parámetro de la configuración.

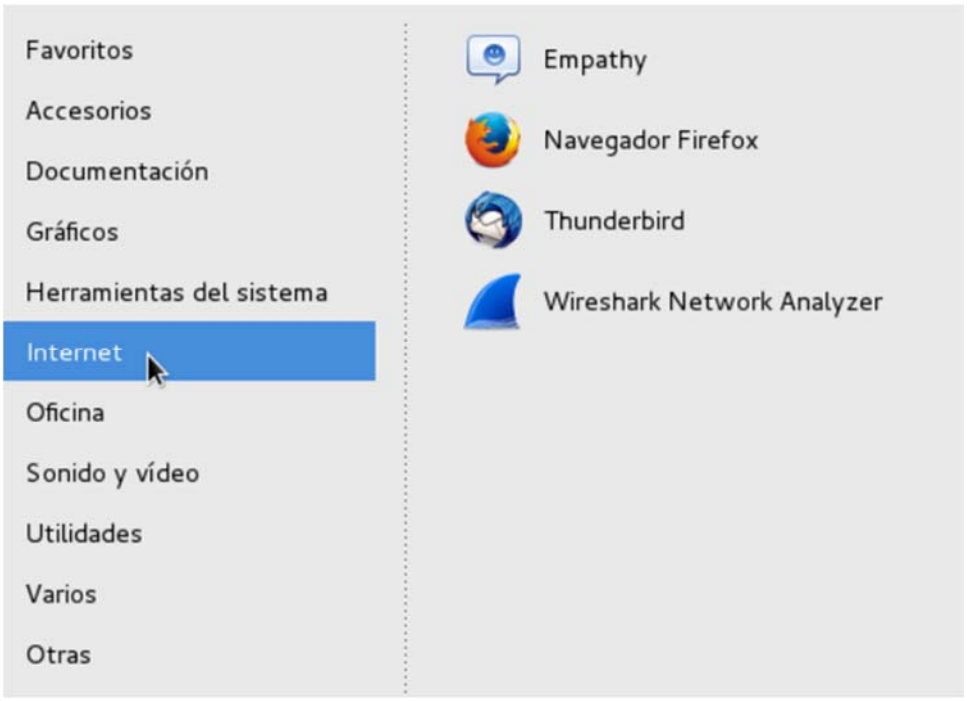
Editamos el fichero /etc/postfix/master.cf y quitamos comentario y/o añadimos las que no estén,



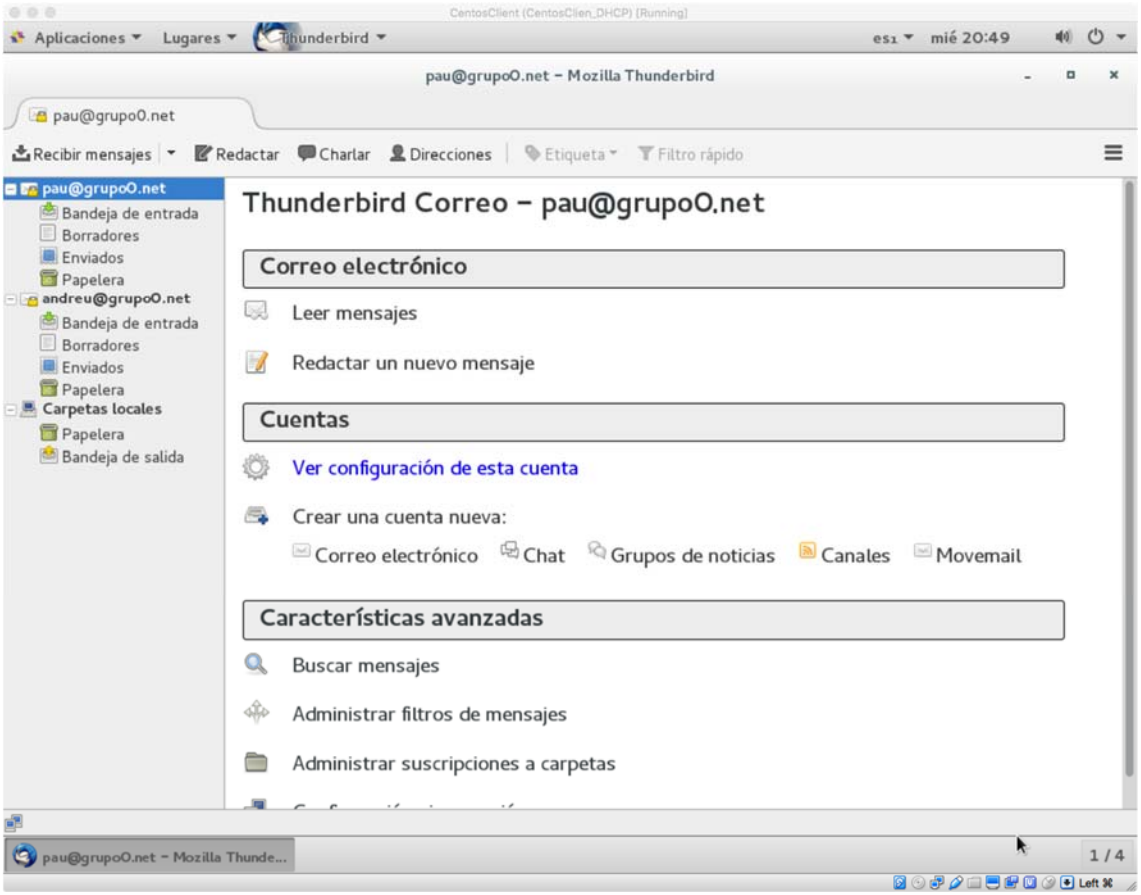
Reiniciamos el servicio postfix.

7. **Comprobación del funcionamiento de los servicios STMP-TLS/IMAP-SSL/POP3-SSL.**

   Desde la máquina cliente de centos, configure la aplicación Thunderbird para el envío y recepción de correo entre los diferentes clientes de correo haya creado.



Tal como se muestra en la siguiente pantall



Compruebe usando wireshark que la comunicación es

8. Consulta de registros

Con el fin de tener el máximo control de los servidores hemos de saber dónde se encuentran los registros o logs. Cualquier tipo de problema que tenga el servidor de correo se almacena en estos ficheros.