

EL SISTEMA DE CORREO ELECTRÓNICO (SMTP Y POP3)

El correo electrónico es una de las aplicaciones TCP/IP más utilizadas en estos días. En su forma más sencilla, el correo electrónico, es una manera de enviar mensajes o cartas electrónicas de un computador a otro.

Estructura y protocolos del correo electrónico

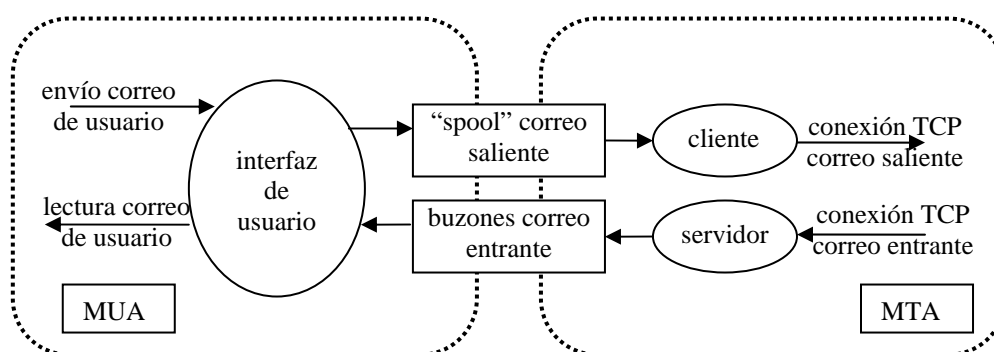
El correo electrónico de Internet se implementó originalmente como una función del protocolo FTP. En 1980 Suzanne Sluizer y Jon Postel realizaron trabajos con un protocolo que posteriormente se denominaría SMTP ("Simple Mail Transfer Protocol"). Hoy en día se sigue utilizando este protocolo, con los avances lógicos que requiere el tipo de transferencia actual.

El protocolo SMTP fue desarrollado pensando en que los sistemas que intercambiarían mensajes, eran grandes computadores, de tiempo compartido y multiusuario conectados permanentemente a la red Internet. Sin embargo, con la aparición de los computadores personales, que tienen una conectividad ocasional, se hizo necesaria una solución para que el correo llegase a estos equipos. Para solventar esta limitación, en 1984 surge POP.

Este protocolo, en su especificación inicial, solo permite funciones básicas como recuperar todos los mensajes, mantenerlos en el servidor y borrarlos. En sucesivas versiones del protocolo (POP2 y POP3) se han ampliado las funciones, permitiendo una mejor gestión del correo.

Por lo tanto, podemos discriminar dos tipos de agentes que están involucrados en la transferencia de correo, MUA y MTA:

- Agente de usuario (*MUA*), interfaz para leer y escribir los mensajes. (POP).
- Agente de transporte (*MTA* o *estafeta*), encargado del transporte de los mensajes. (SMTP).



Estructura general del sistema de correo electrónico



SMTP (SIMPLE MAIL TRANSFER PROTOCOL)

El significado de las siglas de SMTP es Protocolo Simple de Transmisión de Correo ("Simple Mail Transfer Protocol", RFC 821). Este protocolo es el estándar de Internet para el intercambio de correo electrónico. SMTP necesita que el sistema de transmisión ponga a su disposición un canal de comunicación fiable y con entrega ordenada de paquetes, con lo cual, el uso del protocolo TCP (puerto 25) en la capa de transporte, es lo adecuado. Para que dos sistemas intercambien correo mediante el protocolo SMTP, no es necesario que exista una *conexión interactiva*, ya que este protocolo usa métodos de almacenamiento y reenvío de mensajes.

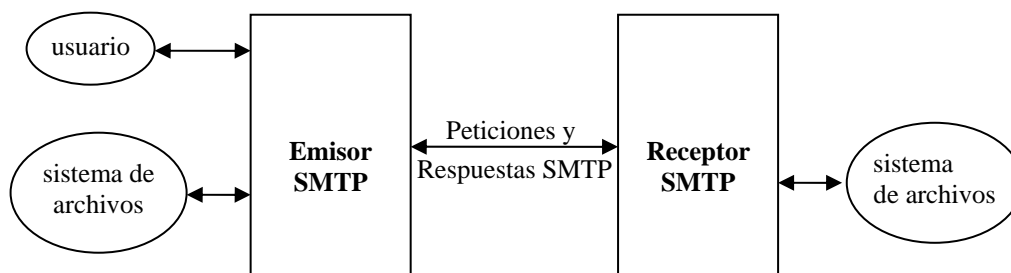
Modo de comunicación SMTP

Cuando un servidor de SMTP, requiere transmitir un mensaje a otro servidor SMTP, el emisor (servidor que inicia la sesión SMTP) establece una conexión con el receptor (servidor que recibe petición de establecer sesión SMTP). Esta conexión es *unidireccional*, es decir, el emisor puede enviar correo al receptor, pero durante esa conexión, el receptor no puede enviar correo al emisor. Si el receptor tiene que enviar correo al emisor, tiene que esperar a que finalice la conexión establecida y establecer otra en sentido contrario, cambiando los papeles de emisor y receptor. Una vez establecida la conexión, el emisor envía comandos y mensajes.

Por lo tanto, el diseño de SMTP se basa en el siguiente modelo de comunicación:

1. Como respuesta a una solicitud de un usuario de enviar un correo electrónico, el emisor SMTP establece una conexión con el receptor SMTP. El receptor SMTP debe ser el destinatario último del correo o un intermediario. Para ello el emisor genera los comandos SMTP en formato ASCII y los envía al receptor y el receptor genera las respuestas a los comandos enviados por el emisor.
2. Una vez establecido el canal de transmisión, el emisor envía el comando MAIL para indicando que el es el emisor del correo. Si el receptor puede aceptar correo responde con el comando OK.
3. El emisor envía el comando RCPT identificando el destinatario del correo. Si el receptor puede aceptar correo para ese destino responde con una respuesta OK; si no, responde rechazando el correo para ese destino.
4. Una vez negociado el destino, el emisor comienza a enviar datos (cabeceras y cuerpo), terminando con una secuencia especial. Si el receptor ha procesado correctamente los datos, responde con el comando OK.

La siguiente figura muestra los componentes que actúan en este modelo de uso del protocolo.



Ejemplo de SMTP

A continuación se muestra como sería una conversación con un receptor SMTP:

```
#> telnet goliat.ugr.es 25
```

```
Trying 150.214.20.3...
```

```
Connected to goliat.ugr.es.
```

```
Escape character is '^['.
```

```
220 goliat.ugr.es ESMTP Sendmail 8.9.3/8.9.1 (IRIS 3.0); Wed, 5 Mar 2003 12:17:19 +0100
```

```
ehlo fobos.ugr.es
```

```
250-goliat.ugr.es Hello fobos.ugr.es [150.214.190.83]
```

250-8BITMIME
250-SIZE 8000000
250-DSN
250-ONEX
250-ETRN
250-XUSR
250-HELP

help

214-This is Sendmail version 8.9.3
214-Topics:
214- HELO EHLO MAIL RCPT DATA
214- RSET NOOP QUIT HELP VRFY
214- EXPN VERB ETRN DSN
214-For more info use "HELP <topic>".
214 End of HELP info

mail from: usuario@fobos.ugr.es

250 usuario@fobos.ugr.es... Sender ok

rcpt to: pgteodor@ugr.es

250 pgteodor@ugr.es... Recipient ok

data

354 Enter mail, end with "." on a line by itself

Subject: Resumen del mensaje

Cc: usuario2@dominio

Cuerpo del mensaje ...

.

250 MAA21341 Message accepted for delivery

quit

221 goliat.ugr.es closing connection
Connection closed by foreign host.

#> _

POP (POST OFFICE PROTOCOL)

El protocolo de oficina de correo, POP, es un protocolo cuya misión es la de entrega final del correo al destinatario. Puesto que con SMTP lo único que se consigue es la transferencia del correo entre buzones, es necesario un protocolo como POP (o también IMAP) con el que podamos descargar el mensaje desde el buzón.

Modelo de comunicaciones POP

La descripción del protocolo POP la podemos encontrar en el RFC 1725, y como se acaba de comentar el modelo de comunicaciones POP se basa en mantener un buzón (estafeta) central en la que se almacenan los mensajes hasta que el usuario solicita la descarga de los mismos.

El cliente POP se conecta con el servidor a través del puerto TCP, 110. Para conectarse al servidor, es necesario una cuenta de identificación en dicha máquina (lo que le permite tener un espacio reservado para sus correos). A continuación es necesario verificar que es dueño de la cuenta a través de una clave. Una vez conectado al sistema, el cliente POP puede dialogar con el servidor para saber, en otros, si existen mensajes en la casilla, cuántos mensajes son o para solicitar la descarga de alguno de ellos.

Para poder ofrecer estas funciones, el modelo de comunicación POP se basa en estados:

- Autorización.
- Transacción.

- Actualización.

Después de establecer la conexión, el servidor POP se encuentra en un estado de autorización, esperando que el cliente le envíe el nombre y clave de la cuenta de usuario. Cuando se verifica que el nombre y la clave son correctos, el servidor pasa a un estado de transacción. Antes de pasar a este estado, el servidor POP bloquea el buzón para impedir que los usuarios modifiquen o borren el correo antes de pasar al estado siguiente. En este estado de transacción el servidor atiende las peticiones del cliente. Después de enviar al servidor el comando QUIT, el servidor pasa al estado de actualización (estado siguiente). En este estado el servidor elimina los mensajes que están con la marca de borrado y finaliza la conexión.

Ejemplo de POP3

#> telnet goliat.ugr.es 110

+OK QPOP (version 2.4) at Goliat starting.

AUTH KERBEROS_V4

-ERR This command is not supported yet

USER usuario

+OK Password required for usuario.

PASS clave

+OK usuario has 2 messages (320 octets).

STAT

+OK 2 320

LIST

+OK 2 messages (320 octets)

1 120

2 200

.

RETR 3

-ERR Message 3 does not exist.

RETR 2

+OK 120 octets <mensaje1> ...

.

DELE 2

+OK Message 2 has been deleted.

QUIT

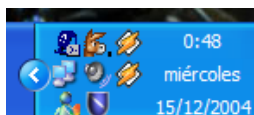
+OK Pop server at goliat signing off.

ARGOSOFT Mail Server

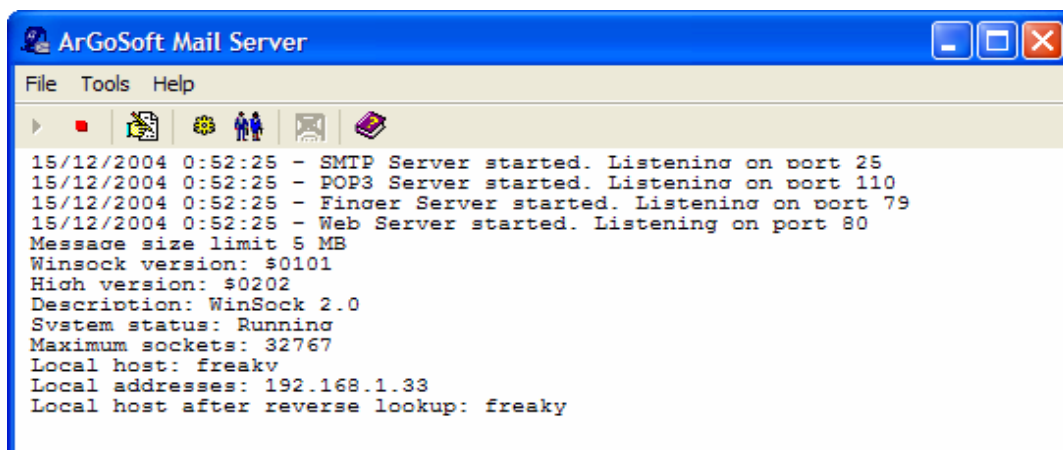
Argosoft Mail Server es un programa que incluye un servidor SMTP y un servidor POP3 con los que podremos aprender el funcionamiento del sistema de correo electrónico. Vamos a realizar la demostración de su funcionamiento con la versión freeware. Para ello entramos en la página de descargas de Argosoft <http://www.argosoft.org/mailserver/download.aspx>. En esta web nos aparecerán 3 tipos nosotros vamos a descargar esta ArGoSoft Mail Server (Freeware) que es la versión gratuita, y al pinchar para descargar nos aparece un formulario que hay que rellenar.

Configuración del programa

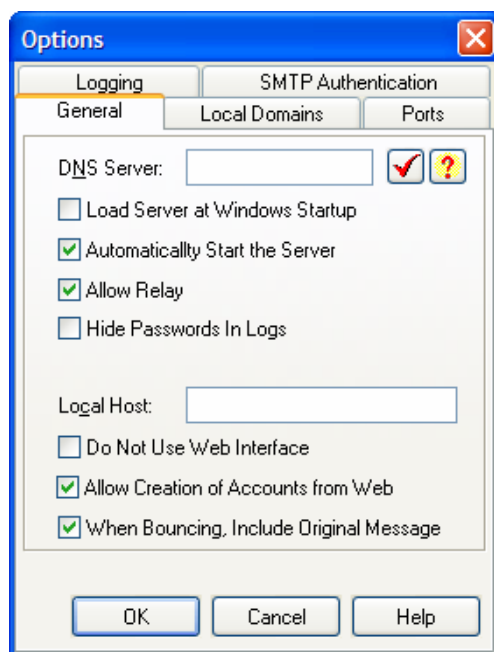
Una vez instalado se queda abierto pero minimizado en la parte inferior derecha de la pantalla, ver la siguiente imagen. Pulsamos doble clic para iniciarlo.



En la figura que se muestra a continuación, podemos ver el aspecto del programa.

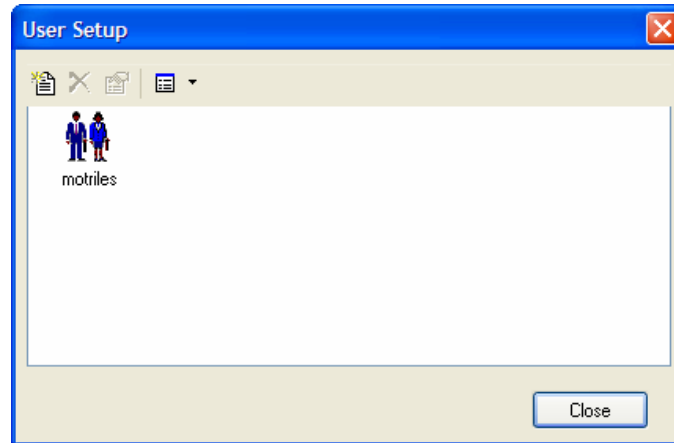


Lo primero que tenemos que hacer es configurar el programa, para ello iremos a Tools-Options, con lo que se mostrará la siguiente ventana.



Puesto que la configuración que vamos a hacer es para probar que los protocolos funcionan correctamente, no vamos a hacer grandes cambios en la configuración ni vamos a registrar un dominio. Lo único que cambiaremos en esta ventana es activar Allow Relay.

Lo siguiente que vamos a hacer es crear al menos un usuario con el que poder hacer las pruebas: Tools-Users.



Pulsando sobre el icono de la barra de herramientas de la izquierda se muestra una ventana en la que introduciremos un nombre de usuario junto con su password, que posteriormente utilizaremos para enviar y recibir los correos electrónicos.

Ejemplo de uso

Lo primero que vamos a hacer es enviar con SMTP un correo, para ello abrimos un Símbolo del sistema y escribimos los comandos que en negrita se muestran.

telnet localhost 25

220 ArGoSoft Mail Server Freeware, Version 1.8 (1.8.6.9)

ehlo

250-Welcome [127.0.0.1], pleased to meet you
250-SIZE 5242880
250 HELP

mail from: motriles

250 Sender "motriles" OK...

rcpt to: motriles

250 Recipient "motriles" OK...

data

354 Enter mail, end with "." on a line by itself

subject: probando que esto funcioan

Cuerpo del mensaje

.

250 Message accepted for delivery. <wzz7bl5vz43lwxo.151220040108@freaky>

quit

A continuación lo leeremos con POP3, para ello escribimos los siguientes comandos que están en negrita en un Símbolo del sistema:

telnet localhost 110

+OK ArGoSoft Mail Server Freeware, Version 1.8 (1.8.6.9)

user motriles

+OK Password required for motriles

pass jose

+OK Mailbox locked and ready

stat

+OK 4 995

list

+OK

1 248

2 237

3 230

4 280

.

retr 4

+OK 280 octets

Received: from [127.0.0.1] by freaky

(ArGoSoft Mail Server Freeware, Version 1.8 (1.8.6.9)); Wed, 15 Dec 2004 01:08 :32 +0100

Message-ID: <wzz7bl5vz43lwzo.151220040108@freaky>

Date: Wed, 15 Dec 2004 01:08:32 +0100

subject: probando que esto funcioan

Cuerpo del mensaje

.

dele 1

+OK Message number 1 marked

quit

Entorno web de Argosoft

Aunque el programa es muy simple, incluye también un entorno web al que podremos acceder escribiendo en una ventana del navegador <http://localhost> siempre y cuando no tengamos ningún conflicto con los puertos de otros servidores web. Mediante este entorno podemos comprobar el correo de una forma muy sencilla.

Date 15/12/2004 Time 1:16:09



Welcome to ArGoSoft Mail Server

You can check your mail from here

Login

Add New User

Your browser must support **JavaScript** also, accept **cookies**, otherwise you will be unable to use this service.


 **Click Here!** 

ArGoSoft Mail Server Freeware, Version 1.8 (1.8.6.9)

[© ArGo Software Design, 1999-2004](#)

En la siguiente imagen podemos ver el "Inbox" para el usuario motriles.

Date 15/12/2004 Time 1:24:31





User **motriles**

You have **4** messages waiting.

Compose **Delete** **Check Mail** **Settings** **Log Out**


	From	Subject	Date	Size
<input type="checkbox"/>	[Unknown]	[No Subject]	15/12/2004 0:26:05	248
<input type="checkbox"/>	[Unknown]	[No Subject]	15/12/2004 0:33:19	237
<input type="checkbox"/>	[Unknown]	[No Subject]	15/12/2004 0:35:27	230
<input type="checkbox"/>	[Unknown]	[No Subject]	15/12/2004 1:08:33	280

**...BY EMAIL**

ArGoSoft Mail Server Freeware, Version 1.8 (1.8.6.9)

Si queremos ver el correo que nos hemos enviado manualmente no tenemos más que navegar al estilo de cualquier webmail.


Date 15/12/2004 Time 1:25:04



Compose **Reply** **Next** **Previous** **Msg List** **Delete**

From: [Unknown]
Subject: [No Subject]
Date: Wed, 15 Dec 2004 01:08:32 +0100

subject: probando que esto funcioan00
Cuerpo del mensaje

**GET FREE STUFF...**

ArGoSoft Mail Server Freeware, Version 1.8 (1.8.6.9)

© ArGo Software Design, 1999-2004