

SMTP vía TLS con Evolution, Kmail y Mutt.

# Correo Más Seguro

La transmisión segura de correo electrónico mediante la Seguridad de la Capa de Transporte (TSL) no puede reemplazar la encriptación individual de mensajes de correo. Desafortunadamente, a pesar de que los programas modernos de correo son capaces de usar los métodos mencionados, tienden a complicar la vida innecesariamente de cualquier usuario que desee aprovechar los beneficios de la confidencialidad y seguridad.

POR PATRICIA JUNG



Los mensajes de correo electrónico son como una postal - cualquier "cartero" los puede leer. Si bien la mayoría de los administradores de sistemas escrupulosos evitarán leer el correo privado de otras personas, al menos en teoría, los privilegiados poseedores de la contraseña *root*, autorizados o no (pienso en los hackers), de una máquina usada para remitir o almacenar correo tendrán acceso a los mensajes. Los servicios secretos de todo el mundo aclaman exultantes a los padres (y madres) fundadores de Internet, que fueron tan confiados que concibieron la transmisión de todo tipo de datos a través de la red de forma transparente.

Encriptar, por ejemplo con GnuPG y PGP, es el equivalente electrónico de un sobre en el mundo real. Sin embargo, los métodos anteriores plantean un problema: si un país niega a sus ciudadanos el acceso a métodos seguros de encriptación - típicamente argumentando que es en ayuda de la lucha contra el crimen - el protocolo de transporte de correo, SMTP (Protocolo de Transporte de Correo Simple), se convertirá en un delator. El contenido de mensajes encriptados con PGP o GnuPG descubiertos en la red no pueden ser leídos, pero podemos observar cuales han sido encriptados y cuales no. De esta forma,

el uso de sólidos métodos de encriptación se convierte en una prueba de definitiva en contra de todos esos "criminales" que demandan confidencialidad en sus mensajes de correo.

Sine embargo, cada vez más proveedores de correo están intercambiando el corre por medio túneles TSL (Transport Layer Security - Seguridad de la Capa de Transporte) encriptados en lugar de al descubierto, previniendo por tanto ataques de búsqueda. Hay otras razones para seguir la tendencia de alejarse de los mensajes transmitidos en texto libre, usando SSH en lugar del venerable protocolo telnet o el protocolo *https* para páginas web que piden datos personales.

La introducción de una política que impone el uso de correo encriptado en una empresa con distintas sedes puede provocar ciertos inconvenientes: ¿Qué ocurre si una tercera persona necesita acceder al correo posteriormente? ¿Qué ocurre con las claves utilizadas por una persona de la empresa que deja de trabajar en ella, o con su correspondencia encriptada? Como la encriptación normalmente solo se utiliza para prevenir que datos sensibles crucen la red al descubierto, un sistema de encriptación que funcione en el servidor y que sea transparente para los usuarios, como SMTP/TLS es una buena solución.

## Agentes Secretos

Esta solución ofrece adicionalmente la ventaja de ser libre de mantenimiento tras la configuración inicial del servidor. Si tanto el cliente y el servidor de correo "hablan" TLS, ellos negociarán un intercambio seguro sin necesidad de intervención exterior. Para permitir esto, ambos ordenadores intercambian certificados durante el saludo inicial facilitando la identificación mutua. Es como si cada ordenador le muestra su pasaporte al otro. Por motivos prácticos, solo será necesario que se autentifique el receptor principal en el entorno del correo, no habiendo diferencia entre si el receptor es otro servidor de correo o el programa de correo del usuario. Como el cliente ya puede confiar en el servidor, las dos máquinas acuerdan un algoritmo y una llave secreta de encriptación se utilizarán para codificar el tráfico resultante.

## GLOSARIO

**https:** Si bien puede que sorprenda saberlo, pero *https* no es un protocolo independiente. La comunicación entre buscadores y servidores en sitios Web que usan URLs *https* sigue estando basada en el "Hypertext Transfer Protocol" HTTP (Protocolo de Transferencia de Hiper Texto), pero encapsulado en un túnel TLS.

La gente tiende a confiar en las contraseñas porque se fían de la autoridad que las otorga. De forma similar, el cliente de correo solo aceptará los certificados de un servidor si confía en que la autoridad que lo certifica ha hecho el proceso con consistencia. La autoridad autentifica el certificado firmándolo con su propio certificado, que a su vez ha sido firmado por un proveedor seguro de mayor rango.

A pesar de todo, para confiar en una autoridad, el cliente necesitará almacenar los certificados de las autoridades de confianza superior, los certificados raíz.

## Cómo se Hace

Mientras que los servidores de correo (o Agentes de Transferencia de Correo) como Postfix o Sendmail usarán automáticamente TLS si ambas partes son capaces de hacerlo, los desarrolladores de clientes de correo actuales [1] (los llamados Agentes de Usuarios de Correo) tienden a prestar poca atención a la facilidad de uso. En lugar de preconfigurar sus programas para usar SMTP/TLS cuando sea posible, esperan que los usuarios sean los que conozcan la materia y que elijan y sepan implementar estos valores seguros.

En el caso de *KMail* sobre KDE3.x, si seleccionamos *Settings/Configure...* y después añadimos una nueva cuenta de salida (*Outgoing account*) con SMTP en

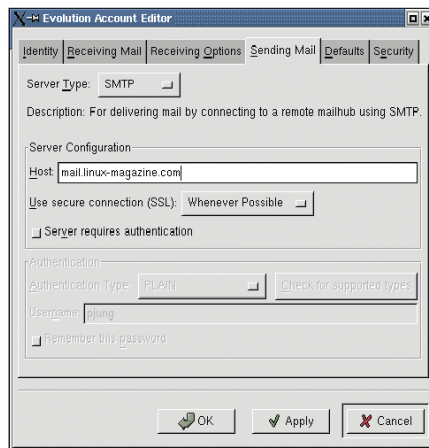


Figura 2: ¿Por qué no es esta la opción por defecto?

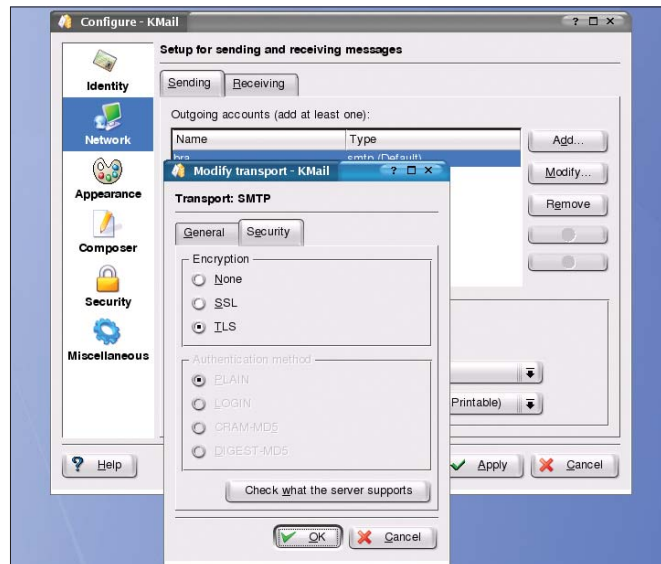


Figura 1: KMail deja en manos del usuario descubrir si el servidor de correo conoce TLS.

la pestaña *Sending* de la opción *Network* o modificamos una cuenta existente, necesitamos seleccionar adicionalmente la pestaña *Security* en el cuadro de diálogo que aparece (figura 1). Esto al menos nos proporcionará la opción de marcar en *Check what the server supports* para verificar lo amigable que es la encriptación del anfitrión inteligente seleccionado en la pestaña *General*. Si la repuesta es positiva (la alternativa, SSL - Capa de conexión segura- es un antecedente de TLS), podemos pulsar *OK* para asegurar que Kmail usará comunicaciones encriptadas.

La versión 1.0.x de *Evolution* proporciona una opción segura SSL para el anfitrión inteligente. Desafortunadamente, marcando *Use secure connection (SSL)* (Usar conexión segura) se convierte en un escollo en muchos casos. La mayoría de los servidores no soportan el método obsoleto de SMTP vía SSL; por otro lado, *Evolution* 1.0.x no habla TLS. Si intentamos transmitir correo con esta configuración, el programa de correo no cooperará y emitirá un mensaje muy explícito como *Connection to name.of.mailservers (Port 465) could not be established. The connection was reset by the communication partner.* (La conexión con nombre.de.los.servidores (puerto 165) no se pudo establecer. La conexión fue restablecida por el asistente de comunicación). El cliente ni siquiera intentará la comunicación no encriptada hasta que quitemos la marca en *Mail*

*Preferences* (Preferencias del correo).

La versión 1.2 resuelve los dos problemas, si bien el asistente o el cuadro de diálogo de *Use secure connection (SSL)* (figura 2) se refiere de nuevo sólo a SSL, sin mencionar TLS, éste último protocolo también esta soportado. Solo Dios sabe por que el valor *Whenever possible* (Cuando sea posible) no es el valor por defecto.

Cuando intentamos enviar correo encriptado por primera vez, *Evolution* nos mostrará información del certificado del servidor y nos solicitará que lo aceptemos o que no utilicemos el túnel

SSL/TLS (ver figura 3). Como el programa no nos ayuda a tomar esta decisión (por ejemplo, que significa si una firma es *BAD*), puede que sea mejor dejar que el programa tome la decisión él mismo en lugar de preocupar al usuario preguntándole.

El programa de la línea de comandos *mutt* es ejemplar: como siempre usa servidores de correo local para transmitir el correo mediante el interfaz */usr/sbin/sendmail*, no necesita soportar SMTP/TLS. El mensaje cogerá automáticamente un camino seguro a través de la red si el Agente Local de Transferencia de Correo y el servidor de destino lo pueden proporcionar sin ninguna interacción por parte del usuario. Pero no podemos evitar proporcionar a nuestro servidor de correo local de este potencial en nuestro propio ordenador. ■



Figura 3: ¿Confías en este certificado?

## RECURSOS

- [1] Patricia Jung y Andrea Müller: "Mail and more", *Linux Magazine International*, número 29, abril del 2003, página 44, <http://www.linux-magazine.com/issue/29/MailUserAgents.pdf>