# Correo electrónico con SSL

## ¿Cuál es la diferencia entre SSL y TLS?

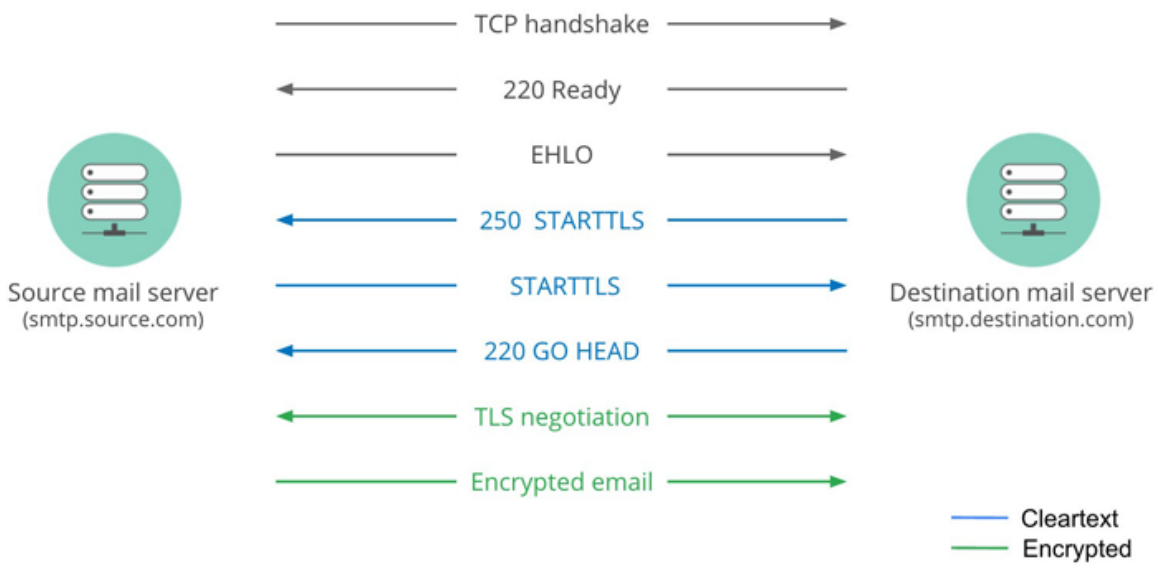
### SSL

SSL significa Secure Sockets Layer (capa de conexión segura). Cuando se utiliza SMTP, este modo SSL se caracteriza por necesitar una conexión segura desde el momento en que se establece la conexión con los servidores de correo. Normalmente, el modo SSL se asocia con el puerto 465, pero no siempre es así. Lo mejor es consultar con tu otro proveedor de correo para obtener más información.

### TLS (conocido más formalmente como STARTTLS)

TLS significa Transport Layer Security (seguridad de la capa de transporte). Cuando se utiliza SMTP, este tipo de protocolo de autenticación comienza con una conexión no segura con el servidor, seguida de un comando STARTTLS y, a continuación, se establece una conexión segura cuando realmente comienza la transmisión de datos.
Por lo general, el modo TLS está asociado con los puertos 25 y 587, pero no siempre es así.

Ten en cuenta que ambos protocolos de seguridad son válidos para garantizar que se encriptarán tus datos, como tu nombre de usuario, tu contraseña y tus mensajes. Esto significa que el texto escrito se traduce a un código para que nadie pueda leer lo que envías. Cuando añades otra dirección de envío, es muy importante saber qué puerto y protocolo de autenticación admite tu otro proveedor de correo.



# Generando los certificados digitales

## Método de certificado autofirmado

Generamos la clave privada para el servidor. Es importante que esta clave no esté con contraseña para que hmailserver pueda funcionar correctamente con este certificado. También es recomendable una longitud de clave de 2048.
**openssl genrsa -out server.key 2048**
Este comando por tanto no valdría, ya que saca la clave con contraseña
~~openssl genrsa -des3 -out~~ **server**~~.key 2048~~

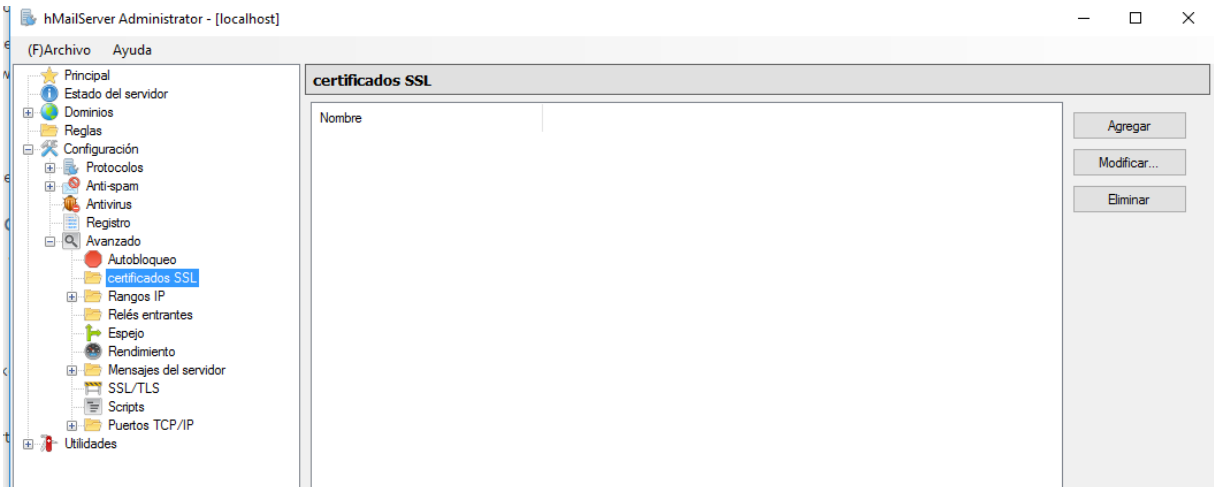
Tambien genero la clave pública aunque no la voy a necesitar
**openssl rsa -in server.key -pubout -out serverPublica.key**
Crear un pedido de firma de certificado
**openssl req -new -key server.key -out server.csr**
El comando nos pedirá una serie de datos, el más importante será el **FQDN** ya que tendrá que coincidir con el nombre de dominio nuestro: **cipformacion.com**
**openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt**

Al final obtenemos 4 ficheros

- **server.key**: la clave privada (ojo sin cifrar, peligroso pero necesario ya que si la ciframos, apache pedirá la clave cada vez que reiniciamos el servidor)
- **serverPublica.key**: la clave pública que no la vamos a necesitar
- **server.csr**: Fichero de pedido de firma de certificado
- **server.crt**: El certificado digital firmado por nosotros mismos

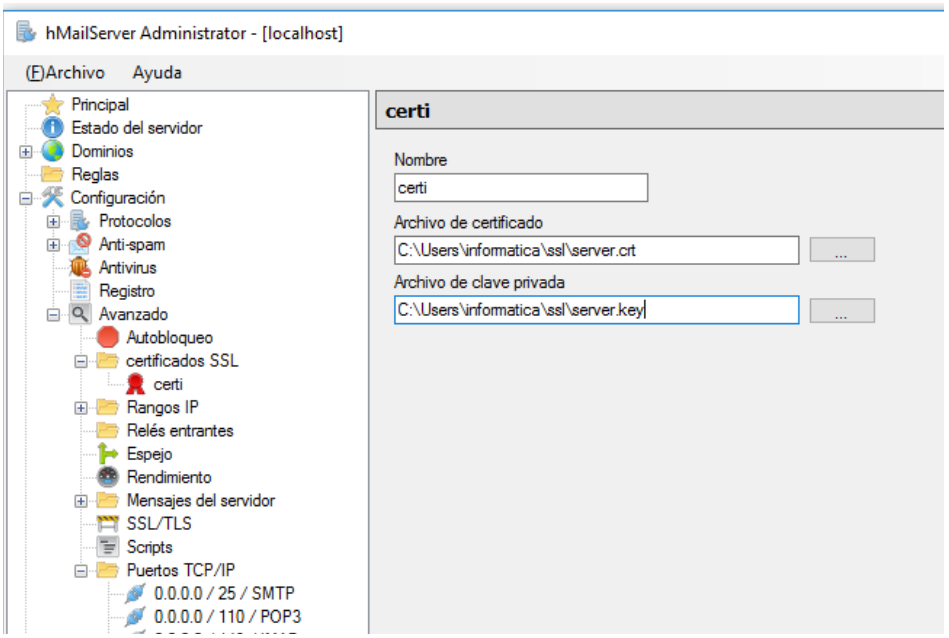
# Configurando hmailserver con ssl

Accede a configuración→ avanzado→ Certificados SSL
Añade un nuevo certificado pulsando en agregar

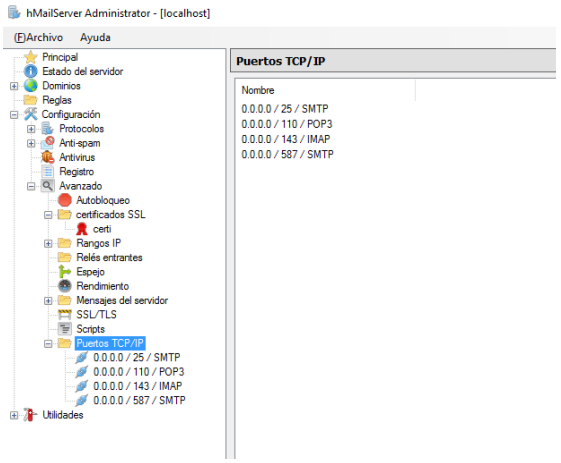



Rellena los datos del certificado:

- Nombre:cualquiera
- Archivo certificado: el fichero con extensión cert
- Archivo de clave privada: el fichero con extensión key (sin proteger con contraseña)



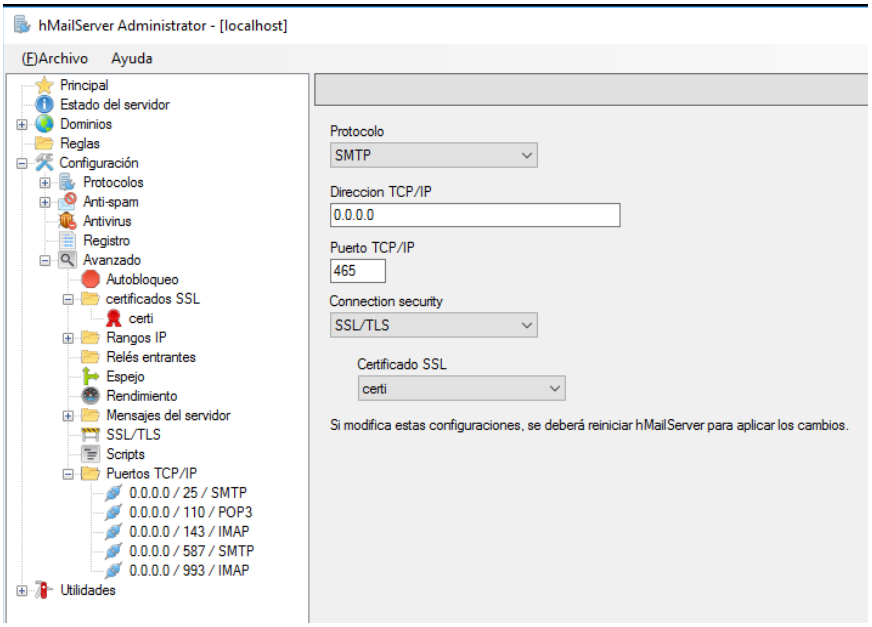

Accede a configuración→ avanzado→ Puertos tcp/ip

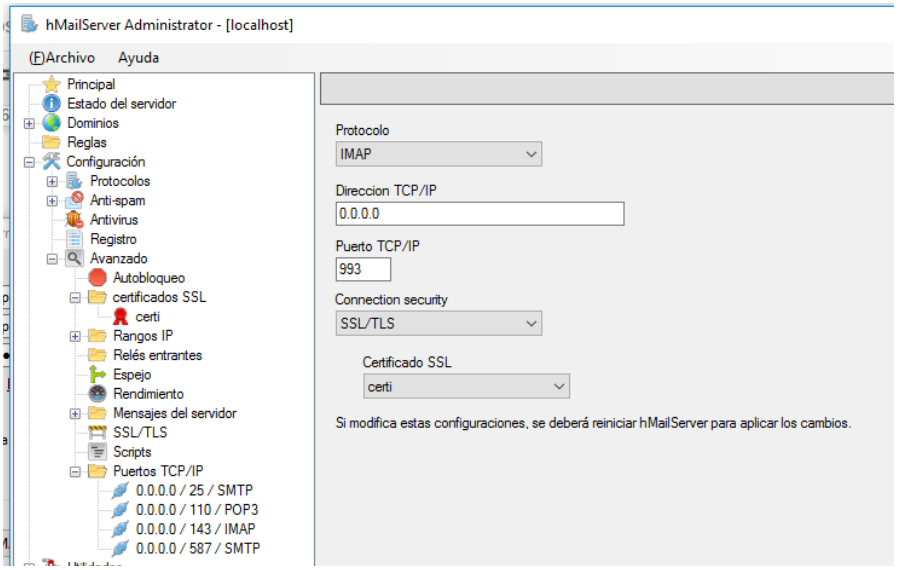


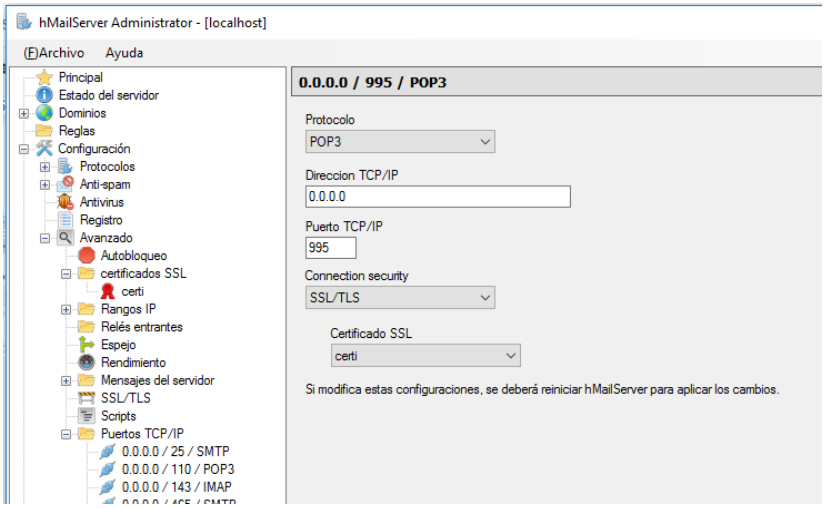
Vamos a configurar puertos cifrados

- SMTP:465
- POP3:995
- IMAP:993

Agrega los siguientes puertos





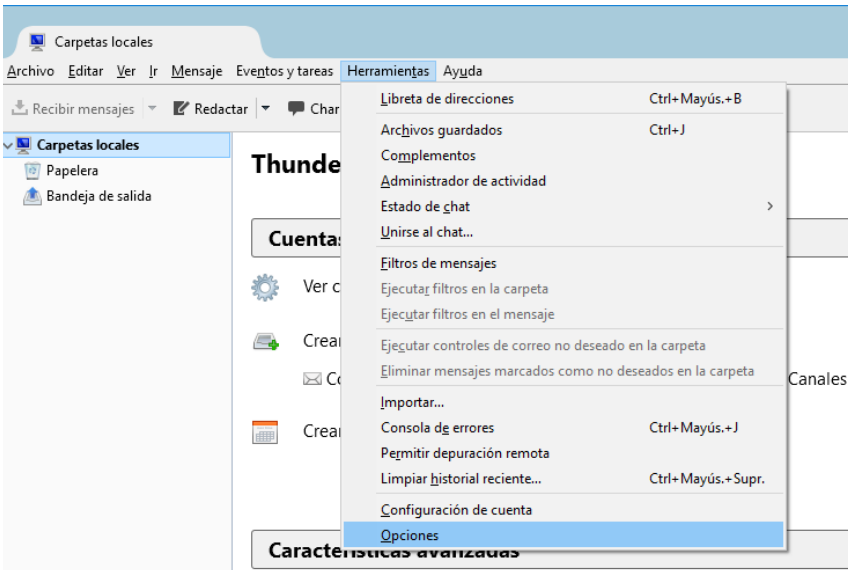

# Probando el correo

*openssl s_client -connect your.maildomain.com:465*

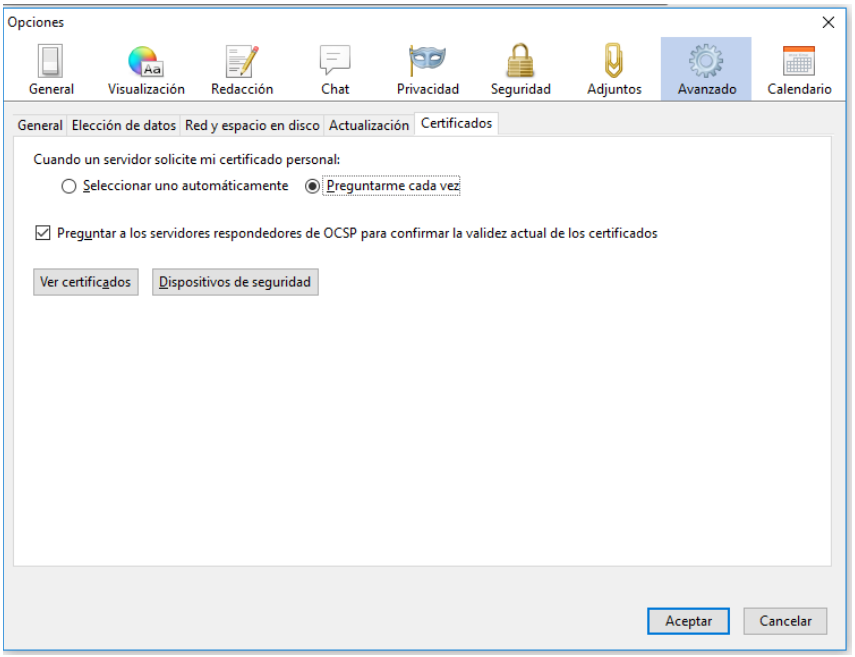
# Configurando el cliente ThunderBird

Cada cliente se configura de una forma diferentes.
Lo primero es añadir nuestro certificado digital a ThunderBird

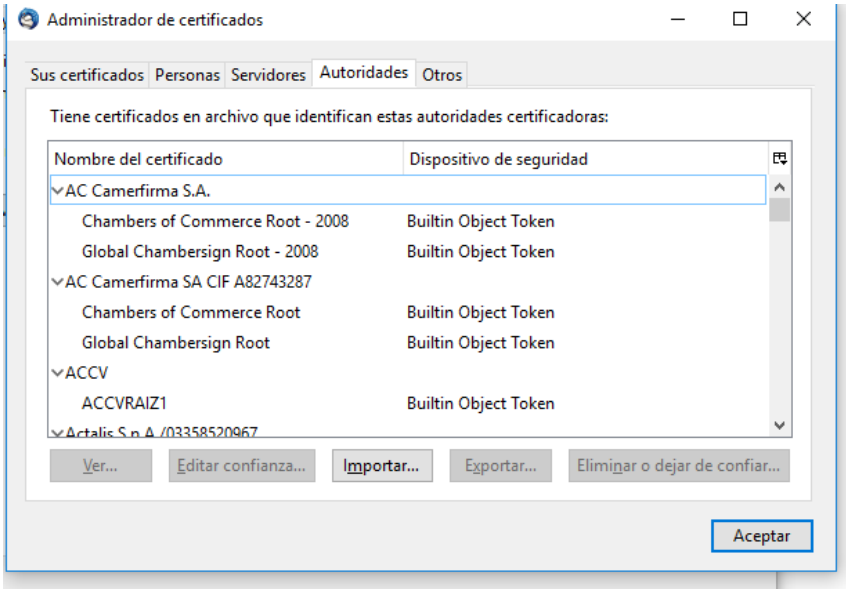
Entramos en herramientas→ opciones



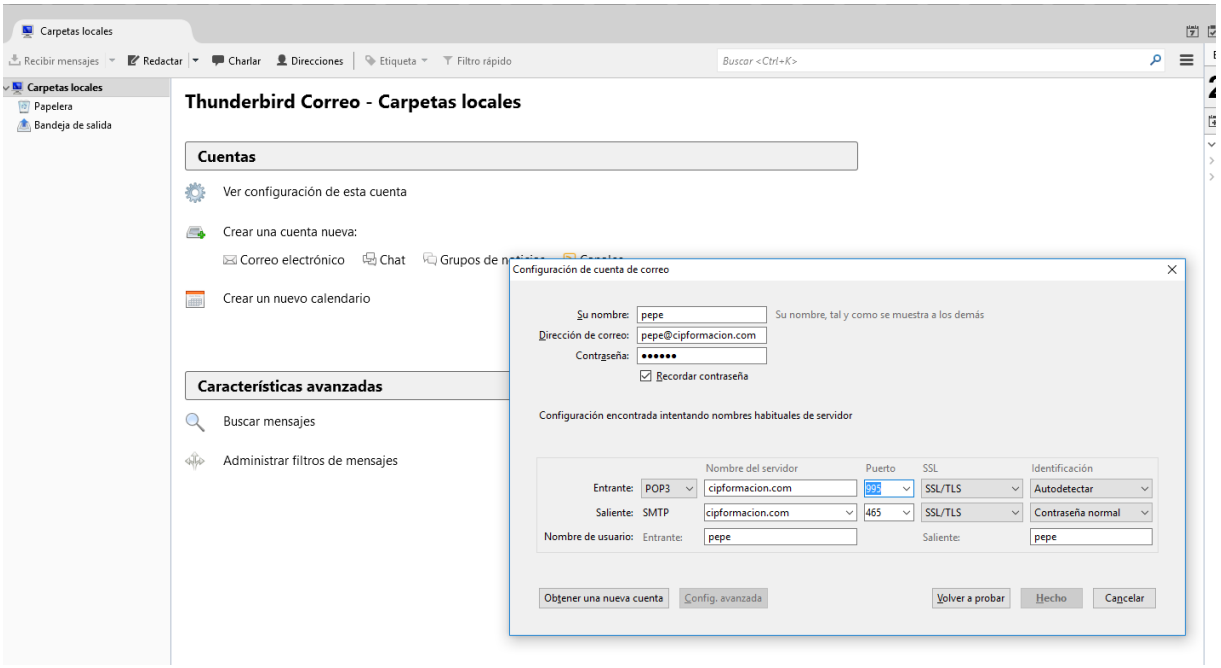

Nos vamos a Avanzado→ Certificados → Ver certificados




Importamos nuestro certificado (fichero con extensión crt).




Añadimos una nueva cuenta (en Thunderbird) correo configurada de la siguiente forma.




# Índice


Generando los certificados digitales
    Método de certificado autofirmado
Configurando hmailserver con ssl
Probando el correo
Configurando el cliente ThunderBird
Índice