

5.3. Seguridad en la red



Índice

Objetivos.....	3
Seguridad en las comunicaciones.....	4
Mecanismos de seguridad	4
Bastion host.....	4
DMZ o zona perimetral.....	4
Protocolos de seguridad en redes IP	6
Seguridad en IPv6.....	6
Protocolos de seguridad	7
Seguridad en redes inalámbricas	9
Mecanismos y protocolos de seguridad en redes inalámbricas	10
Recomendaciones de seguridad en redes inalámbricas.....	12
Cifrado simétrico y asimétrico.....	12
Ejemplo de cifrado simétrico.....	14
Ejemplo de cifrado asimétrico.....	17
Generación de las claves en Ubuntu.....	18
Acceso a redes de área amplia.....	23
Tecnologías de acceso.....	24
Principios y medidas de seguridad.....	24
Mecanismos de seguridad.....	25
¿Qué medidas podemos tomar?	26
Riesgos en las comunicaciones	27
Amenazas pasivas frente a amenazas activas.....	27
Ejercicio práctico: conexión SSH Windows-Linux	28
Despedida	31
Resumen.....	31

La seguridad en las conexiones de red es un elemento de extrema necesidad hoy en día; ya apenas concebimos el trabajo sin “estar conectados”. Conocer las diferentes medidas a tomar para garantizar una seguridad básica es algo que todo administrador de un sistema debe saber. En este módulo haremos una primera introducción al tema de la seguridad en las redes, aunque en particular en esta materia debes estar dispuesto a un continuo aprendizaje.

Objetivos

Con esta lección perseguimos los siguientes objetivos:

1. Estudiar la seguridad en las conexiones de red como elemento de extrema necesidad en una realidad de ordenadores conectados.
2. Conocer las principales medidas a tomar por el administrador de la red para garantizar una seguridad básica.
3. Conocer los peligros y amenazas que corremos cuando estamos conectados a la red, así como los modos de prevenirlos y subsanarlos.

Seguridad en las comunicaciones

Mecanismos de seguridad

Además de los *firewalls* podemos incluir funciones de seguridad en otros dispositivos de red.

Por ejemplo, a la vez que realizan el encaminamiento de la información, **los routers pueden hacer filtrado de paquetes en base a sus direcciones IP** (mirando las cabeceras). Hacer este filtrado de paquetes requerirá una cierta configuración de sus tablas de enrutamiento.

Existen diversos mecanismos de seguridad que podemos implementar en nuestra red. Veamos algunos.

Bastion host

Se puede configurar un **bastion host**, que se trata de un equipo seguro que se pone como punto de contacto visible desde el exterior, y a través del cual se envía y recibe el tráfico permitido, concentrando en él la seguridad (aunque evidentemente hay que asegurar que no se accede a él con privilegios como para alterarlo).

Un **servidor proxy** puede ser un ejemplo de "bastion host". Normalmente se instala con un software que enmascara y protege la identidad de las máquinas internas de la red, al mismo tiempo que realiza funciones de filtrado y *routing* a varios niveles, desde el nivel de red al nivel de aplicación.

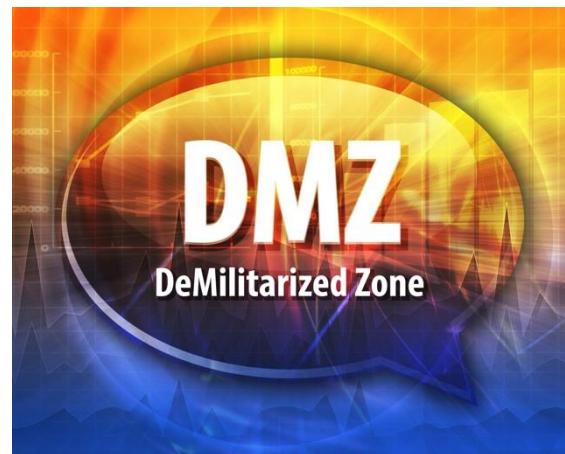
DMZ o zona perimetral

Otro mecanismo de seguridad puede ser el establecimiento de una "**DMZ**" o "**zona perimetral**". Una DMZ **es un trozo de red que se interpone entre la red interna y la conexión exterior**, de forma que sus equipos pueden ofrecer servicios hacia el exterior, pero no tienen acceso a la red interna y actúan de muro/frontera para los agentes exteriores que intenten conectarse hacia el interior de nuestra red.

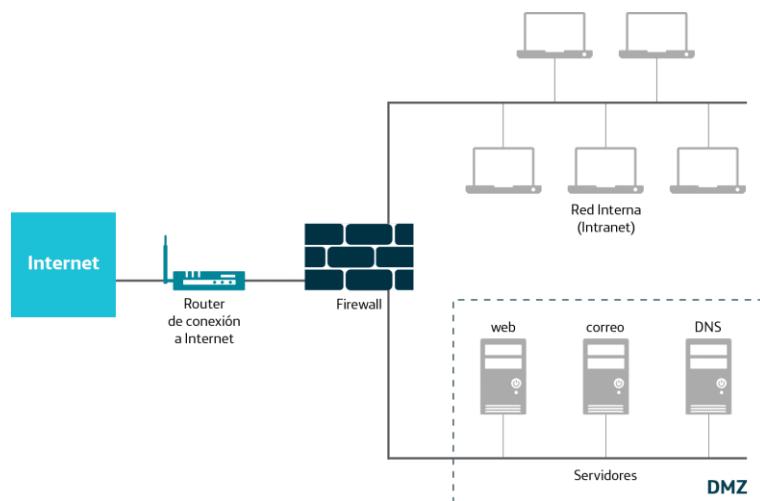
La DMZ puede construirse utilizando las opciones de configuración del "*firewall*", conectando cada red a un puerto distinto del equipo. Esta configuración se llama cortafuegos en trípode o "**three-legged firewall**".

También pueden usarse dos cortafuegos, lo cual es más seguro y ayuda a prevenir el acceso desde el exterior. Uno de los cortafuegos es el llamado "**front-end**" y debe permitir el tráfico únicamente del exterior hacia la DMZ. El otro cortafuegos ("**back-end**") solo permite el tráfico desde la DMZ hacia la red interna.

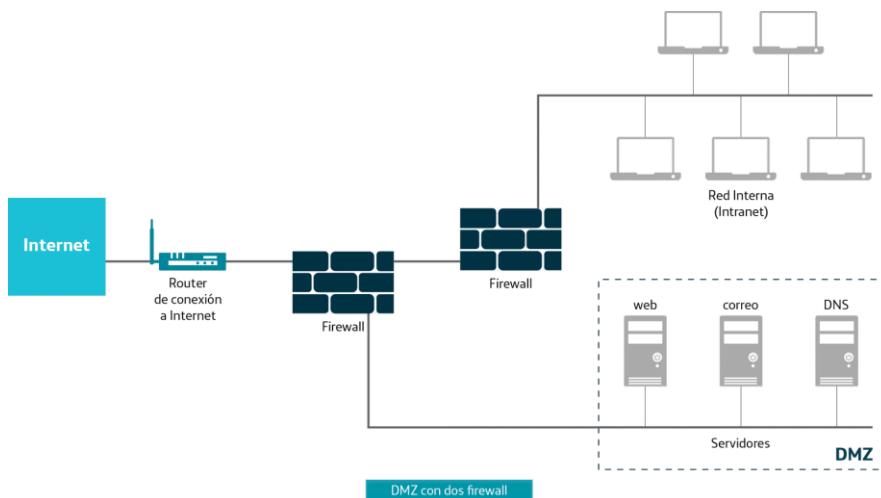
La DMZ suele usarse habitualmente para ubicar servidores a los que es necesario acceder desde fuera de la red interna (como servidores de correo electrónico, web y DNS), protegiendo el resto de nuestra infraestructura de red.



Diagramas de ejemplo: cómo montar una DMZ



DMZ con un firewall (cortafuegos en trípode o “*three legged firewall*”).



DMZ con dos firewalls (“*front end firewall*” + *backend firewall*”).

Protocolos de seguridad en redes IP

La seguridad en las redes IP puede venir dada por la fortaleza de los protocolos empleados o por mecanismos específicos de seguridad para el intercambio de información sensible.

Hemos de tener en cuenta que el protocolo **IPv4**, en su diseño original, no contemplaba mecanismos de seguridad, por lo que se hizo necesario incluirlos a nivel de la "capa de aplicación". Los principales protocolos utilizados en redes IP son:

- **SSH** (Secure SHell).
- **SSL** (Secure Socket Layer).
- **S-HTTP** (Secure Hypertext Transfer Protocol).
- **S/MIME** (Secure/Multipurpose Internet Mail Extensions).
- **SET** (Secure Electronic Transaction).
- **Firma digital** (Autentificación, integridad y no repudio).
- **IPSec**.

Seguridad en IPv6

Precisamente una de las ventajas más importantes de **IPv6** es la introducción de una información de autentificación situada entre la cabecera del mensaje y el campo de datos. Esta cabecera no modifica el comportamiento de los protocolos de alto nivel y proporciona una seguridad del origen del datagrama, que de esta forma puede ser aceptado o rechazado por los niveles superiores.



Protocolos de seguridad

Veamos algunos de los protocolos de seguridad más usados en las redes IP, aunque ten en cuenta que no buscamos que los aprendas en detalle (cualquiera de ellos es bastante complejo) sino que te suenen y conozcas sus características generales:

SSH

SSH (Secure SHell) es un protocolo que **sirve para acceder, a través de la red, a máquinas remotas mediante un intérprete de comandos**. Además nos permite copiar datos de forma segura (archivos sueltos o simular sesiones FTP cifradas), gestionar claves RSA y enviar datos de otra aplicación por un **canal seguro "tunelizado"**.

SSH trabaja de forma similar a Telnet, pero usando técnicas de cifrado de la información que dificultan los ataques (aunque no los impiden del todo) y protegen, tanto al usuario y a la contraseña de la conexión, como la información intercambiada durante la sesión.

SSL

SSL ("Secure Socket Layer") y su sucesor "Transport Layer Security" (TLS) son protocolos criptográficos enfocados al establecimiento de comunicaciones seguras a través de la red. SSL proporciona autenticación y privacidad de la información entre origen y destino, **garantizando la identidad del servidor** mientras que **el cliente se mantiene sin autenticar**.

El protocolo SSL sigue un **proceso en tres fases**:

- Origen y destino negocian el algoritmo que se usará en la comunicación.
- Se intercambian claves públicas y de autenticación mediante certificados digitales.
- Se intercambia el tráfico basado en cifrado simétrico.

VENTAJAS DE SSL

- Está normalmente incluido en los navegadores actuales, por lo que no es necesaria ninguna instalación adicional.
- El hecho de que el cliente use SSL no afecta significativamente al rendimiento del servidor.
- SSL puede usarse como alternativa a una VPN (red privada virtual) si esta no existe.

DESVENTAJAS DE SSL

- Al usar SSL el *firewall* no puede ver el contenido real de los datos encriptados, y sería posible incluir un virus en ellos.
- La seguridad de SSL depende del cliente; si los códigos que genera no son suficientemente "aleatorios" sería fácil desencriptar la sesión y un atacante podría "adivinar" la clave.



S-HTTP

Es un protocolo de transferencia segura de hipertexto, usado para transacciones seguras en la web y que **no debemos confundir con "HTTPS"** ("Hypertext Transfer Protocol Secure"), que es HTTP sobre SSL o TLS.

S-HTTP provee una variedad amplia de mecanismos para disponer de confidencialidad, autenticación e integridad. Las encapsulaciones pueden incluir cifrado, firma digital o una autenticación basada en MAC. Sobre un mismo mensaje pueden aplicarse varias transformaciones de seguridad.

S/MIME

S/MIME es un estándar para criptografía de clave pública y firmado de correo electrónico encapsulado en MIME. Provee los siguientes servicios de seguridad criptográfica para aplicaciones de mensajería electrónica:

- Autenticación, integridad y no repudio (mediante el uso de firma digital).
- Privacidad y seguridad de los datos (mediante el uso de cifrado).

Para poder usar el protocolo S/MIME debe obtenerse e instalar en el equipo una clave/certificado individual, tanto de la autoridad certificadora (AC) interna como de una AC pública. Una buena práctica es usar claves privadas separadas (y certificados asociados) para firma y para cifrado.

SET

Es un protocolo estándar, independiente de los mecanismos de transporte, diseñado para proporcionar seguridad en **transacciones con tarjetas de crédito** realizadas a través de Internet (y en general en redes no seguras).

SET utiliza técnicas de certificados digitales y criptografía de clave pública para la autenticación de las entidades entre sí e intercambiar información de manera segura proporcionando confidencialidad en las operaciones de pago y órdenes de compra.

Firma digital

La firma digital es un **mecanismo**, en realidad **independiente de la red**, que permite al receptor de un mensaje, firmado digitalmente, asegurarse de quién es la entidad que ha enviado el mensaje (**garantías de autenticación del origen y no repudio**), y a la vez confirmar que el mensaje no ha sido alterado desde que fue firmado (**garantía de integridad**).

La firma digital es adecuada cuando es importante verificar la autenticidad y la integridad de los datos como elemento para detectar una posible falsificación y/o manipulación del contenido.

IPsec

IPsec es un conjunto de protocolos que proporcionan servicios de seguridad basados en criptografía, tanto a nivel de capa de red (IP), como a todos los protocolos superiores basados en IP (TCP y UDP, etc.).

Una ventaja de IPsec sobre SSL y otros protocolos de capas superiores es que una aplicación puede usar IPsec directamente sin hacer ningún cambio, mientras que para usar SSL y otros protocolos de niveles superiores pueden tener que modificar su código. IPsec se puede integrar sobre la versión actual de IPv4 y se incorpora por defecto en la versión de IPv6.

Los servicios de seguridad a los que está orientado IPsec son, por ejemplo:

- Cifrar el tráfico.
- Asegurar su integridad (garantizar que no ha sido modificado).
- Autenticar los extremos.
- Anti-repetición (proteger contra la repetición de la sesión segura).

Seguridad en redes inalámbricas

Las redes inalámbricas más extendidas son las basadas en la familia de estándares 802.11 (Ethernet inalámbrica), más conocidas como **redes wifi**, en las cuales tenemos dos elementos básicos: las **estaciones cliente** y los puntos de **acceso inalámbricos**.

En este tipo de redes, al ser el espacio abierto y las ondas de radio el medio utilizado para transmitir los datos, surgen nuevos riesgos de seguridad, pues las transmisiones están expuestas a la "escucha" fácil de estaciones ajenas.

Podemos decir que el hecho de ser transmisiones vía radio añade a las comunicaciones algunos **riesgos**, como por ejemplo:

- **Es fácil realizar ataques por inserción** de usuarios no autorizados o por suplantación del punto de acceso a la red, creando otro con emisión más potente.
- Un intruso dentro de la red **puede atacar directamente a una estación cliente**.
- Se pueden **duplicar las direcciones IP o MAC** de los equipos.
- Los puntos de acceso están **expuestos permanentemente a ataques**: por ejemplo para averiguar las "passwords" de la red.

Mecanismos y protocolos de seguridad en redes inalámbricas

Para intentar contrarrestar estos riesgos disponemos de varios **mecanismos y protocolos de seguridad** específicos para redes inalámbricas:

WEP

WEP ("*Wired Equivalent Privacy*") es un **sistema de compresión y encriptación** estándar implementado en la capa MAC. Se encripta el cuerpo y el CRC de cada trama 802.11 utilizando el algoritmo de encriptación RC4 de RSA Security y claves de hasta 128 bits. Las claves están formadas por una parte que tiene que configurar el usuario y otra que se genera de forma automática y se llama "vector de inicialización" y que sirve para obtener claves diferentes para cada trama que se transmite en la red.

Aunque el mecanismo requiere usar una contraseña, e inicialmente se pensaba que era seguro, enseguida se vio, sin embargo, que **WEP resulta vulnerable**, principalmente porque las claves permanecen siempre estáticas y los 24 bits del vector de inicialización resultan escasos, de forma que si un intruso puede recoger (leer) suficientes tramas, puede descifrar la llave compartida del protocolo en muy poco tiempo.

OSA

OSA ("*Open System Authentication*") es un **mecanismo de autenticación** para las peticiones de acceso recibidas en redes que usan el protocolo WEP, pero es **poco fiable** porque no realiza comprobación de la estación cliente y las tramas de gestión son enviadas sin encriptar. Con OSA un equipo con acceso a la red inalámbrica puede acceder a ella si usa el protocolo WEP y recibir ficheros NO encriptados.

ACL

ACL ("*Access Control List*") es un mecanismo soportado por la mayoría de los productos comerciales y está basado en la **separación de privilegios de los usuarios, delimitando permisos de acceso**. Utiliza para la autenticación la **dirección MAC** del cliente, creando una lista con las MAC autorizadas, permitiendo el acceso a las que se encuentran en la lista de control de acceso.

Es un método **muy poco** seguro, pues basta con "escuchar" (con un *sniffer*) la red para obtener una dirección MAC válida, que luego configurar en nuestro adaptador de red.

CNAC

CNAC ("*Closed Network Access Control*") simplemente **controla el acceso** permitiéndoselo a aquellas estaciones cliente que **conozcan el nombre de la red (SSID)**, utilizándolo a modo de contraseña (es decir, impide el acceso a los dispositivos que no conozcan la identidad de la red).

WPA

WPA ("Wireless Protected Access") surgió como una evolución de WEP con algunas mejoras, como el TKIP (*Temporal Key Integrity Protocol*), un protocolo para gestionar claves dinámicas que varía de forma autónoma la contraseña de cifrado cada cierto tiempo.

WPA realiza la **autentificación de usuarios mediante un servidor** que almacena las credenciales y contraseñas de todos ellos, y si no se desea usar el servidor se puede usar un sistema de clave pre-compartida similar al de WEP (que requiere introducir la misma clave en todos los equipos de red).

WPA2

WPA2 ("Wireless Protected Access 2") soluciona algunos problemas de vulnerabilidad de WPA e incorpora las características del estándar IEEE 802.11i (que no incluía WAP).

Como mejoras principales incorpora un **protocolo de autentificación** "Counter-Mode (CBC-MAC)", que se considera más seguro y reemplaza el algoritmo RC4 por el AES ("Advanced Encryption Standard"), que es uno de los mas seguros actualmente. Tiene el inconveniente de que no todos los *router* permiten ese tipo de cifrado y no es compatible con el sistema WAP.

La **contraseña debe ser suficientemente larga** (más de 20 caracteres) y ser difícilmente adivinable. Es la opción recomendada para redes domésticas o de pequeñas empresas.

WPA2 Enterprise

WPA2 Enterprise realiza el cifrado con **AES** y la autentificación con el protocolo 802.1X/EAP, con contraseñas aleatorias y muchos posibles sistemas de verificación (usuario + contraseña, tarjetas inteligentes, certificados digitales, etc.). Es la **opción recomendada para redes de grandes empresas** o corporativas.

WIDS

WIDS ("Wireless Intrusion Detection System") es un **sistema de detección de intrusos** inalámbrico. Monitoriza la banda de radio empleada por la red para detectar puntos de acceso no autorizados y reacciona frente a ataques a la red wifi.

Recomendaciones de seguridad en redes inalámbricas

Algunas recomendaciones de seguridad a la hora de implantar redes inalámbricas son:

1. **Instalar cortafuegos y mecanismos de autenticación** entre la red inalámbrica y la red cableada de la organización.
2. Los clientes de la red inalámbrica deben acceder utilizando SSH, VPN o IPsec, y **mecanismos de autorización, autenticación y encriptación de tráfico (SSL)**, idealmente aplicando un nivel de seguridad distinto según la aplicación accedida.
3. Utilizar **detectores de señales no deseadas** y estaciones de monitorización pasivas para detectar direcciones MAC no registradas o clonadas, o el aumento de tramas de re-autentificación.
4. Utilizar **WPA2** como mecanismo de seguridad.
5. **Inhabilitar el DHCP** para las redes inalámbricas, asignando IP fijas.
6. **Cambiar el SSID** por defecto (conocidos en general) de los puntos de acceso.
7. **Inhabilitar la emisión en broadcast** del SSID de la red (ocultar el nombre de la red).
8. **Reducir la propagación** de las ondas de radio de la red fuera del edificio.

Cifrado simétrico y asimétrico

Aunque vemos que, en general, hay muchos protocolos para proteger el envío de información a través de las redes, el mecanismo fundamental para garantizar la seguridad en las comunicaciones es el "cifrado" o "encriptado" de la información.

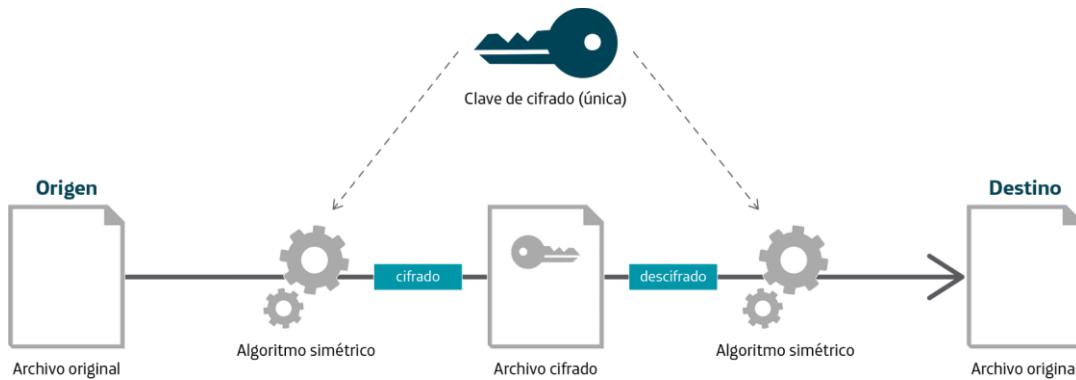
Al hablar de cifrado o encriptado nos referimos a aplicar un **algoritmo sobre la información a transmitir utilizando unas claves**, de forma que solamente alguien conocedor del proceso inverso (algoritmo) y de las claves, pueda "desencriptar" lo recibido y entender el mensaje.

Al mismo tiempo, suele servir para garantizar que nadie ha alterado la información a lo largo del camino, pues cualquier modificación haría que "la cuenta inversa" no funcionase.

A lo largo del tiempo la "criptografía" ha evolucionado mucho y podemos distinguir varios tipos de sistemas:

Criptografía simétrica

Los sistemas de cifrado simétrico utilizan la **misma clave para encriptar y desencriptar** la información, por lo cual el principal problema radica en disponer de un canal seguro para el intercambio de las claves (entre emisor y receptor), proteger su almacenamiento y también en que estas claves sean lo suficientemente robustas, lo cual depende básicamente de su complejidad y longitud.



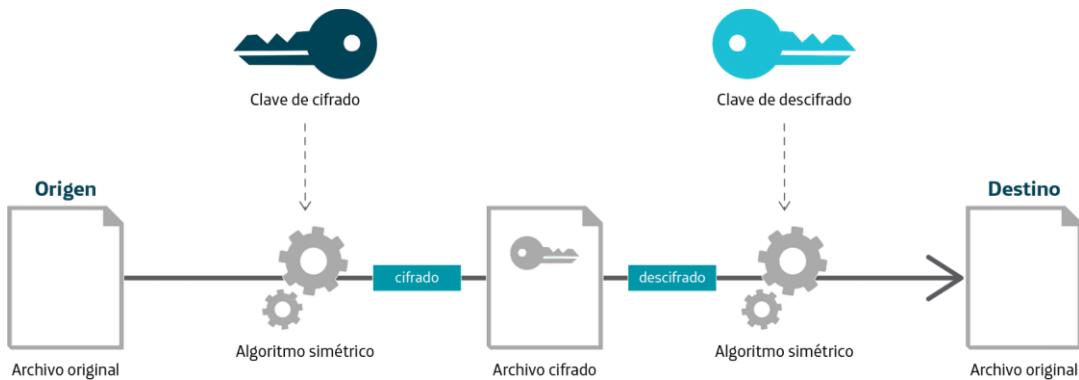
Criptografía asimétrica o de clave pública

Estos sistemas utilizan dos claves:

- Una **clave pública** que se puede enviar y ser conocida por cualquiera.
- Una **clave privada** que se guarda con seguridad y no se transmite ni se publica.

El sistema funciona utilizando algoritmos que permiten que la información sea cifrada usando una de las claves, y solamente pueda ser descifrada utilizando la otra. Con ello podemos conseguir varias cosas:

- Si queremos garantizar que la información recibida proviene de un determinado origen el emisor la encripta con una clave (privada) y publica la otra clave. Para descifrar el mensaje utilizaremos la otra clave, que garantiza que la información proviene de ese emisor.
- Si queremos enviar una información a un receptor y que nadie más pueda leerla podemos cifrar el mensaje con la clave pública del destinatario y solamente él podrá descifrarlo usando su clave privada.



Criptografía híbrida

La criptografía simétrica es más insegura y la asimétrica es más lenta. Por eso los sistemas híbridos intentan utilizar las ventajas de los dos anteriores. Consiste en utilizar un sistema de cifrado asimétrico para intercambiar de forma segura una clave (simétrica) que luego será usada para encriptar la información útil a transmitir.

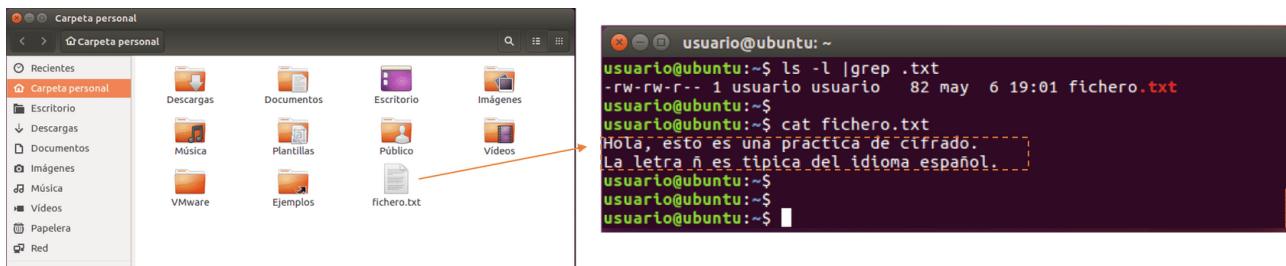
Ejemplo de cifrado simétrico

Veamos un par de ejemplos de cifrado simétrico y asimétrico, que pueden resultarte muy útiles. Empezaremos por el cifrado con claves simétricas.

Ejemplo de cifrado simétrico

En este caso, como ya sabes, la clave que se usa para encriptar la información y para desencriptarla es la misma. Por lo tanto tendríamos que asegurarnos que le comunicamos la clave empleada al receptor de la información por un medio seguro, o que por ejemplo nos hemos puesto de acuerdo de antemano para saber la clave a utilizar.

Empezaremos sobre nuestro sistema Linux Ubuntu, donde hemos creado en nuestra carpeta personal un archivo de texto al que llamamos "fichero.txt" (evidentemente el proceso que vas a ver sería extensible a cualquier otro archivo).



Para encriptar el fichero emplearemos una herramienta que viene preinstalada en muchas versiones de Linux, y está presente en nuestro Ubuntu. Se trata del programa "**gpg**" (GNUpg), que nos va a permitir utilizar claves simétricas y asimétricas. Primero veremos cómo cifrar el fichero con una clave simétrica, y lo mejor es que te lo mostremos en una imagen con la secuencia de comandos a emplear:

```

usuario@ubuntu:~$ ls -l | grep fichero
-rw-rw-r-- 1 usuario usuario 82 may 6 19:01 fichero.txt
usuario@ubuntu:~$ cat fichero.txt
Hola, esto es una práctica de cifrado.
La letra ñ es típica del idioma español.
usuario@ubuntu:~$ gpg --symmetric fichero.txt
usuario@ubuntu:~$ ls -l | grep fichero
-rw-rw-r-- 1 usuario usuario 82 may 6 19:01 fichero.txt
-rw-rw-r-- 1 usuario usuario 155 may 6 19:09 fichero.txt.gpg
usuario@ubuntu:~$ gpg -a --symmetric fichero.txt
usuario@ubuntu:~$ ls -l | grep fichero
-rw-rw-r-- 1 usuario usuario 82 may 6 19:01 fichero.txt
-rw-rw-r-- 1 usuario usuario 291 may 6 19:09 fichero.txt.asc
-rw-rw-r-- 1 usuario usuario 155 may 6 19:09 fichero.txt.gpg
usuario@ubuntu:~$ cat fichero.txt.asc
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1
jA0EBwMCZHBZJ5yPV2Jg0ooB2zbT2t4zhh3m7CQBcIvBmL0zw0C/U5W84WSmr1IB
wi42CkP5KK0P9guLn8H3c0gBD0dpH3Hns4Nho/hxIwWeqieREvxRwBotiyHL0c7D
MxVyo9MDbT1nrB3k3PEka7d/ubio6SBjsFZwxKcKG1i/DTLBzaRp5/KKGPFLGkYgU
TGK14R/WvY5hjZA=
-----END PGP MESSAGE-----
usuario@ubuntu:~$ cat fichero.txt.gpg
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1
jA0EBwMCZHBZJ5yPV2Jg0ooB2zbT2t4zhh3m7CQBcIvBmL0zw0C/U5W84WSmr1IB
wi42CkP5KK0P9guLn8H3c0gBD0dpH3Hns4Nho/hxIwWeqieREvxRwBotiyHL0c7D
MxVyo9MDbT1nrB3k3PEka7d/ubio6SBjsFZwxKcKG1i/DTLBzaRp5/KKGPFLGkYgU
TGK14R/WvY5hjZA=
-----END PGP MESSAGE-----

```

Contenido sin cifrar del fichero de texto.

Con esta orden creamos el fichero binario "fichero.txt.gpg" utilizando cifrado simétrico. El comando al ejecutarse nos pide que introduzcamos una contraseña, que es la que usará para encriptar la información (y deberemos pasársela al receptor para que pueda decodificarla).

Con esta orden creamos el fichero ASCII "fichero.txt.asc" utilizando cifrado simétrico. El comando al ejecutarse nos pide que introduzcamos una contraseña. Utilizar ficheros cifrados ASCII puede ser interesante para poder enviarlos cómodamente por email, insertarlos en un doc. etc.

Contenido del fichero cifrado en formato ASCII.

Contenido del fichero cifrado en formato binario.

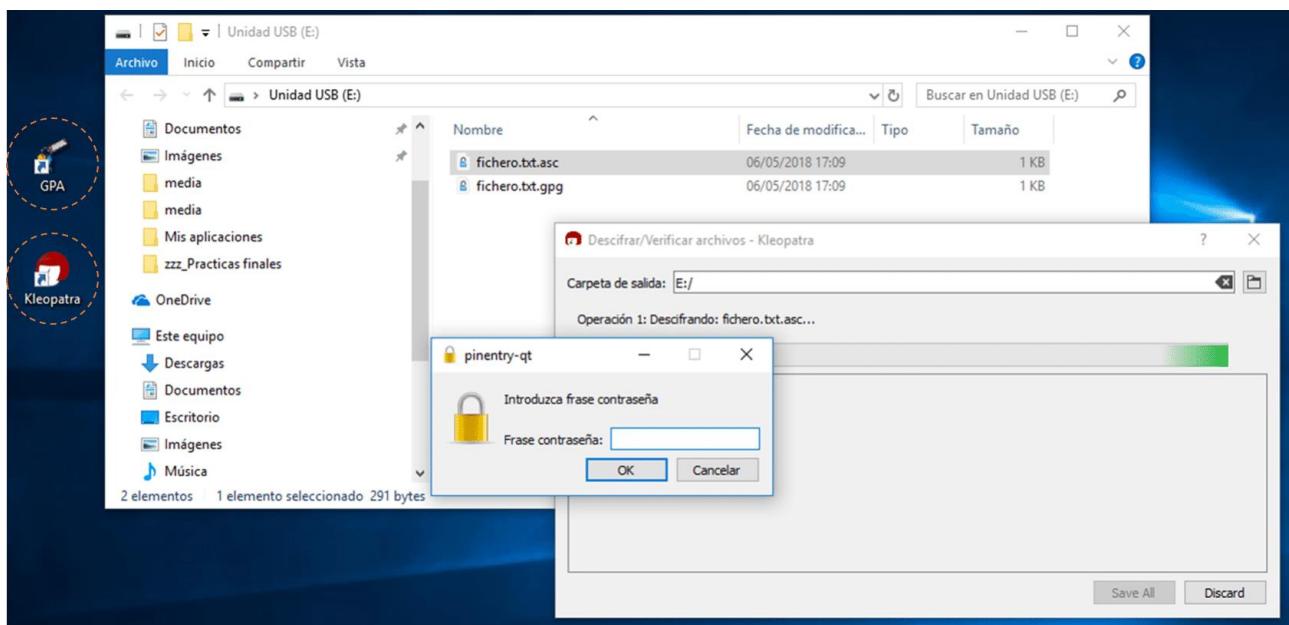
En el ejemplo anterior verás que hemos usado la encriptación para producir tanto un fichero binario ("fichero.txt.gpg") como uno con caracteres ASCII ("fichero.txt.asc"). Son dos opciones que te mostramos, pero en la práctica elegiríamos solamente una de ellas. Estos son los ficheros que enviamos al destino. Vamos a pensar que el destinatario tiene un sistema Windows.



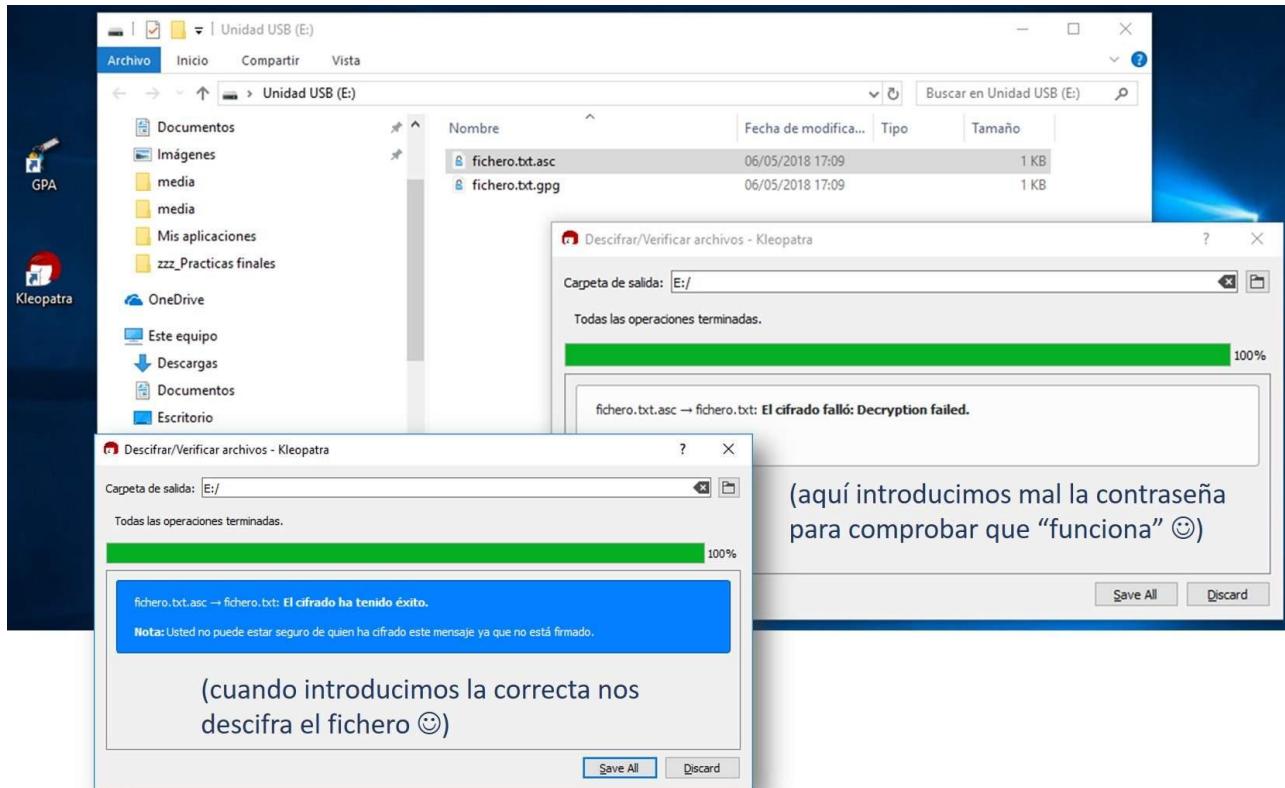
Verás que tal como llegan, encriptados, los ficheros no son entendibles. En la imagen de la izquierda lo que ve el destinatario (sistema Windows):

Tenemos que desencriptarlos. Para ello, en el sistema Windows 10 hemos instalado previamente la versión de [Gnupg para Windows](#), que es "**Gpg4Win**". La instalación (que es muy sencilla) nos dejará dos accesos directos, uno a "**GPA**" para gestionar las claves y otro a "**Kleopatra**" para manejar archivos encriptados.

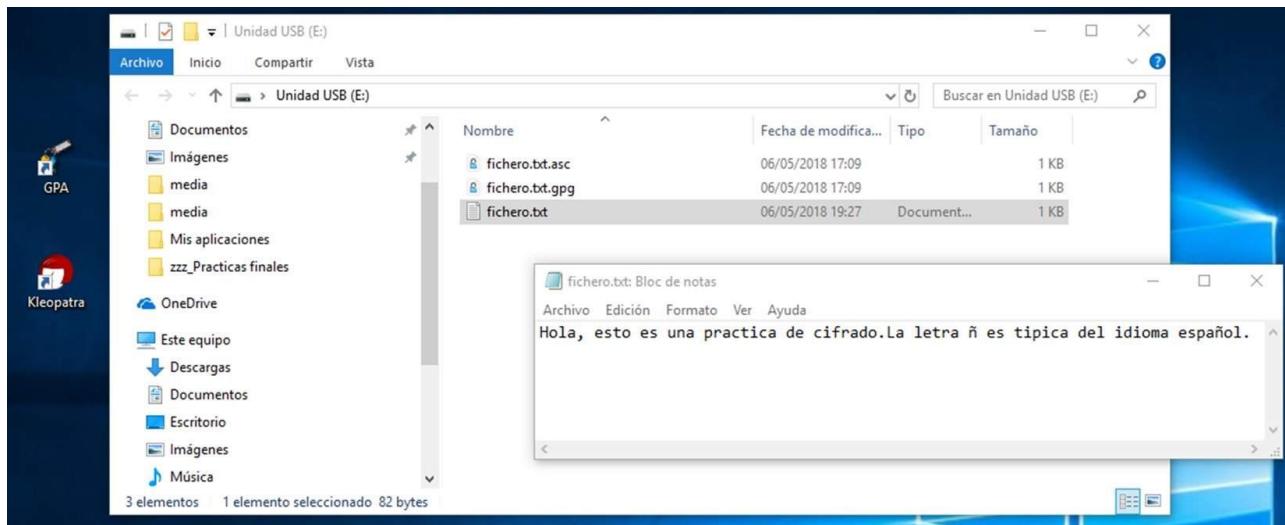
Pasamos los ficheros de un sistema a otro (en este caso a través de un USB, pero podría ser por email, repositorio compartido, etc.) y a intentar abrirlos, si tenemos instalado Gpg4win, vemos que Windows ya nos ofrece la posibilidad de hacerlo con Kleopatra y **nos pide la contraseña**.



Como vemos en la imagen siguiente, si introducimos una clave errónea nos dice que falla el descifrado del archivo. Si la introducimos bien nos permite salvar el archivo desencriptado.



Una vez descifrando, el archivo nos aparece como el fichero original (fichero.txt) en el directorio y podemos verlo (¡eureka!).

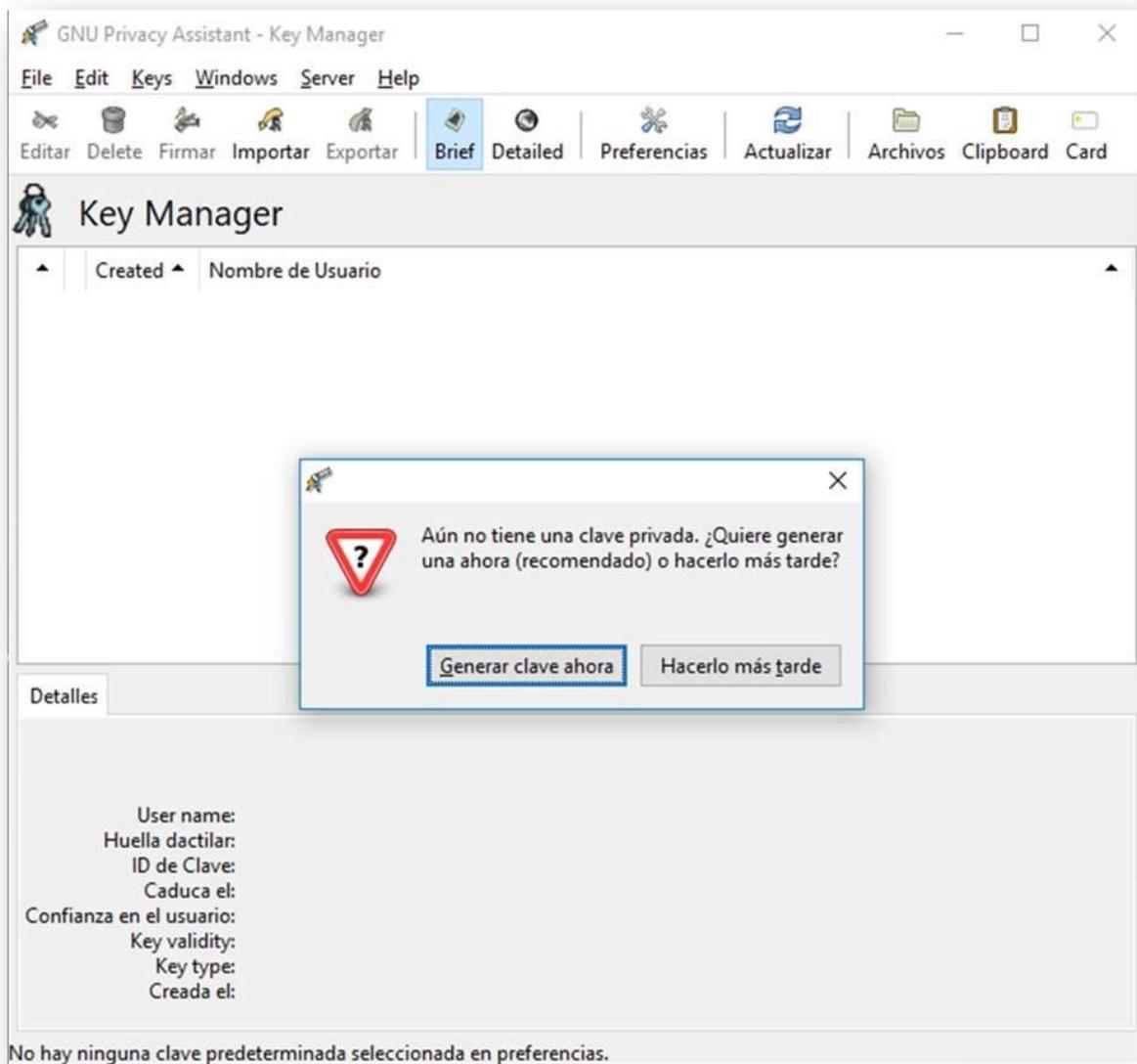


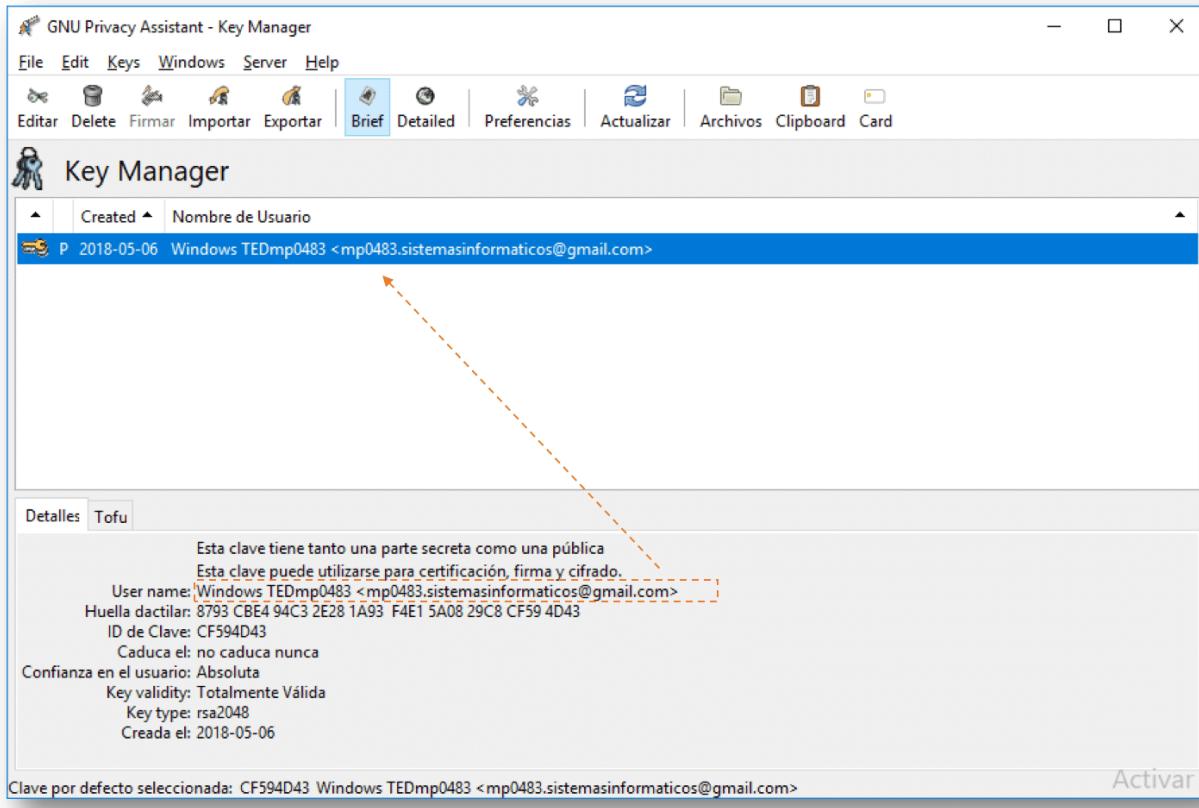
Ejemplo de cifrado asimétrico

Para utilizar claves asimétricas debemos distinguir dos procesos: por un lado la **generación de las claves** (privada y pública) y por otro el **encriptado y desencriptado** de la información.

Generación de las claves en Windows

Si utilizamos una herramienta gráfica, como puede ser Gpg4Win en el caso de Windows, el propio programa nos ofrecerá la posibilidad de crear nuestras claves pública y privada y nos guiará en el proceso. Te lo mostramos en las dos imágenes siguientes. Ten en cuenta que los datos del usuario y el email nos los ha pedido el programa durante el proceso de creación de las claves.





Nota: en este caso no necesitaríamos generar una pareja de claves en el sistema Windows, porque nos van a enviar una clave pública desde Ubuntu, pero lo hacemos para que veas lo sencillo que es y cómo se usa la interfaz gráfica de Gpg4Win.

Generación de las claves en Ubuntu

El proceso en Ubuntu es algo más laborioso al hacerlo con comandos de terminal, pero tampoco es muy difícil. Lo que hacemos básicamente es:

- Generar una pareja de claves (privada y pública).
- Exportar nuestra clave pública a un archivo para enviarla al otro usuario.

Te lo mostraremos en secuencia a través de imágenes.

```
usuario@ubuntu:~$ gpg -gen-key
gpg (GnuPG) 1.4.20; Copyright (C) 2015 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Seleccione el tipo de clave deseado:
 (1) RSA y RSA (por defecto)
 (2) DSA y ElGamal (por defecto)
 (3) DSA (sólo firmar)
 (4) RSA (sólo firmar)
¿Su elección? 1
las claves RSA pueden tener entre 1024 y 4096 bits de longitud.
¿De qué tamaño quiere la clave? (2048) 1024
El tamaño requerido es de 1024 bits
Especifique el periodo de validez de la clave.
 0 = la clave nunca caduca
 <n> = la clave caduca en n días
 <n>w = la clave caduca en n semanas
 <n>m = la clave caduca en n meses
 <n>y = la clave caduca en n años
¿Validez de la clave (0)? 0
La clave nunca caduca
¿Es correcto? (s/n) s

Necesita un identificador de usuario para identificar su clave. El programa
construye el identificador a partir del Nombre Real, Comentario y Dirección
de Correo electrónico de esta forma:
 "Heinrich Heine (Der Dichter) <heinrich@duesseldorf.de>"
```

Creamos pareja de claves privada y pública, y le vamos dando los parámetros que nos pide (tipo de algoritmo, tamaño de las claves, etc.)

```
Nombre y apellido: TEDmp0483
Dirección de correo electrónico: mp0483.sistemasinformaticos@gmail.com
Comentario: Claves asimétricas para el curso de SS00
Ha seleccionado este ID de usuario:
 «TEDmp0483 (Claves asimétricas para el curso de SS00) <mp0483.sistemasinformaticos@gmail.com>»
```

Como podemos tener más de una pareja de claves, hemos de darle un identificador y asociarlas a un email.

```
?Cambia (N)ombre, (C)omentario, (D)irección o (V)ale/(/S)alir? V
Necesita una contraseña para proteger su clave secreta. 
Repita contraseña: ■
```

El sistema también nos pide una contraseña de gestión, necesaria para cualquier cambio posterior y para poder autorizar el uso de la clave privada.

```
Es necesario generar muchos bytes aleatorios. Es una buena idea realizar
alguna otra tarea (trabajar en otra ventana/consola, mover el ratón, usar
la red y los discos) durante la generación de números primos. Esto da al
generador de números aleatorios mayor oportunidad de recoger suficiente
entropía.

No hay suficientes bytes aleatorios disponibles. Haga algún
otro trabajo para que el sistema pueda recolectar más entropía
(se necesitan 217 bytes más).
+++++
.....+
Es necesario generar muchos bytes aleatorios. Es una buena idea realizar
alguna otra tarea (trabajar en otra ventana/consola, mover el ratón, usar
la red y los discos) durante la generación de números primos. Esto da al
generador de números aleatorios mayor oportunidad de recoger suficiente
entropía.
.....+
....+
gpg: clave 91605BD1 marcada como de confianza absoluta
claves pública y secreta creadas y firmadas.
```

El proceso de creación de las claves usa la información de la actividad del sistema como fuente de entropía para los cálculos. Simplemente tenemos que permitirle que "piense" y mientras hacer cualquier otra actividad.

```
gpg: comprobando base de datos de confianza
gpg: 3 dudosas(s) necesaria(s), 1 completa(s) necesaria(s),
modelo de confianza PGP
gpg: nivel: 0 _validez: 1 _firmada: 0 _confianza: 0-, 0q, 0n, 0m, 0f, 1u
pub [1024R/91605BD1] 2018-05-06
  Huella de clave = F190 66D0 28CA 18CE 0529 031E 3ADB 9160 5BD1
uid          TEDmp0483 (Claves asimétricas para el curso de SS00) <mp0483.sistemasinformaticos@gmail.com>
sub [1024R/BFDCC8E7C] 2018-05-06

usuario@ubuntu:~$
```

Finalmente nos informa que ha creado las claves dentro de nuestro "anillo de claves" (BD de confianza en nuestro sistema). Cada clave tendrá un identificador (alfanumérico) que nos servirá también para realizar acciones con ella o sobre ella (ej borrarla, exportarla, etc.). La "pub" es la pública y la "sub" la privada o secreta.

```
usuario@ubuntu:~$ gpg --list-keys
/home/usuario/.gnupg/pubring.gpg   Visualizamos las claves y tenemos una pública (pub) y otra privada (sub.)
pub  1024R/91605BD1 2018-05-06
uid          TEDmp0483 (Claves asimétricas para el curso de SS00) <mp0483.sistemasinformaticos@gmail.com>
sub  1024R/BFDCC8E7C 2018-05-06

usuario@ubuntu:~$ gpg -a --export -o miclave.pub TEDmp0483
usuario@ubuntu:~$ ls -l | grep miclave
-rw-rw-r-- 1 usuario usuario 1085 may  6 20:46 miclave.pub
```

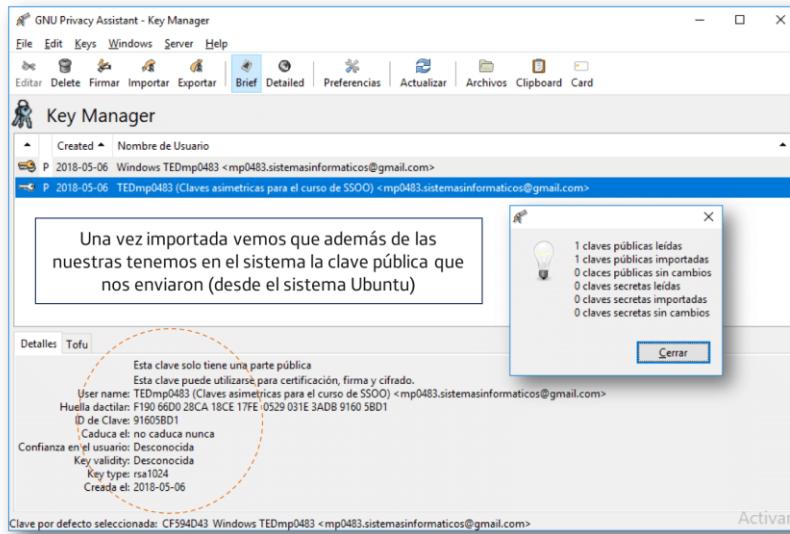
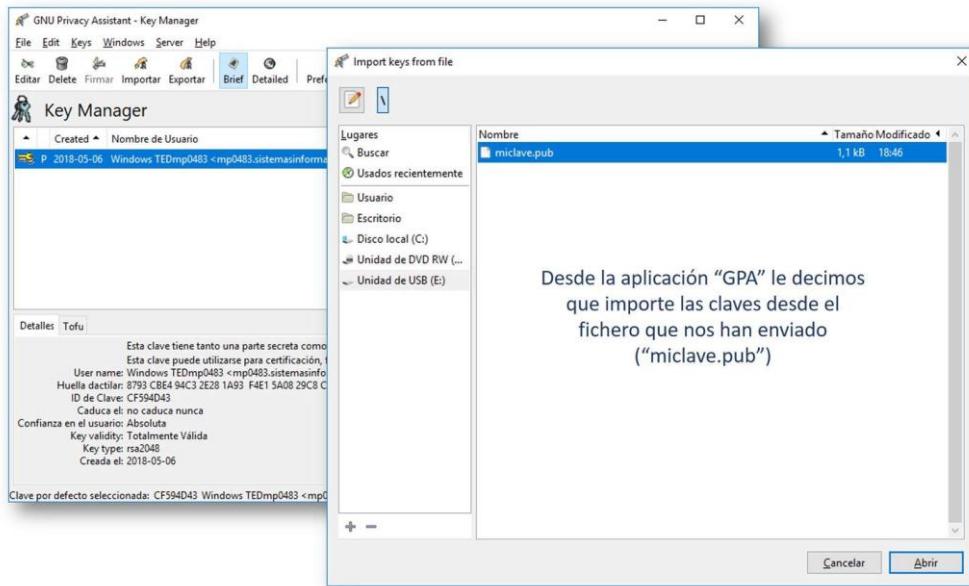
Con este comando exportamos nuestra clave pública al fichero "miclave.pub" y se la haremos llegar al otro usuario (o la publicaremos).

```
usuario@ubuntu:~$ pwd
/home/usuario
usuario@ubuntu:~$
```

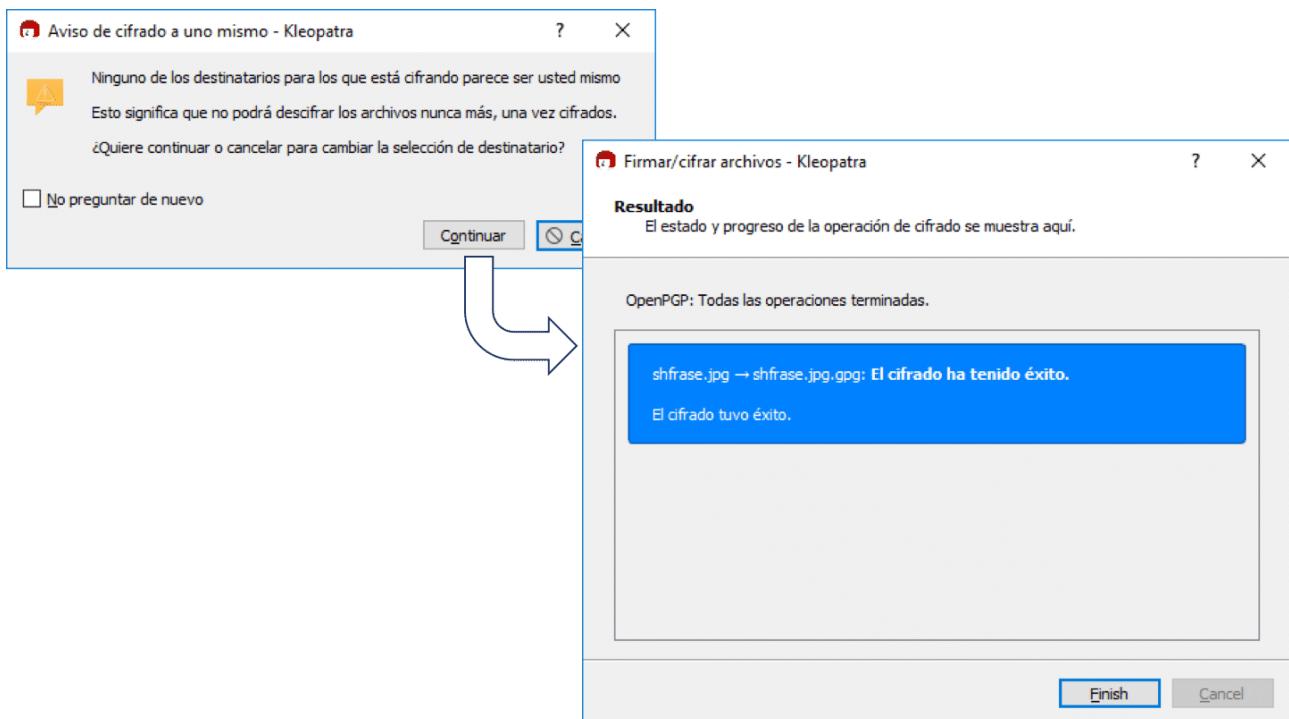
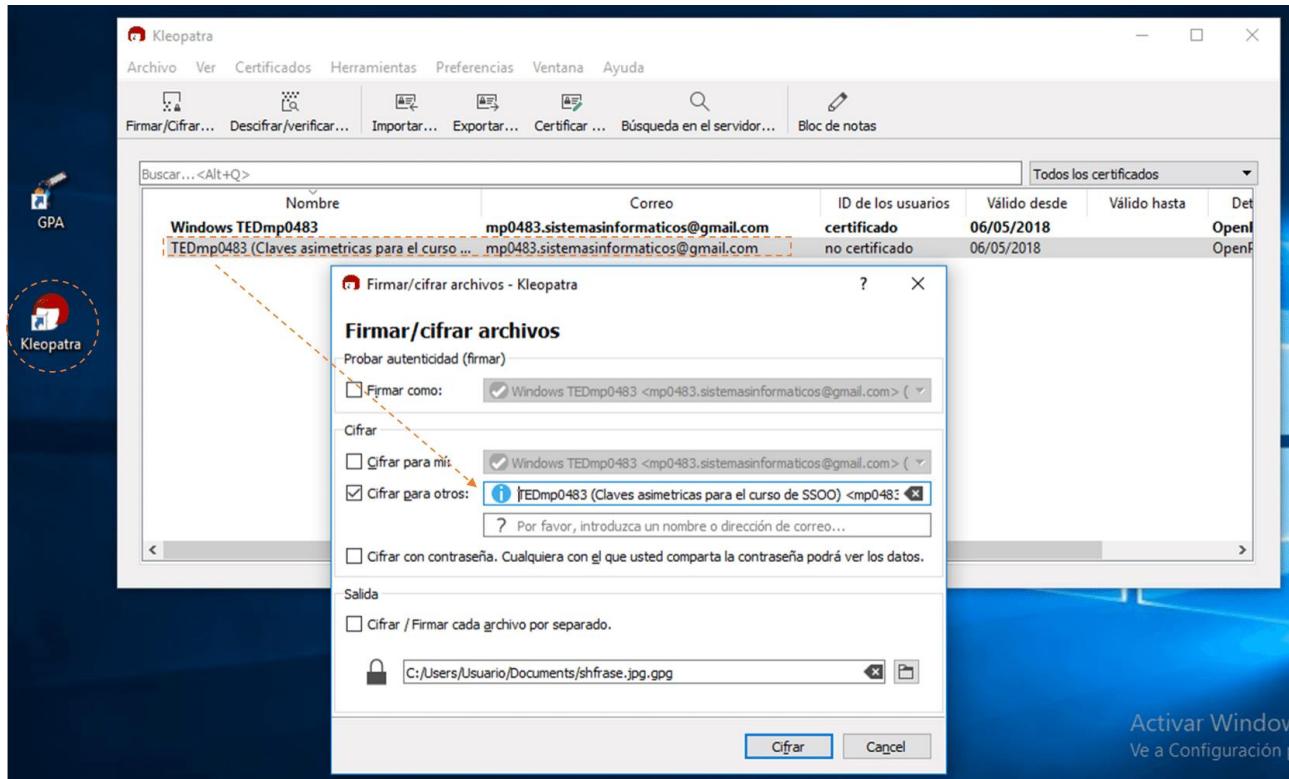
Una vez generadas ambas claves (privada y pública) y exportada la clave pública a un archivo, le enviaríamos al otro usuario nuestra **clave pública** para que nos envíe información cifrada con ella. Podemos hacerlo por un medio normal porque solo le servirá para cifrar la información, pero nunca para descifrarla. Otra opción sería publicar nuestra clave pública en un **servidor** para que la utilizase el que quisiera enviarnos información cifrada a nosotros.

Importación de claves y cifrado

En el otro extremo, supongamos que estamos en un sistema Windows, una vez recibido el archivo que la contiene, deberemos importar la clave pública a nuestro "anillo de claves" antes de poder utilizarla.



Una vez importada la clave (TEDmp0483) procedemos a cifrar con ella un fichero que queremos enviar al dueño de esa clave pública, y que solamente él podrá descifrar con su clave privada.



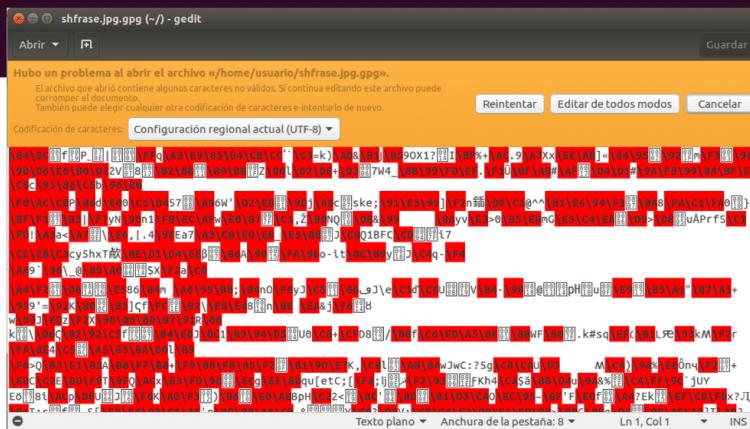
Ahora enviamos sin temor el fichero cifrado (en este ejemplo sería "shfrase.jpg.gpg") al dueño de la clave pública.

Recepción y descifrado

Al recibir el fichero ("shfrase.jpg.gpg") cifrado con nuestra clave pública, como solamente nosotros tenemos la pareja de esa clave (nuestra clave privada), seremos los únicos que podremos descifrarlo. Hagámoslo.

```
usuario@ubuntu:~$ ls -l | grep .gpg
-rw-r--r-- 1 usuario usuario 85576 may  6  2018 shfrase.jpg.gpg
usuario@ubuntu:~$
```

El fichero que el otro usuario ha encryptado usando nuestra clave pública nos llega cifrado e ilegible.



```
usuario@ubuntu:~$ ls -l | grep .gpg
-rw-r--r-- 1 usuario usuario 85576 may  6  2018 shfrase.jpg.gpg
usuario@ubuntu:~$
```

Con el mismo comando gpg y la opción "-d" podemos desencriptar el fichero (nos pedirá la contraseña de gestión que pusimos al crear las claves) y entonces podremos regenerar el fichero original.

Necesita una contraseña para desbloquear la clave secreta del usuario: "TEDmp0483 (Claves asimétricas para el curso de SS00) <mp0483.sistemasinformaticos@gmail.com>" clave RSA de 1024 bits, ID BFDC8E7C, creada el 2018-05-06 (identificador de clave primaria 91605BD1)

```
gpg: cifrado con clave RSA de 1024 bits, ID BFDC8E7C, creada el 2018-05-06
  «TEDmp0483 (Claves asimétricas para el curso de SS00) <mp0483.sistemasinformaticos@gmail.com>»
usuario@ubuntu:~$
```

total 232

drwxr-xr-x 2	usuario	usuario	4096	may 4	00:21	Descargas
drwxr-xr-x 2	usuario	usuario	4096	may 4	00:21	Documentos
drwxr-xr-x 2	usuario	usuario	4096	may 4	00:21	Escritorio
-rw-r--r-- 1	usuario	usuario	8980	may 4	00:07	examples.desktop
-rw-rw-r-- 1	usuario	usuario	82	may 6	19:01	fichero.txt
drwxr-xr-x 2	usuario	usuario	4096	may 4	00:21	Imágenes
-rw-rw-r-- 1	usuario	usuario	1085	may 6	20:46	mclave.pub
drwxr-xr-x 2	usuario	usuario	4096	may 4	00:21	Música
drwxr-xr-x 2	usuario	usuario	4096	may 4	00:21	Plantillas
drwxr-xr-x 2	usuario	usuario	4096	may 4	00:21	Público
-rw-rw-r-- 1	usuario	usuario	93082	may 6	21:53	shfrase.jpg
drwxr-xr-x 2	usuario	usuario	4096	may 4	00:21	Vídeos
drwxrwxr-x 3	usuario	usuario	4096	may 4	23:40	VMware



¡Vaya, el mensaje era una imagen!

Acceso a redes de área amplia

Sabemos que una red de área amplia puede tener ámbito nacional o internacional.

Históricamente la mayor de las redes de área amplia fue la conocida como "red telefónica", que estaba especializada en servicios de voz y, aunque podía transportar datos, lo hacía a muy baja velocidad.

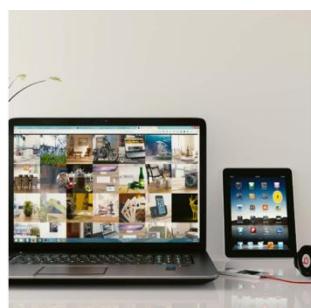
Con la extensión de los protocolos de datos (antes de IP) empezaron a existir otras **redes especializadas en el transporte de datos**.

Pero estas redes eran en gran medida manejadas por profesionales de las telecomunicaciones. Eran por ejemplo las redes **X.25** o **Frame Relay**, la red **ATM** o las redes de área metropolitana, en su mayor parte desconocidas para el público en general.

Años más tarde la red telefónica mejoró sus capacidades con la aparición de la **RDSI** (Red Digital de Servicios Integrados), pero en pocos años se vio superada por nuevas tecnologías de aprovechamiento del par de cobre (las tecnologías **xDSL**) y las nuevas redes de acceso.



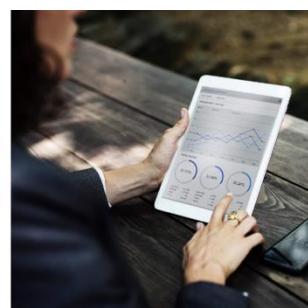
Toda esta evolución ha tenido (y tiene) un gran motor, que es sin duda el fenómeno **Internet**. En pocos años las tecnologías de acceso se han modernizado para mejorar el ancho de banda de acceso a "La Red" y por tanto soportar mejores servicios. La información de voz, datos o multimedia ya se manda unificada sobre la pila de protocolos TCP/IP, y en la actualidad la actividad profesional y personal no se entiende en el mundo desarrollado sin pasar por la conexión a Internet.



Ordenador portátil y tableta.



Smartphone.



Tabletas y kindles.



Smartwatch.

Tecnologías de acceso

En toda esta evolución tecnológica el número de tecnologías de acceso ha aumentado enormemente. Aunque algunas de ellas no sean predominantes en nuestro entorno, sí lo son en otros. Para hacer una clasificación distinguiremos básicamente entre dos tipos:

- Cuando el usuario accede (se conecta) a través de un "cable" (del tipo que sea).
- Cuando nos conectamos vía radio.

ACCESO POR LÍNEA (CABLE)

Tecnologías de acceso por línea (cable)

- Acceso por par de cobre (**ADSL** y la familia **xDSL** en general, par telefónico, RDSI).
- **Redes HFC** de acceso por cable (híbridas fibra-coaxial): las que conocemos de los operadores de cable que nos dan TV + Telefonía + Internet.
- Acceso a través de la red eléctrica (**PLC**): con más presencia en unos países que en otros.
- Acceso por **fibra óptica**: el de mayor capacidad y "puro" (solo fibra) o mixto, con alguna de las soluciones anteriores. Hoy en día es el que más se ofrece por parte de los operadores.

ACCESO POR RADIO (INALÁMBRICO)

Tecnologías de acceso por radio (inalámbrico)

- Acceso radio inalámbrico (WLL, LMDS, MMDS) punto-punto o punto-multipunto.
- Redes LAN inalámbricas (WLAN, **WIFI**, **WIMAX**, HiperLan2).
- Redes de telefonía móvil a partir de la segunda generación (2G GSM/GPRS, 3G UMTS, **4G**,...).
- Redes de acceso vía satélite (muy importantes en zonas alejadas de los grandes núcleos de población).

Principios y medidas de seguridad

Para **garantizar la seguridad** en nuestras comunicaciones existen **cuatro principios fundamentales** a cumplir. Pueden ser aplicados, tanto a los datos intercambiados por la red, como al servicio de comunicaciones y los equipos en sí mismos.

Los cuatro principios de seguridad:

1. **Confidencialidad**: que la información no esté disponible para terceros no autorizados.
2. **Integridad**: garantizar que la información no es modificada de forma indeseada (accidental o ilícitamente).
3. **Disponibilidad**: que la información y/o el servicio se encuentran disponibles para los usuarios en el momento y forma establecidos.
4. **Autenticidad**: garantía de que la información entregada en el destino es reflejo fiel de la generada en el origen.

Pero para poder entender estos principios debemos definir los **siguientes conceptos**:

Vulnerabilidad

Propiedad del sistema (en su conjunto) o de la red, que consiste en que es susceptible de ser atacado o de sufrir un fallo que dañe su funcionamiento, los equipos o los datos manejados.

Amenazas

Posibles factores (naturales, fortuitos o atacantes) que pueden afectar al sistema o a la red en alguna de sus vulnerabilidades.

Contramedidas

Acciones que podemos realizar para hacer frente a las amenazas, ya sean medidas **preventivas** (las que realizamos antes de que ocurra nada) o **de contingencia** (las que se realizan solo en el caso de que se manifieste la amenaza).

Mecanismos de seguridad

Los requisitos de funcionamiento de la red y los equipos conectados a ella deben cumplir las **prestaciones** en los siguientes aspectos:

- Confidencialidad.
- Autenticación.
- Integridad.
- Autorización y control de acceso.
- Privacidad y anonimato.
- Disponibilidad del servicio.
- Limitación de eventos.
- Informes de eventos (registro de ataques, fallos, etc.).

Los mecanismos a seguir, además, podrán adoptar alguna de las siguientes **estrategias de actuación**:

- Prevención.
- Informe.
- Limitación.
- Restablecimiento.
- Disuasión.

Como vemos, el ámbito de la seguridad va más allá de la simple ocultación o cifrado de la información a transmitir, teniendo que encargarse de garantizar en lo posible la integridad de todo el sistema de comunicaciones.

¿Qué medidas podemos tomar?

Pueden ser variadas y normalmente no tomaremos solamente una de ellas, sino que debemos diseñar cuáles son las más apropiadas para garantizar que nuestro sistema esté protegido.

MEDIDAS FÍSICAS

Son **mecanismos para impedir el acceso físico no autorizado** a los elementos y la infraestructura de la red. También pueden incluir medidas de protección contra desastres naturales, accidentes (p. ej. incendios) o condiciones adversas.

Normalmente incluyen el establecimiento de un perímetro de seguridad alrededor de instalaciones críticas, con mecanismos de protección contra:

- El acceso de personas no autorizadas.
- Protección contra daños físicos y accidentes (por ejemplo puertas y muros de blindaje, armarios ignífugos en los cuartos de comutadores, etc.).
- Medidas de recuperación en caso de fallo (redundancia en las conexiones, equipos de conmutación y los medios de almacenamiento físico, etc.)

MEDIDAS LÓGICAS

Son mecanismos de protección lógica contra el acceso a los recursos y la información almacenada o transmitida, por ejemplo:

- Política de control de accesos que exija la identificación y autentificación de los usuarios (p. ej. uso de **identificadores** ("user-id") y **claves** ("passwords").
- Medidas de **backup** de configuración y copia de respaldo del software de provisión del servicio.
- **Criptografía** para la protección de los datos enviados a través de la red.
- Uso de cortafuegos ("**firewalls**") para proteger la red interna del exterior.
- Procedimientos de **verificación, monitorización** ("logging") y **auditoría** ("auditing") de toda la red.

MEDIDAS LEGALES

Aunque pueda parecer menos importante, deben tenerse en cuenta aquellos aspectos que tienen que ver con las posibles vulnerabilidades "humanas". Es bastante común que en los centros de datos y de conmutación de las redes existan una serie de normas y compromisos a cumplir por parte de las personas que operan la red. En general los responsables de su administración se encargarán de:

- Publicación de la **política de seguridad** y de las medidas tomadas para ponerla en práctica.
- **Documentación** clara y actualizada sobre la red y establecimiento de su nivel de seguridad y acceso.
- **Definición de las funciones y responsabilidades** de los diferentes tipos de usuarios y administradores.
- Establecimiento de un **plan de formación** para asegurarnos de que todos conocen lo que tienen que hacer y cómo hacerlo, que disponen de los conocimientos técnicos necesarios y de los compromisos sobre los que responden en cuanto a seguridad.

MEDIDAS ADMINISTRATIVAS

Podríamos pensar en ellas como simple “trabajo administrativo”, pero son esenciales y a menudo trascienden al ámbito de la empresa y **suelen derivar de la legislación vigente**.

Sirven también para evitar un mal uso de las instalaciones, a veces para disuadir a un posible atacante o para informarle de las consecuencias de su ataque, pero también para asegurar que el sistema cumple con todo lo exigido para la operación en su entorno.

Un ejemplo pueden ser las medidas de cumplimiento del "Reglamento General de Protección de Datos" (**RGPD**) o la "Ley de Servicios de la Sociedad de la Información y Comercio Electrónico" (**LSSICE**).

Riesgos en las comunicaciones

Ahora vamos a repasar a qué peligros estamos expuestos cuando usamos las redes, ya sean cableadas o inalámbricas.

En particular, al usar las redes y transmitir información a través de ellas, existen algunas **amenazas** que básicamente tienen que ver con:

INTERCEPCIÓN

Alguien no autorizado accede a un recurso o a los datos de la transmisión (afecta a la confidencialidad de la información).

INTERRUPCIÓN

Se impide que la información llegue a su destino (afecta a la disponibilidad de los datos).

MODIFICACIÓN

Los datos son alterados y llegan a su destino con valores no originales (afecta a su integridad).

GENERACIÓN

Un agente externo inserta información en la red (afectaría a la integridad y la autenticidad).

Amenazas pasivas frente a amenazas activas

Además, dentro de estas amenazas podemos distinguir dos tipos de actuación:

- **Amenazas pasivas:** actúan observando (leyendo) el tráfico de información sin afectar a la funcionalidad.
- **Amenazas activas:** afectan de algún modo a la funcionalidad del servicio de la red.

EJEMPLOS DE AMENAZAS ACTIVAS

Las agresiones activas suponen una alteración en la comunicación normal, bien porque alteren el flujo de datos, inserten información no deseada en la transmisión, etc. Se subdividen en cuatro categorías:

- **Enmascaramiento:** cuando una entidad de la comunicación pretende ser otra diferente, por ejemplo para suplantar a un usuario válido, para obtener privilegios, etc.
- **Repetición:** captura pasiva de unidades de información y su retransmisión para producir efectos no deseados sobre el sistema (congestión, reconfiguración ilícita, etc.).
- **Modificación:** se alteran las características de la transmisión (p. ej. retardo), el tipo de tráfico (p. ej. prioridad) o el propio contenido de los mensajes (acceso a información).
- **Denegación:** se impide el uso del servicio de comunicaciones total o parcialmente, por ejemplo se sobrecarga la red o un servidor impidiendo su funcionamiento, o se interceptan y destruyen los mensajes dirigidos a la central de seguridad, etc.

EJEMPLOS DE AMENAZAS PASIVAS

Suelen ser muy difíciles de detectar, por lo cual nos enfocamos más en prevenirlas que en su detección. Algunos ejemplos pueden ser:

- **Divulgación del contenido:** la información transmitida es obtenida por medios ilícitos y suministrada a un tercero no autorizado (p. ej. escuchas telefónicas, copia de ficheros transmitidos por FTP, etc.). La técnica más común para proteger el contenido es el encriptado.
- **Análisis de tráfico:** no se trata aquí del contenido de los mensajes, sino del propio diálogo de la comunicación: origen, destino, identidades de email, frecuencia de las comunicaciones, etc. Todos estos datos pueden aportar una información valiosa para un tercero.

Ejercicio práctico: conexión SSH Windows-Linux

A modo de ejemplo vamos a ver cómo podríamos hacer una conexión segura utilizando el protocolo SSH entre un sistema Windows y otro Linux.

Para hacerlo utilizaremos tu sistema anfitrión y tu máquina virtual Linux Ubuntu. Como en toda conexión de este tipo necesitaremos un "cliente SSH" (lo usaremos sobre Windows) y un "servidor SSH" (en este caso sobre Ubuntu). Ambos, cliente y servidor, no suelen estar preinstalados en los sistemas, así que los cargaremos. Te señalamos el proceso paso a paso:

Ejecutar sobre un terminal de la máquina virtual de Ubuntu los siguientes pasos:

1. `sudo apt-get update`
2. `sudo apt-get upgrade`
3. `sudo apt-get install ssh` (con esto instalamos el servidor SSH).
4. `ifconfig` (para comprobar la dirección IP de nuestro sistema Ubuntu).
5. `netstat -a | grep ssh` (comprobamos que SSH está activo y escuchando).

```

usuario@ubuntu:~ 
usuario@ubuntu:~$ ifconfig
ens33    Link encap:Ethernet direcciónHW 00:0c:29:85:23:80
          Direc. inet:192.168.43.238  Difus.:192.168.43.255 Másc:255.255.255.0
          Dirección inet6: fe80::59bf:15c7:37e7:9505/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500 Métrica:1
          Paquetes RX:6095 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:3548 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:6749816 (6.7 MB) TX bytes:348779 (348.7 KB)
          Interrupción:19 Dirección base: 0x2000

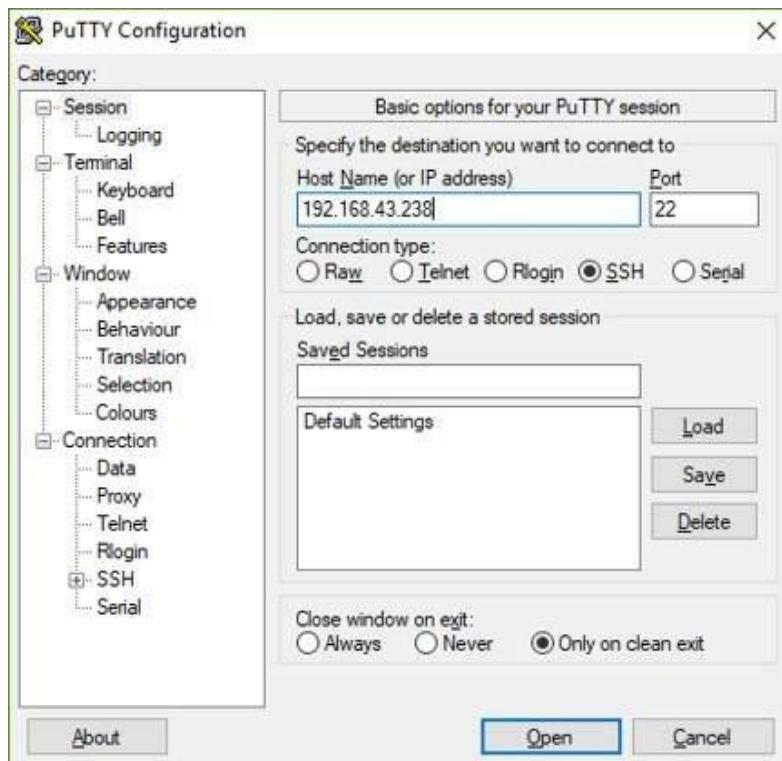
lo      Link encap:Bucle local
          Direc. inet:127.0.0.1 Másc:255.0.0.0
          Dirección inet6: ::1/128 Alcance:Anfitrión
          ACTIVO BUCLE FUNCIONANDO MTU:65536 Métrica:1
          Paquetes RX:1426 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:1426 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:136192 (136.1 KB) TX bytes:136192 (136.1 KB)

usuario@ubuntu:~$ 
usuario@ubuntu:~$ netstat -a | grep ssh
tcp        0      0  *:ssh                           *:*                  ESCUCHAR
tcp6       0      0  [::]:ssh                         [::]:*                ESCUCHAR
unix  2      [ ACC ]   FLUJO      ESCUCHANDO      23652    /run/user/1000/keyring/ssh
usuario@ubuntu:~$ 

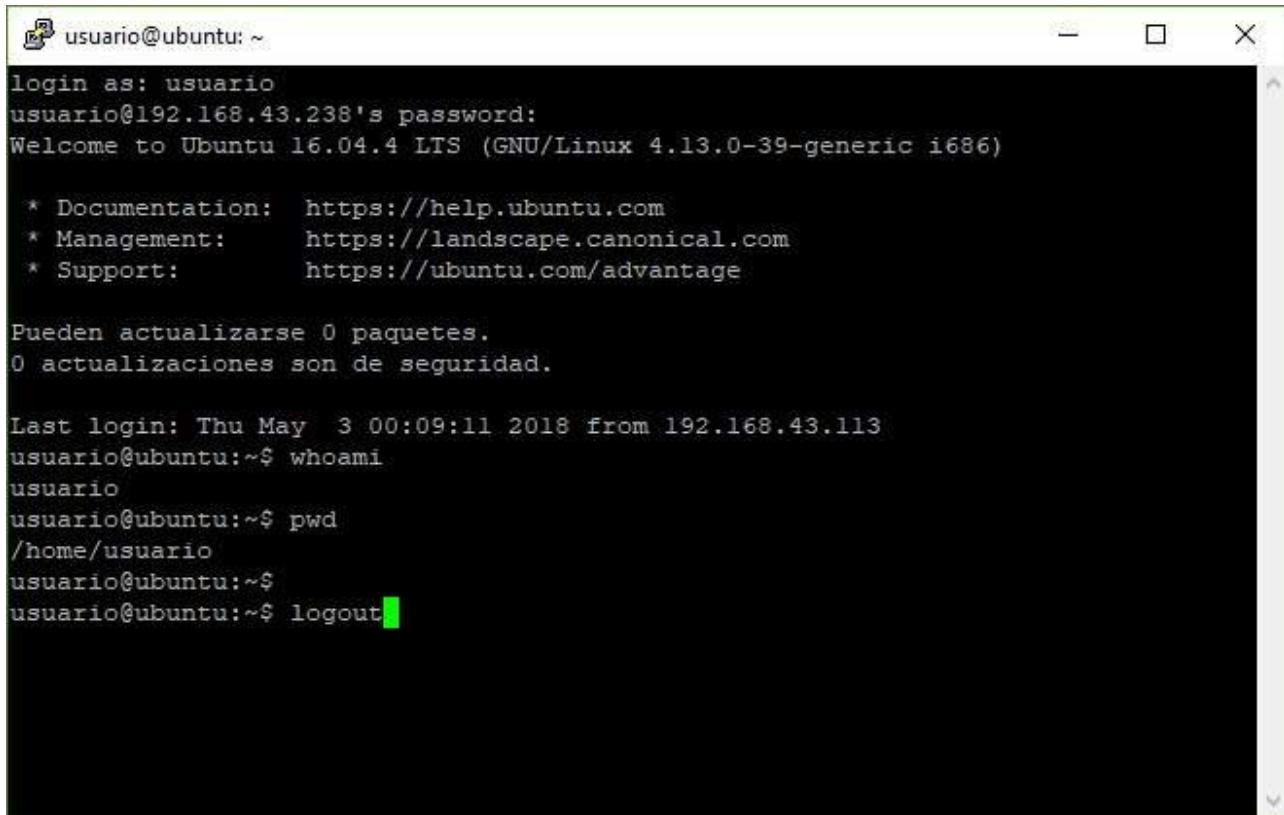
```

Ejecutar sobre tu sistema anfitrión Windows:

6. Descargar la aplicación “**putty.exe**” desde <https://www.putty.org/> (puedes descargar directamente el ejecutable o el instalador).
7. Ejecutar **putty.exe** y poner la dirección IP de la máquina virtual Ubuntu. Hacer clic sobre “Open” para establecer la conexión.



8. En la ventana que aparecerá, introducir usuario y clave de Ubuntu, y luego podemos probar comandos de Linux. Para terminar la conexión debemos teclear "logout", pero antes de hacerlo pasa al punto 9 (haremos algo antes de salir).



```
usuario@ubuntu: ~
login as: usuario
usuario@192.168.43.238's password:
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.13.0-39-generic i686)

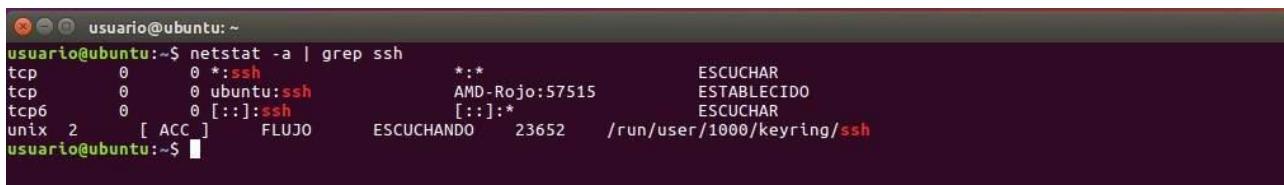
 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

Pueden actualizarse 0 paquetes.
0 actualizaciones son de seguridad.

Last login: Thu May  3 00:09:11 2018 from 192.168.43.113
usuario@ubuntu:~$ whoami
usuario
usuario@ubuntu:~$ pwd
/home/usuario
usuario@ubuntu:~$ 
usuario@ubuntu:~$ logout
```

Visualizar la sesión SSH en Ubuntu:

9. Mientras está la sesión de SSH activa, volver a insertar el comando "netstat -a | grep ssh" y vemos que en Ubuntu aparece como conectado el PC de Windows.



```
usuario@ubuntu: ~
usuario@ubuntu:~$ netstat -a | grep ssh
tcp        0      0 *:ssh          *:*                  ESCUCHAR
tcp        0      0 ubuntu:ssh      AMD-Rojo:57515    ESTABLECIDO
tcp6       0      0 [::]:ssh       [::]:*              ESCUCHAR
unix  2      [ ACC ]   FLUJO      ESCUCHANDO     23652    /run/user/1000/keyring/ssh
usuario@ubuntu:~$
```

10. Hacer "logout" sobre la ventana de SSH de putty (en Windows).

11. Volver a comprobar con "netstat" que SSH sigue activo, pero ya no tiene la conexión establecida.



```
usuario@ubuntu: ~
usuario@ubuntu:~$ netstat -a | grep ssh
tcp        0      0 *:ssh          *:*                  ESCUCHAR
tcp6       0      0 [::]:ssh       [::]:*              ESCUCHAR
unix  2      [ ACC ]   FLUJO      ESTABLISHED        23652    /run/user/1000/keyring/ssh
usuario@ubuntu:~$
```

Despedida

Resumen

Has terminado la lección, veamos los puntos más importantes que hemos tratado:

El tema de la seguridad en las comunicaciones es complejo, amplio y quizás es el de mayor importancia en la actualidad. Por ello, ser un especialista en seguridad requiere de un estudio profundo. En este módulo simplemente hemos pretendido trasladarte algunos conceptos básicos que te ayuden en tu aprendizaje.

Recuerda que, por norma general, es extremadamente importante que se realice una autenticación de los usuarios que van a usar el sistema o van a intercambiar información con él a través de las redes.

Esto afecta a la propia configuración de la red en la que nos encontramos, en la que podremos tener un *firewall* o una DMZ, pero también a nuestro propio equipo si lo contemplamos como un sistema al que se accede desde el exterior.

Por otro lado, los protocolos de comunicaciones actuales son múltiples y algunos son más seguros que otros. Disponer de la comunicación adecuada al nivel de seguridad y criticidad que exige la información a intercambiar es crucial; no es lo mismo ver un vídeo de YouTube que operar sobre nuestra cuenta en el banco. Y lo más importante, la actuación con bajos niveles de seguridad en unas tareas puede hacer que se infecte nuestro sistema con virus y agentes no deseados y que luego actúen cuando pensamos que estamos manteniendo comunicaciones seguras.