

Tema 1: Características de seguridad en la publicación de páginas web

Introducción

Cualquier desarrollador web debe ser consciente del material con el que trabaja y hacerse una serie de preguntas antes de publicar un sitio web:

- ¿Los datos que manipulo son sensibles y deben cumplir con la Ley Orgánica de Protección de Datos?
- ¿Tengo medidas de protección para mi equipo antimalware y antivirus?
- ¿Es necesario acceder al ordenador o sistema mediante credenciales de acceso (usuario y clave)?
- ¿Mi zona de trabajo está protegida para impedir el acceso de otros usuarios del equipo de trabajo?
- ¿El servidor destino o de producción cumple con las reglas mencionadas previamente?
- ¿El servidor dispone del software actualizado y adecuado?

Las respuestas podemos encontrarlas reflexionando sobre las siguientes premisas:

- El desarrollador debe contemplar la seguridad en los equipos en los cuales se desarrolle el producto y el lugar donde, finalmente, se instale el producto final del sitio web.
- En los equipos locales debe asegurar su sistema con software de seguridad como antimalware, antivirus, así como también un control de seguridad de acceso al equipo (login) a través de acceso con credenciales: nombre de usuario y clave.
- En cuanto al equipo de producción o hosting debe cumplir y exigir unas medidas de seguridad: primero de acceso, a través del usuario y clave, y luego a través de actualización de software, antivirus del servidor, así como una configuración adecuada del servidor.

Más información sobre la LOPD (Ley Orgánica de Protección de Datos) por la cual deben registrarse todo sistema web que se implemente dentro del territorio europeo.

<https://www.aepd.es/es/documento/reglamento-ue-2016-679-consolidado.pdf>

1.1. Seguridad en distintos sistemas de archivos

¿Qué es un sistema de archivos? Un sistema de archivos consiste en un método y una estructura de datos el cual un sistema operativo o sistema base utiliza para organizar los distintos archivos de un sistema de almacenamiento masivo y/o partición de este.

También suele referirse al tipo de método u organización de dicho sistema de archivos. Como, por ejemplo, FAT, EXT4, NTFS, etc.

En definitiva, los sistemas de archivos forman parte integral de los sistemas operativos (Windows, UNIX, GNU/Linux, Mac OS, etc.). Sin embargo, en la actualidad donde todos los

dispositivos están interconectados, el sistema operativo puede integrar elementos ajenos al propio hardware e integrarlos en el propio sistema de archivos para tratarlos como propios.

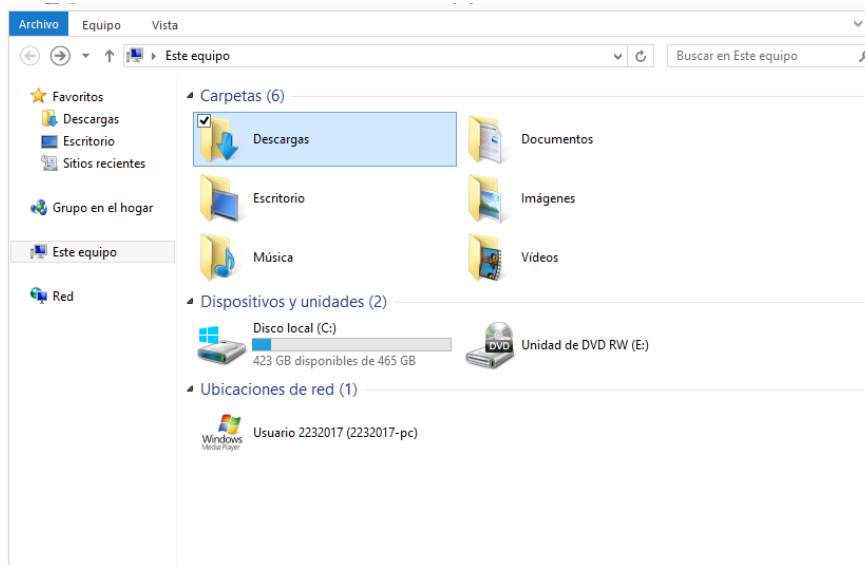


Figura 1.1.

En la imagen 1.1 podemos observar los dispositivos de almacenamiento correspondientes a un disco duro, así como también una unidad DVD de lectura-escritura y una unidad de red. Esta unidad hace referencia a zonas de almacenamiento de otro dispositivo conectado a la red local.

Una vez hecha esta pequeña introducción a los sistemas de archivos debemos comprender que igual que protegemos documentos en soporte papel también debemos proteger los archivos o datos con los que trabajamos. Más si cabe si existen datos sobre personas o entidades.

En general, la forma que tenemos para proteger nuestros archivos será: Mediante las herramientas del propio SO y mediante software de terceros.

Podemos mencionar como herramientas del SO la capacidad de este para discernir a un usuario y a otro a través de credenciales. Además, el administrador del equipo podrá seleccionar el perfil adecuado que permita limitar los privilegios de todos y cada uno de los usuarios. Por ejemplo, un usuario no debería poder acceder a la zona de otro usuario sin tener privilegios mayores que los de un usuario del sistema. Si el usuario en cuestión, se autentica a través de una red contra un servidor, los niveles de seguridad serán mayores.

Si se ha de compartir información se debe indicar quién puede acceder y con qué privilegios. Por ejemplo, imaginemos que compartimos una carpeta o directorio, pero debemos indicar qué usuarios pueden: leer, escribir y/o modificar.

El software de terceros que podemos utilizar para proteger nuestro equipo puede ser: cortafuegos, antivirus, antimalware, antispyware, etc.

Debemos tener en cuenta el desarrollo en paralelo de los sistemas distribuidos y redes de datos, comentado anteriormente, que ha dado lugar a la aparición de nuevos riesgos de seguridad relativos a la distribución de la información entre los sistemas informáticos y a la

necesidad de reforzar y adaptar al nuevo entorno los controles de seguridad de los sistemas individuales.

En la era actual, todos los ordenadores están interconectados generalmente, y la propagación de virus es mucho más rápida gracias también a la facilidad al acceso y distribución de la información.

Dada esta situación se hace necesario integrar en las arquitecturas de comunicaciones existentes las funcionalidades propias de la seguridad. Este proceso de integración necesariamente implicará la implementación de mecanismos y servicios con funciones de seguridad que se apoyarán en muchos casos en servicios, mecanismos y funciones ya implementados en la propia arquitectura de comunicaciones. El resultado final será la **Arquitectura de seguridad**.

El estándar ISO 7498-2 define un servicio de seguridad como el servicio proporcionado por un nivel de un sistema abierto que garantiza la seguridad de los sistemas abiertos o a las transferencias de datos en dichos sistemas. Estos servicios están divididos en cinco categorías y 14 servicios específicos. Las categorías son:

- **Autenticación:** Asegura que las entidades que se comunican son quienes dicen que son. El estándar ISO 7498-2 define dos servicios de autenticación específicos: Autenticación del origen de los datos y Autenticación de entidades pares.
- **Control de acceso:** el servicio de control de acceso evita el uso fraudulento de los recursos del sistema. Con este servicio se controla quien puede acceder a los recursos en general y en qué condiciones tendrá dicho acceso. Además, podrá limitarse ciertas condiciones a dichos recursos (horario de acceso, privilegios sobre el recurso, etc.).
- **Confidencialidad:** El servicio de confidencialidad asegura que la información no será divulgada o revelada ni estará disponible por: individuos, entidades u organizaciones, procesos o software no autorizados.
- **Integridad:** El servicio de integridad asegura que los datos recibidos son exactamente a como han sido enviados por una entidad autorizada. Es decir, sin duplicaciones o repeticiones, retransmisiones cortadas o continuadas de forma atípica, inserciones o modificaciones.
- **No repudio:** el servicio de no repudio evita que las entidades pares que se comunican entre sí puedan denegar una o ambas al haber participado en la comunicación.

Si queremos un sistema de archivos seguro debemos implementar sistemas, métodos y procedimientos que así lo permitan. Podemos mencionar varios métodos de los más utilizados para mantenerlo seguro: tendremos la asignación de permisos de acceso discrecionales según perfiles de usuarios, utilización de cortafuegos, antivirus, copias de seguridad, cifrado de información y encriptación de archivos y carpetas.

1.1.1. Sistema operativo Linux

Centrándonos en el sistema operativo Linux debemos, primero, indicar que no es un sistema operativo propietario. Encontraremos que hay distribuciones de pago (Red Hat, Suse,

Mandriva) y distribuciones bajo licencia GNU (Ubuntu, Fedora, Centos, OpenSuse, Debian). GNU significa (GNU is Not Unix).

Linux puede implementar diversas soluciones de sistemas de archivos como EXT3, EXT4, ReiserFS, XFS. Todos ellos con un sistema de **logs o journaling** consistentes en llevar a cabo un registro de diario en el que se almacena la información necesaria. Con este sistema se consigue restablecer los datos afectados por la transacción en caso de que esta falle. Por ejemplo, cuando el sistema se cae por falta de corriente eléctrica.

Algunas de las características básicas de estos sistemas operativos son las siguientes:

- **Libre:** cualquiera lo puede usar, distribuir y modificar.
- Tenemos distribuciones gratis pudiendo emplear tantas licencias como se desee.
- Desarrollado por miles de voluntarios en el mundo. Cualquiera puede participar y pertenecer a la comunidad.
- Código fuente abierto a todos.
- Alta estabilidad, por lo que es difícil que se quede colgado.
- Extremadamente seguro ya que tiene varios sistemas de protección. Su cortafuegos está incrustado en el propio núcleo del sistema.
- Facilidad de uso en muchas tareas.
- Lee y escribe en sistemas de archivos de Windows y Macintosh.
- Se comunica con cualquier otro sistema de red.
- Las distribuciones tienen diversos escritorios como Unity, Gnome, KDE, XFCE, LXDE.
- Necesita bajos requerimientos Hardware para poder ejecutarse.
- Ocupa poca memoria debido a la sencillez de UNIX.

En el proceso de instalación, distribuciones Linux como Debian solicitan clave para un usuario, llamado root, que será el administrador del sistema. Además, deberemos indicar las credenciales de un usuario (nombre de usuario y clave) que tendrá el acceso restringido al sistema. Es decir, si queremos realizar tareas normales de usuario utilizaremos el usuario introducido. Sin embargo, si queremos realizar tareas propias de un administrador de equipo, como instalaciones o paquetes, éstas deberán realizarse a través del usuario **root**.

Una de las ventajas del SO Linux es que cualquier usuario puede realizar tareas como administrador. Pero hay una dificultad, siempre necesitará las credenciales de root.

Aunque es un sistema con una fama de seguro no está libre de accesos indeseados, virus, spyware, etc. Con lo cual deberemos implementar mecanismos de defensa activa.

```
(base) javier@javier-VirtualBox:/etc$ sudo iptables -L
Chain INPUT (policy DROP)
target    prot opt source                destination
all -- anywhere             anywhere

Chain FORWARD (policy ACCEPT)
target    prot opt source                destination

Chain OUTPUT (policy DROP)
target    prot opt source                destination
```

Figura 1.2.

En lo que respecta al cortafuegos como se ha comentado previamente. Este es parte integrante del núcleo (kernel) de Linux. Cabe señalar las líneas DROP, con este parámetro conseguimos que las peticiones procedentes de las máquinas cuyas IPs coinciden con las marcadas no reciban respuesta. Es decir, de cara a ellas nuestro ordenador está apagado y fuera de cobertura. El valor ALL indica que las comunicaciones por cualquier puerto están negadas.

Es tal la potencia de IPTABLES que nos puede ayudar a “enrutar” todos los paquetes de una red filtrando el origen de los datos y dando paso según qué puerto y a dónde se dirigen.

Netfilter es un framework disponible en el kernel de Linux que permite interceptar y manipular paquetes de red.

El problema de configurar un cortafuegos en modo comando es la necesidad de aprender no solo el comando sino también todos los parámetros y valores posibles. Para una mayor rapidez en el proceso sería conveniente una aplicación gráfica que nos proporcione una vista mucho más intuitiva del proceso. En Linux tenemos ejemplos como son: Guarddog, Firestarter, KMyFirewall, FWbuilder y otros.

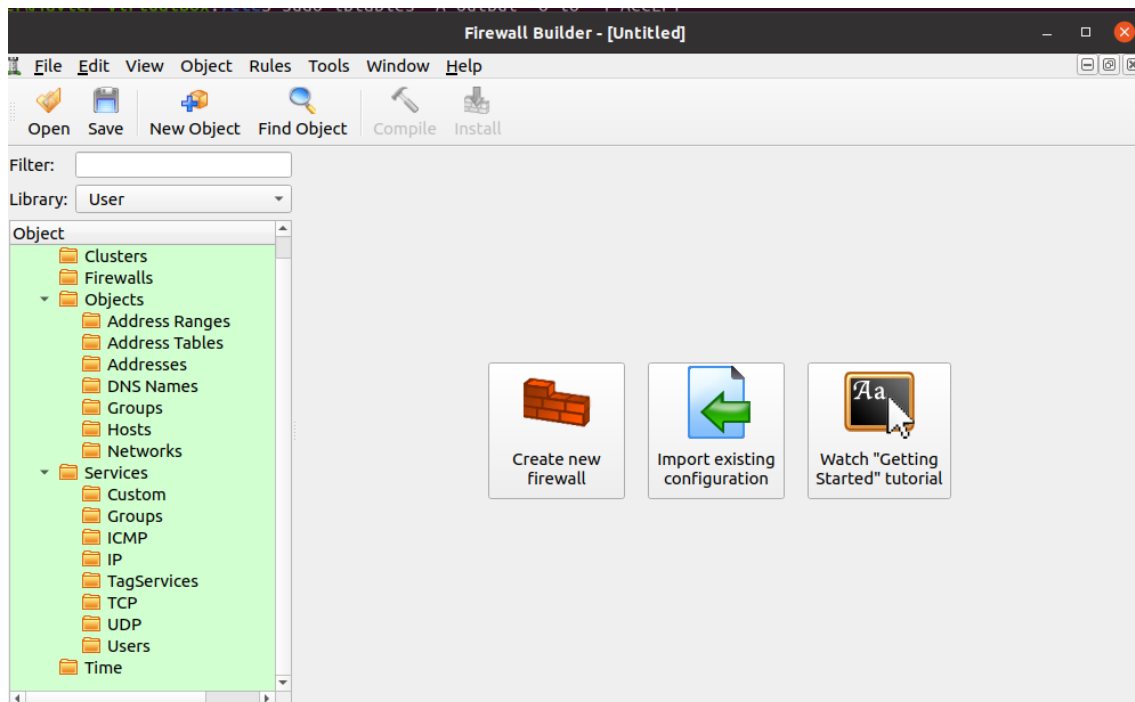


Figura 1.3.

En la figura 1.3 observamos la pantalla del paquete FWBuilder, una de las aplicaciones gráficas más conocidas. Debemos tener en cuenta que no es un cortafuegos sino una herramienta que nos permite generar un archivo con las reglas que nosotros queremos implementar para proteger nuestro equipo de ataques externos o de salidas de datos internas hacia el exterior. Es decir, cumple parte de las funciones que se pueden establecer con IPTABLES, si queremos emplear el archivo deberemos usar IPTABLES para indicar que aplique las reglas que hemos generado con esta herramienta.

Una herramienta muy útil relacionada con IPTABLES es FAIL2BAN. Esta herramienta nos permite indicar qué software debe ser protegido contra ataques por “fuerza bruta” indeseados. ¿Cómo funciona? En su archivo de configuración de reglas, /etc/fail2ban/jail.conf,

indicaremos el software que queremos proteger y cómo debe actuar. En el siguiente ejemplo observamos una porción de la configuración del archivo mencionado previamente y corresponde a cómo proteger el software de acceso remoto SSH por medio del puerto 23.

```
[ssh]
enabled=true
port = ssh
filter = sshd
action = iptables[name=SSH, port=23, protocol=tcp]
logpath=/var/log/auth.log
maxretry=6
```

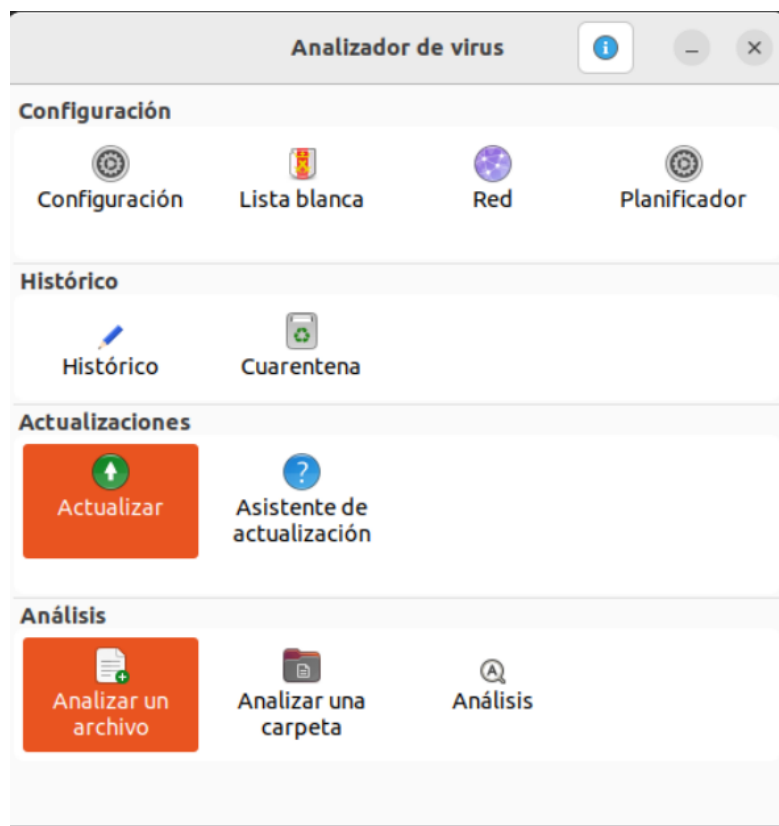


Figura 1.4.

Aunque existen antivirus suficientes para elegir el que consideremos más adecuado, en este caso se va a utilizar un antivirus libre y gratuito denominado CLAMAV. En la figura 1.5 observamos cómo se está trabajando con la aplicación gráfica ClamTK. En cuanto a malware y gusanos es muy complicado infectar un sistema Linux debido a sus niveles de seguridad. Pero cabe recordar que el hecho de que estos sistemas sean seguros no garantiza plena seguridad en los mismos.

1.1.2. Sistema operativo Windows

En Windows generalmente, el primer usuario que se crea tiene privilegios de administrador y es integrante del grupo de administradores.

Es aconsejable que haya más de un usuario con perfil de administrador y, a ser posible, no utilizarlo más que en casos de administración del SO.

Conviene generalmente tener dos usuarios administradores (al menos en Windows) ya que estos sistemas son centro de mucho malware y ataques. Se consigue que el centro del ataque sea el usuario de trabajo estando el usuario administrador (el que seleccionemos como tal), aparentemente, a salvo. En caso de ataque se puede arrancar el sistema “a prueba de errores”, acceder como administrador y realizar las tareas de mantenimiento que se estimen necesarias para arreglar el problema.

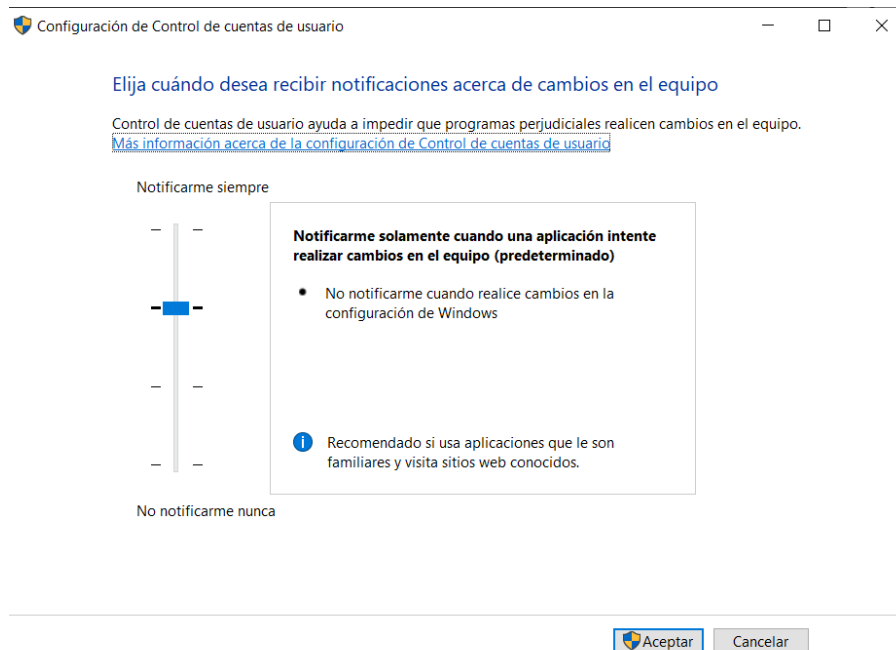


Figura 1.5.

El sistema operativo Windows tiene un sistema de control de usuarios pero, a diferencia de Linux, podemos manipular el nivel de seguridad que se aplicará a estos usuarios.

En el momento que creamos un usuario le indicamos al sistema en qué grupo estará integrado. Es importante definirlo pues determinará el perfil del usuario. Los grupos pueden ser: usuario estándar o administrador. No es aconsejable que todos los usuarios sean administradores. Lo habitual es que haya uno o dos usuarios administradores y el resto de los usuarios, usuarios estándar.

En la figura 1.5 observamos el UAC (User Access Control) de Windows que abre el panel para la gestión. Podremos acceder en Windows escribiendo **Cambiar configuración de Control de cuentas de usuario**, esta característica forma parte del panel de control.

El control consiste en indicarle el nivel de aviso de cambios en el sistema por parte de una aplicación o servicio. El aviso espera la reacción del usuario autorizando o no la realización de esos cambios. Este aviso no informa del alcance de los cambios.

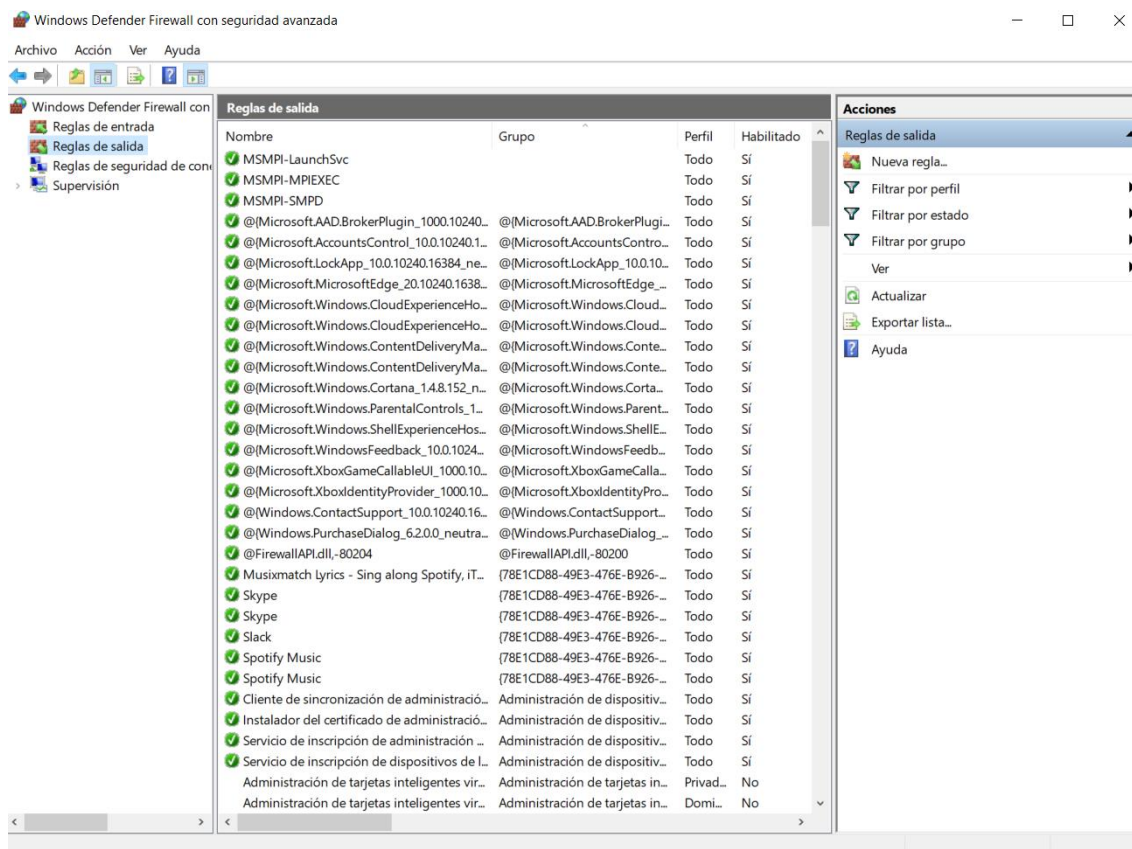


Figura 1.6.

En la figura 1.6 tenemos un panel de configuración avanzada del cortafuegos de Windows 10.

En el apartado de cortafuegos, desde la versión de Windows Vista, dispone de un cortafuegos integrado en el sistema, pero no está integrado en su núcleo (kernel) a diferencia de Linux, lo cual hace de un sistema Windows un sistema más vulnerable.

En cuanto a antimalware y antivirus de Microsoft ofrece, un antivirus propio y gratuito: Microsoft Security Essentials. Su potencia y fiabilidad es bastante cuestionable, aunque garantiza niveles básicos de seguridad. En Windows es mejor usar software de terceros para proteger de malware: como Kaspersky, Panda, Avast...

1.1.3. Otros sistemas operativos

Entre otros sistemas operativos distintos a Windows y Linux, encontramos el más conocido MacOS, así como también otros de gran uso como FreeBSD y SunOS.

Además, si hablamos de dispositivos móviles (smartphones), generalmente cada uno de los mismos ya integran persé sistemas operativos propios como pueden llegar a ser Android, iOS, Ubuntu Touch, Windows Phone, Kindle...

1.2. Permisos de acceso

Uno de los niveles de seguridad mencionados es, sin duda, el control de acceso. ¿Y qué medida de seguridad no lo es cuando lo primero que te piden es indetificarte? Esto es lo que ocurre en

la mayoría de SO que se ejecutan en los ordenadores pero no en otros dispositivos como smartphones en los que la parte de la seguridad del teléfono viene gestionada tanto por la tarjeta SIM como por el sistema operativo que integre.

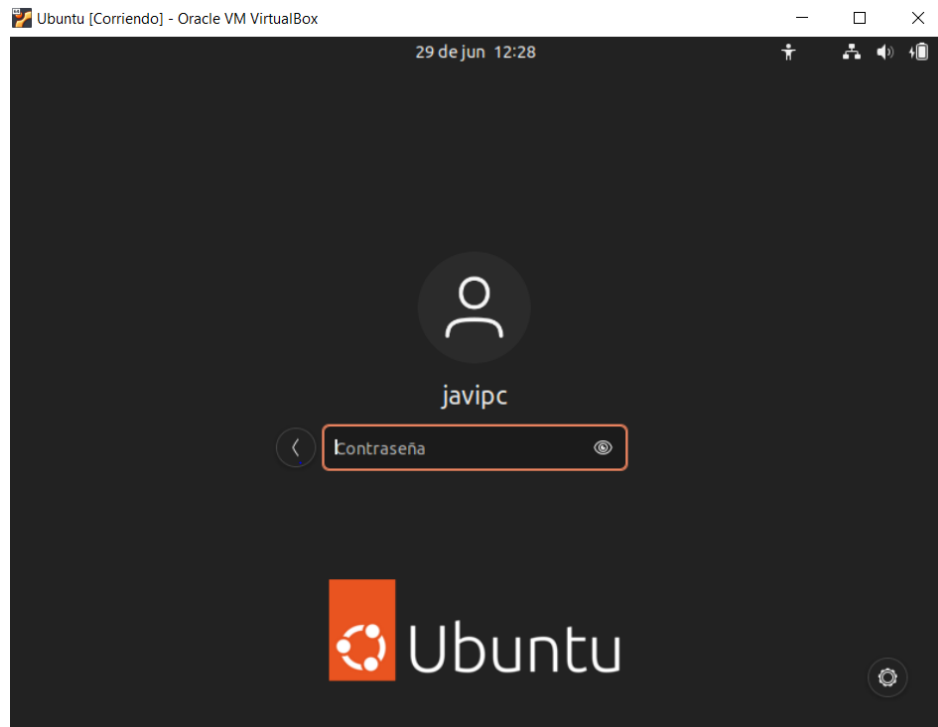


Figura 1.8

En la figura 1.8 observamos el control de acceso de un SO como Ubuntu Desktop 22.04 LTS (versión estable). Con el control de acceso no solo nos estamos identificando ante el SO sino que además, estamos indicando nuestro perfil de acceso que tiene aparejado niveles de seguridad dentro del SO. Es decir, desde que accedemos al sistema tenemos limitados según qué acciones y responsabilidades.

Habitualmente, en los sistemas operativos, los usuarios están agrupados en un grupo como mínimo, pero pudiendo ser miembro de otros.

Los permisos sobre las carpetas pueden ser gestionados teniendo como referencia no solo los usuarios sino agruparlos por grupos y los miembros de estos heredarán los permisos. No solo se gestionan los permisos sino la denegación sobre recursos.

Podemos seguir los principios siguientes:

- Si un recurso es denegado a un grupo o usuario prevalece sobre el resto de los derechos.
- Si a un recurso se le da derechos o privilegios para un grupo, todos los usuarios heredan esos privilegios.
- Si a un usuario se le dan privilegios sobre un recurso, pero no al grupo, solo el usuario tendrá los privilegios.
- Herencia de privilegios. Cuando un usuario tiene privilegios sobre una carpeta, el usuario heredará los mismos privilegios sobre las carpetas hijas, Siempre y cuando no se le deniegue.

1.2.1. Tipos de accesos

Dependiendo del tipo de áreas en el que se trabaje, hay 3 tipos de control de acceso:

- **Control de acceso obligatorio:** Se trata de una herramienta de control de acceso multinivel. Se define para ello una jerarquía de niveles de seguridad. Es decir, una política de roles que garantice la seguridad en el acceso de la información.
- **Control de acceso discrecional:** Aquí todos los objetos tienen un propietario. Es el propietario el que permite otorgar el acceso a los distintos recursos (información) a grupos y/o usuarios.
- **Control de acceso basado en roles:** Hay entornos en los que es difícil detectar quién es el propietario de los recursos. En los sistemas basados en roles, los usuarios tienen asignados roles basados en sus funciones en el sistema. Estos sistemas están centralmente administrados. Es decir, no tienen acceso discrecional.

Sistemas Windows

Estos sistemas contienen componentes de control y seguridad como los siguientes:

- Monitor de referencia de seguridad que hace cumplir las políticas de seguridad en el equipo local. Protege los recursos del sistema, audita y protege en tiempo de ejecución los distintos objetos.
- El administrador de recursos borra, crea y administra objetos ejecutivos y tipos de datos abstractos que son utilizados para representar recursos del SO como procesos, hilos y varios objetos de sincronización.
- El proceso del servidor de autenticación de seguridad local en realidad realiza peticiones de autenticación.

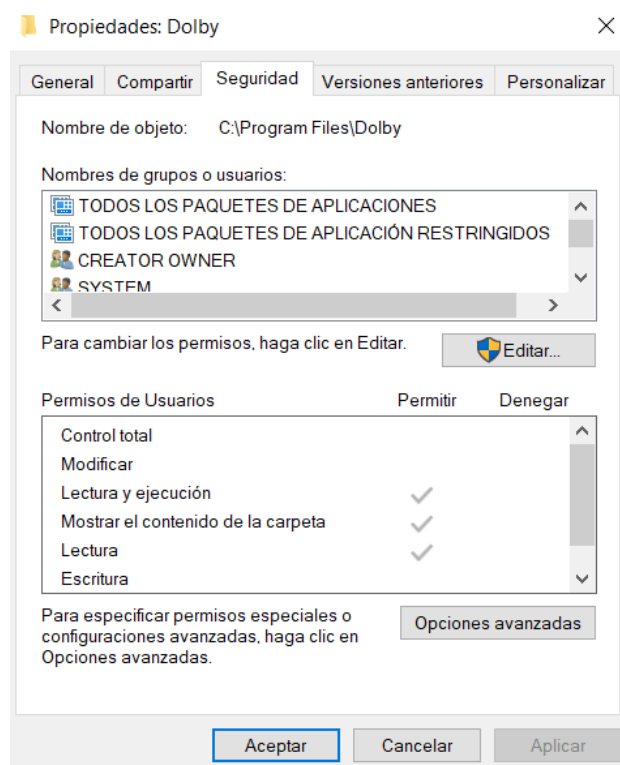


Figura 1.9.

Los sistemas Windows definen 13 permisos para ficheros y directorios o carpetas. Y proporciona incluso más opciones pudiendo decidir si el permiso será “acceso permitido” o “acceso denegado”:

- Recorrer carpeta/Ejecutable.
- Lectura de atributos.
- Lectura extendida de atributos.
- Crear archivos/Escribir datos.
- Crear carpetas/Añadir datos.
- Escribir atributos
- Escribir atributos extendidos.
- Eliminar subcarpetas y archivos.
- Borrado.
- Lectura de permisos.
- Cambio de permisos.
- Tomar posesión.

Y también se predefinen permisos básicos que son más intuitivos y debería ser suficiente para la mayoría de las tareas comunes.

- Mostrar contenido de la carpeta
- Control total
- Modificar
- Lectura
- Escritura
- Leer y ejecutar

En la figura 1.9 observamos usuarios y permisos para el usuario o grupo sobre una carpeta. Estos elementos de permisos son los últimos comentados. Sin embargo, si presionamos en “Opciones avanzadas” podemos desglosar de manera más especificada el nivel de permisos que se quiere otorgar.

Sistemas Linux

Normalmente los SOs Unix y Linux implementan una protección de control de acceso para los archivos que es básica a tres niveles. Los permisos se pueden garantizar a 3 niveles:

- **Usuario (u):** Especifica al usuario, probablemente el dueño del archivo o directorio.
- **Grupo (g):** Especifica los usuarios que pertenecen al mismo grupo como el indicado en el archivo o directorio.
- **Otros (o):** Representa al resto de usuarios.

Para cada clase podemos seleccionar 3 tipos de permisos: Lectura, Escritura y Ejecución.

Estos bits de permisos se comprueban en el siguiente orden:

- Si el UID (User Identifier) del archivo es el mismo que el UID del proceso, solo se aplican permisos de propietario; el grupo y otros permisos no se comprueban.
- Si los UID no coinciden, pero el GID (Group Identifier) del archivo coincide con uno de los GID del proceso, se aplican solo a los permisos de grupo; el propietario y otros permisos no se comprueban.

- Solo si el UID Y GID del proceso no coinciden con los del archivo solo se comprueban los permisos para “otros”. Si estos permisos no permiten la operación solicitada, se producirá un error.

```
javipc@javipc-VirtualBox:~$ ls -l
total 36
drwxr-xr-x 2 javipc javipc 4096 jun 29 10:30 Descargas
drwxr-xr-x 2 javipc javipc 4096 jun 29 10:30 Documentos
drwxr-xr-x 2 javipc javipc 4096 jun 29 10:30 Escritorio
drwxr-xr-x 2 javipc javipc 4096 jun 29 10:30 Imágenes
drwxr-xr-x 2 javipc javipc 4096 jun 29 10:30 Música
drwxr-xr-x 2 javipc javipc 4096 jun 29 10:30 Plantillas
drwxr-xr-x 2 javipc javipc 4096 jun 29 10:30 Público
drwx----- 3 javipc javipc 4096 jun 29 10:30 snap
drwxr-xr-x 2 javipc javipc 4096 jun 29 10:30 Videos
```

Figura 1.10.

Unix/Linux define tres atributos adicionales:

- **SUID (Set User ID):** Después de ejecutar un archivo, el proceso creado por lo general tiene su ID de usuario igual a la ID del usuario que ejecuta el programa. Sin embargo, si el bit SUID se encuentra en un ejecutable, el proceso creado obtiene el ID del propietario del archivo.
 - Activación
 - **chmod u+s archivo**
 - Desactivación
 - **chmod u-s archivo**
- **SGID (Set Group ID):** Un proceso que normalmente tiene el ID de grupo igual al grupo de procesos. Sin embargo, si el bit SGID se encuentra en el ejecutable, se le asigna el proceso cuyo ID de grupo es el correspondiente al grupo de archivos
 - Activación
 - **chmod g+s archivo**
 - Desactivación
 - **chmod g-s archivo**
- **Sticky Bit (bit pegadizo).**

En la **figura 1.10.** observamos, en modo terminal, los derechos y privilegios sobre dos archivos. Se observa el nombre del propietario y el grupo (el de la izquierda es el propietario y el siguiente el grupo). A su izquierda están las propiedades de los archivos o directorios y su representación es la siguiente:

- Si el primer bit es **d** es que se trata de un directorio o carpeta.
- Los siguientes nueve caracteres se dividen en grupos de 3:
 - Los primeros 3 corresponden a los atributos del fichero o directorio con respecto al propietario indicado.
 - Los siguientes corresponden a los atributos del fichero o directorio con respecto al grupo indicado.
 - Y los últimos 3 corresponden al resto de usuarios.

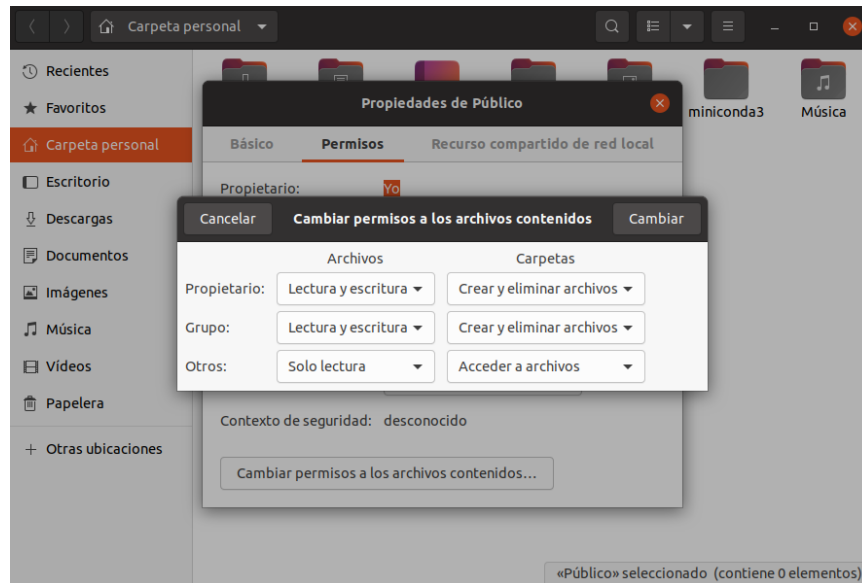


Figura 1.11.

En la **figura 1.11.** observamos una ventana (también llamada cuadro de diálogo) del SO Ubuntu Desktop. Bajo el escritorio Unity.

Se observan los 3 elementos diferenciadores: propietario, grupo y otros. Con dos columnas: archivos y carpetas. También podemos observar los permisos concretos que se establecen en cada caso para el directorio seleccionado.

1.2.2. Elección del tipo de acceso

Tomaremos como referencia los SSOO Linux y Windows por ser los más empleados. Hemos de conocer que Mac OS X es un SO de tipo Unix. Debido a que Mac es compatible con POSIX, la gran mayoría de paquetes escritos para BSD o Linux pueden ser recompilados para ser ejecutados en Mac.

Pero, ¿cuál es el tipo de acceso que debemos elegir?

Casi todos los SSOO, en modo local, permiten el uso del ordenador a varios usuarios. Seguramente, estos SSOO no necesiten de un servidor para comunicarse entre sí. Es el caso de Mac OS X, Windows y Linux.

¿Y qué implica? Que en una red local pueden comunicarse los ordenadores entre sí en una arquitectura de igual a igual. Esto hace pensar que un usuario de nuestro equipo puede acceder desde otro ordenador o dispositivo mediante unas credenciales si es que está libre el acceso.

Estos sistemas generan, a cada usuario, una zona “privativa” que podemos compartir toda o parte, de lectura y/o escritura con otros usuarios (o a todos). Debemos recordar que el usuario administrador o el usuario que tenga perfil de administrador podrán recorrer cualquier parte de nuestro equipo.

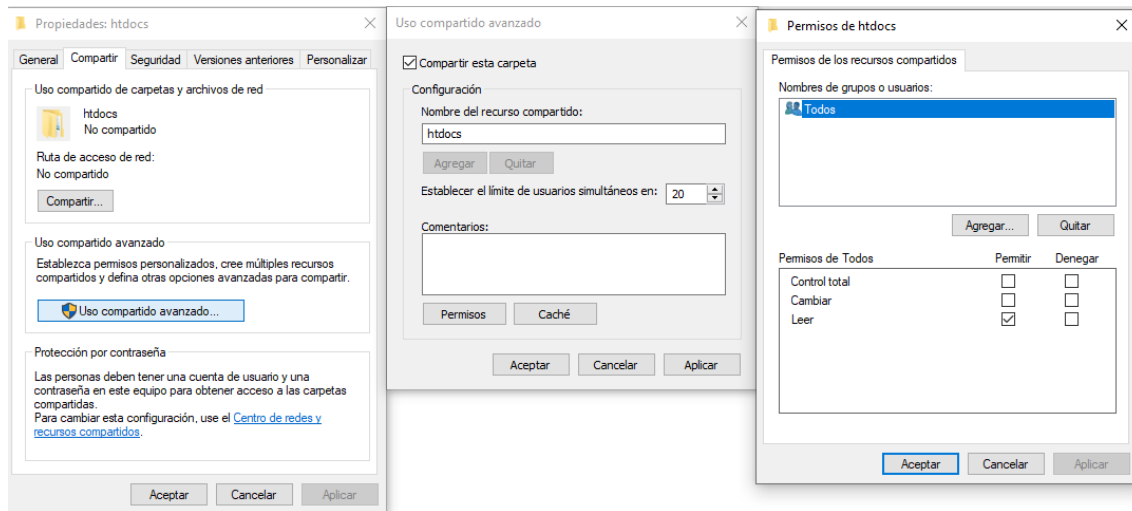


Figura 1.12.

Poniendo como ejemplo la imagen de la **figura 1.12**. Intentamos compartir una carpeta o directorio. ¿Cómo? Seleccionamos la carpeta que queremos compartir y, con el menú contextual, seleccionamos propiedades y, ahí, seleccionaremos la pestaña compartir y podremos configurar la compartición. No olvidemos que estamos configurando el acceso y debemos configurar la pestaña “Seguridad” para no crear conflictos de permisos.

Una forma más rápida es, seleccionando la carpeta objeto de compartición, con el menú contextual, seleccionar la opción “compartir con” donde podremos seleccionar el o los usuarios con aquellos que queremos compartir la carpeta o directorio.

Desde luego, el mejor sistema de compartición de recursos es a través de un servidor. Todos los aspectos de compartición y gestión de recursos quedan en manos de un SO (Servidor) cuyo fin es ese mismo.

1.2.3. Implementación de accesos

Se ha mencionado en el anterior punto que debemos adecuar los permisos de la pestaña “seguridad” con los permisos de compartición.

También se ha comentado que los usuarios administrador y aquellos que tengan perfil del grupo administradores no tendrán ningún problema en el mismo momento de acceder a unos recursos del sistema. Pero no ocurre lo mismo con aquellos usuarios que tienen perfil de “usuario estándar”. Es por ello que debemos “afinar” los permisos tanto para los grupos, si están agrupados, como para los usuarios si queremos que puedan realizar las tareas ya definidas sobre el recurso.

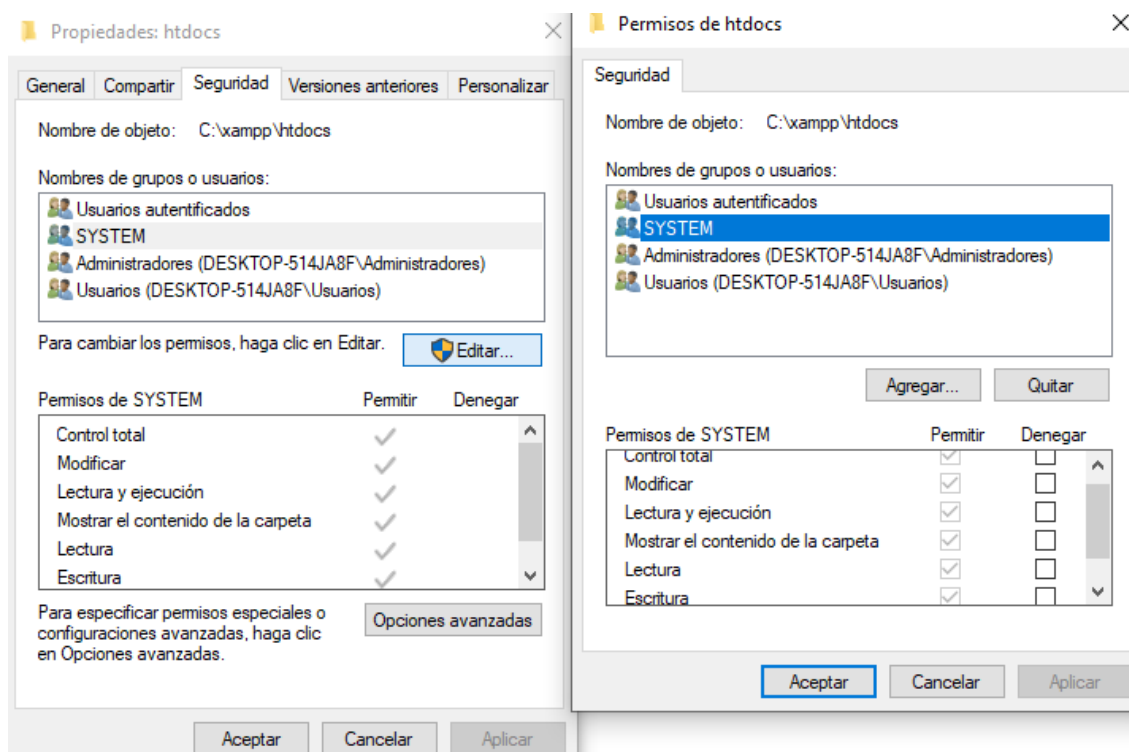


Figura 1.13.

En la **figura 1.13.** observamos, en la parte izquierda de la imagen, cómo están los permisos para el grupo SYSTEM. Si deseamos conocer los permisos de otro componente, basta con seleccionarlo para que aparezcan los permisos actuales que tienen asignados. Si se quieren modificar los permisos o añadir un usuario o grupo pulsaremos el botón “Opciones avanzadas”. Nos llevará a una ventana (cuadro de diálogo) que está a la derecha de la imagen. En esta ventana podremos agregar nuevos permisos y modificar los actuales que tiene asignado el grupo SYSTEM.

1.3. Órdenes de creación, modificación y borrado.

En el caso de querer realizar un proyecto será una buena idea crear una carpeta donde se guarde, organizadamente, cada uno de los archivos que tienen referencia al proyecto. Si hablamos de un proyecto web este constatará de archivos HTML, CSS, JavaScript, imágenes y documentación relacionada con el proyecto.

Cuando el autor indica “organizadamente” se refiere a separar los distintos tipos de archivos en carpetas. Es decir, dentro de la raíz de la carpeta “proyecto” crear una carpeta que se llame “CSS”, otra “imágenes”, etc. Solo se menciona como orientación.

1.3.1. Descripción de órdenes en distintos sistemas

Los sistemas operativos modernos tienen desarrollado una interfaz gráfica intuitiva que nos permite realizar todos los procesos de creación, modificación y borrado tanto de archivos como de carpetas. Esto no significa que el uso de órdenes a través de la consola quede apartado, aunque las personas que utilizan el ordenador a “nivel de usuario” crean que no son necesarias.

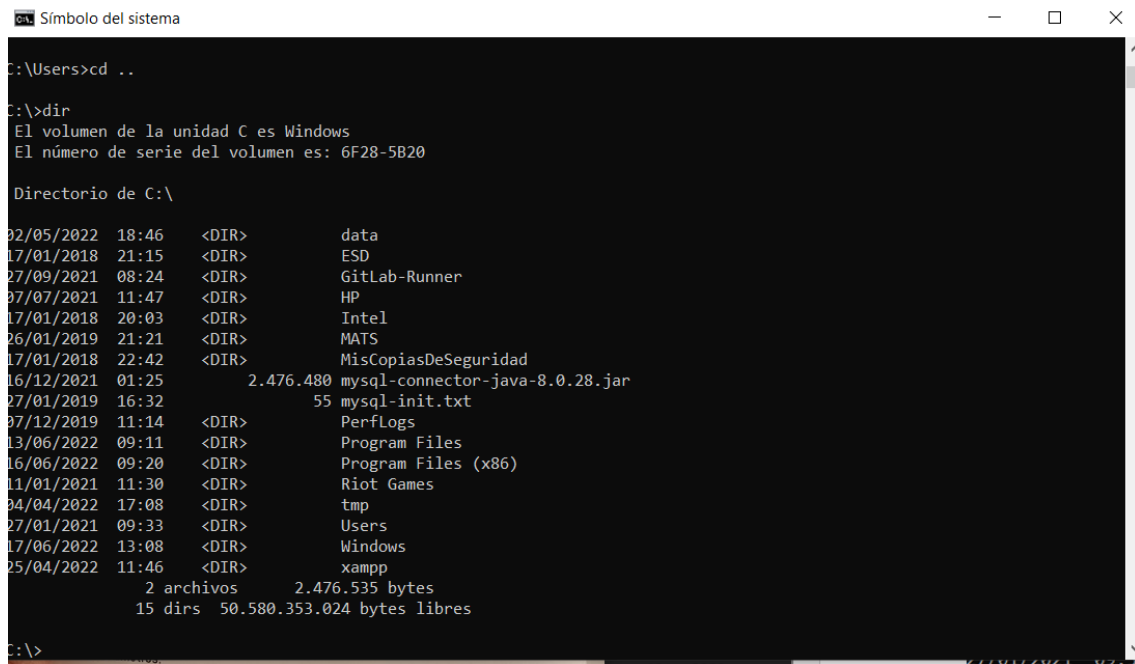
¿Cómo podemos ejecutar os comandos en modo consola? Pues dependerá del sistema operativo con el que estamos trabajando. No serán las mismas órdenes en Windows que en

GNU/Linux. El sistema operativo Mac OS X utiliza muchas ordenes y/o comandos igual que Linux y con las mismas funcionalidades.

Repasaremos dos sistemas operativos muy utilizados: Windows y GNU/Linux.

WINDOWS

Para obtener la entrada en consola o también denominada terminal debemos escribir **cmd** en la opción inicio -> buscar. Conseguiremos que nos aparezca una “pantalla negra” similar a la representada en la figura 1.14. Normalmente nos moveremos en la zona asignada al usuario (tal como aparece en la imagen mencionada). También observamos cómo aparecen carpetas o directorios y archivos indicando la orden dir.



```

C:\Users>cd ..

C:\>dir
El volumen de la unidad C es Windows
El número de serie del volumen es: 6F28-5B20

Directorio de C:\

02/05/2022  18:46    <DIR>          data
17/01/2018  21:15    <DIR>          ESD
27/09/2021  08:24    <DIR>          GitLab-Runner
07/07/2021  11:47    <DIR>          HP
17/01/2018  20:03    <DIR>          Intel
26/01/2019  21:21    <DIR>          MATS
17/01/2018  22:42    <DIR>          MisCopiasDeSeguridad
16/12/2021  01:25    2.476.480 mysql-connector-java-8.0.28.jar
27/01/2019  16:32    55 mysql-init.txt
07/12/2019  11:14    <DIR>          PerfLogs
13/06/2022  09:11    <DIR>          Program Files
16/06/2022  09:20    <DIR>          Program Files (x86)
11/01/2021  11:30    <DIR>          Riot Games
04/04/2022  17:08    <DIR>          tmp
27/01/2021  09:33    <DIR>          Users
17/06/2022  13:08    <DIR>          Windows
25/04/2022  11:46    <DIR>          xampp
                2 archivos      2.476.535 bytes
                15 dirs  50.580.353.024 bytes libres

C:\>

```

Figura 1.14.

Una vez que tenemos la consola abierta podremos utilizar los comandos y movernos por las carpetas utilizando, para ello, las órdenes o comandos que nos ofrece el sistema. Las órdenes o comandos que se utilizan son aquellos que se crearon con el sistema operativo MS-DOS, sistema operativo de Microsoft anterior a la serie Windows.

De siguiente manera, se mostrará con algún ejemplo, las órdenes que se utilizan para la creación, modificación y borrado de archivos y directorios.

- Orden **dir**: Con esta orden obtenemos información de los archivos y directorios que están en la ubicación indicada como parámetro.
 - **Sintaxis:**
 - dir [unidad\directorio\fichero]
 - **Parámetros:**
 - Algunos de los parámetros que se pueden emplear para actualizar el listado de directorios y archivos son los siguientes:
 - **/P ->** Muestra por pantalla el listado, para visualizar la pantalla siguiente basta con pulsar una tecla. Al pulsar la tecla se procesará el siguiente bloque de listado y así sucesivamente.

- /O
 - /ON
 - /OE
 - /OG
 - /OS
 - /OD
 - /O-X
- /S
- Orden **cd**: Permite cambiar de directorio al que se le especifique en la ruta.
 - **Sintaxis**
 - cd [unidad:]\[ruta]\[directorio]
 - **Observación**
 - Si deseamos bajar un nivel en el árbol de directorios, solo es necesario escribir **cd**.
 - Nota: Se deben tener en cuenta los valores de rutas relativas al directorio activo o rutas absolutas que serán independientes del directorio activo.
- Orden **md** o **mkdir**: crea un directorio o carpeta en la ruta que se especifique.
 - **Sintaxis**:
 - md [unidad\ruta\] <nombre>
 - En algunos casos se puede hacer mkdir dir1\dir2 y sería equivalente a las siguientes acciones: mkdir dir1; cd dir1; mkdir dir2; cd dir2;
- Orden **rd**: Borra un directorio (solo si está vacío).
 - **Sintaxis**
 - rd [unidad\ruta\]<nombre>
 - **Parámetros**
 - Los parámetros que se pueden utilizar con este comando son:
 - /S -> Elimina todo el directorio a borrar, aunque no esté vacío, pero pide confirmación.
 - /O-> No pide confirmación para eliminar un árbol de directorios cuando se utiliza junto con la opción.
- Nota: Disponemos de otros comandos para gestionar archivos como, por ejemplo:
 - **rename**:
 - Renombra un archivo
 - **del**:
 - Elimina un archivo
 - **move**:
 - Mueve un archivo

Si deseamos conocer más información de un comando podemos utilizar el parámetro /?. Por ejemplo: **dir /?** Nos ofrece información de ayuda del comando dir.

GNU/LINUX

En el caso de GNU/Linux utilizaremos la orden "terminal" para acceder a la consola. Como el sistema operativo Linux es multiterminal podemos pasar del entorno gráfico a modo terminal pulsando simultáneamente CTRL-ALT-F2 y obtendremos acceso al terminal 2 (tty2). Para pasar al termina donde "corre" el entorno gráfico utilizaremos CTRL-ALT-F7.

```
javipc@javipc-VirtualBox:~$ ls -all
total 68
drwxr-x--- 14 javipc javipc 4096 jun 23 14:07 .
drwxr-xr-x  3 root   root   4096 jun 23 13:23 ..
-rw-r--r--  1 javipc javipc  220 jun 23 13:23 .bash_logout
-rw-r--r--  1 javipc javipc 3771 jun 23 13:23 .bashrc
drwx----- 10 javipc javipc 4096 jun 27 11:29 .cache
drwx----- 11 javipc javipc 4096 jun 27 10:44 .config
drwxr-xr-x  2 javipc javipc 4096 jun 23 14:07 Descargas
drwxr-xr-x  2 javipc javipc 4096 jun 23 14:07 Documentos
drwxr-xr-x  2 javipc javipc 4096 jun 23 14:07 Escritorio
drwxr-xr-x  2 javipc javipc 4096 jun 23 14:07 Imágenes
drwx-----  3 javipc javipc 4096 jun 23 14:07 .local
drwxr-xr-x  2 javipc javipc 4096 jun 23 14:07 Música
drwxr-xr-x  2 javipc javipc 4096 jun 23 14:07 Plantillas
-rw-r--r--  1 javipc javipc  807 jun 23 13:23 .profile
drwxr-xr-x  2 javipc javipc 4096 jun 23 14:07 Público
drwx-----  3 javipc javipc 4096 jun 23 14:07 snap
drwxr-xr-x  2 javipc javipc 4096 jun 23 14:07 Videos
```

Figura 1.15.

En la imagen 1.15 observamos la ventana de “terminal” en una distribución Ubuntu Desktop.

A continuación, mostramos las órdenes utilizadas para gestionar archivos a través de la terminal.

INSTRUCCIÓN	¿QUÉ HACE?
cd [directorio]	Cambia el directorio por el especificado como parámetro.
mkdir directorio	Crea un nuevo directorio.
rmdir directorio	Borra un directorio vacío
mv fichero [fichero2... ficheroN] destino	Mueve o renombra ficheros o directorios
rm fichero1 [fichero2...ficheroN] destino	Borra ficheros y directorios con el parámetro -R (recursivo)
cp fichero1 [fichero2... ficheroN] destino	Copia ficheros y directorios en el directorio indicado.
pwd	Muestra en pantalla la ruta completa del directorio actual o activo.

Estas órdenes tienen la misma funcionalidad en sistemas operativos Mac OS X.

1.3.2. Implementación y comprobación de las distintas órdenes

Una manera de comprobar si las acciones realizadas que utilizan los comandos es mediante el explorador de archivos. A través del explorador de archivos, independientemente del SO con el que trabajemos, nos facilita explorar de forma más amplia y global si cabe, la estructura de los directorios o carpetas y los archivos que lo contienen.

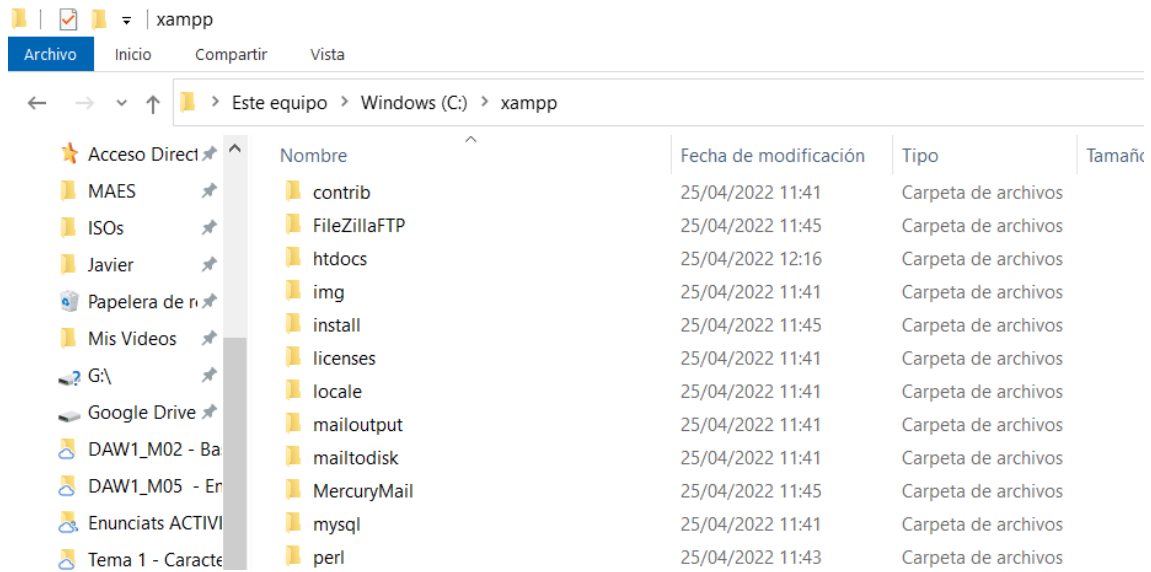


Figura 1.16.

En la figura 1.16 observamos la navegación gráfica. Esto marca una diferencia en cuanto al uso de comandos a través de la consola. Aunque la realización de tareas resulte más rápida con comandos de consola resulta siempre más cómodo de cara al usuario común acceder y manipular la información desde el explorador de archivos.

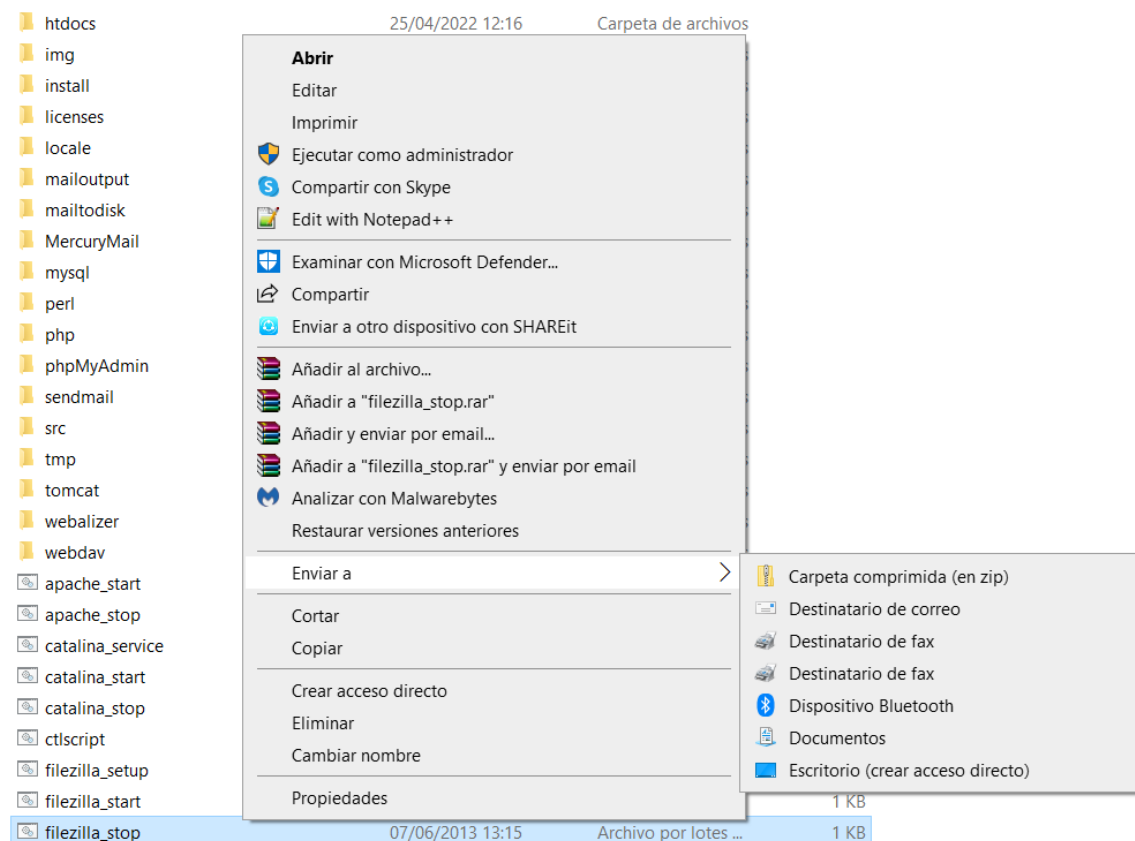


Figura 1.17.

Independientemente de las opciones que se encuentran en la barra de tareas, también disponemos de un menú contextual que nos permite realizar más operaciones sobre un archivo o directorio como podemos observar en la figura 1.17.

Todas estas herramientas software permiten realizar operaciones sobre el sistema de archivos.

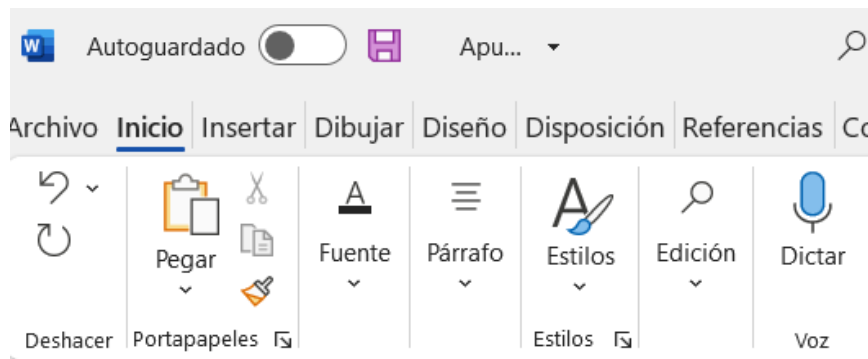


Figura 1.18

Como se ve en la figura 1.18, algunos paquetes de ofimática como Office o herramientas como DreamWeaver o Notepad++, disponen de opciones que permiten la creación y modificación de los documentos de la zona del sistema de archivos al que tengamos acceso y privilegios para hacerlo.

Volviendo a la herramienta gráfica del explorador de archivos. Con esta herramienta del SO se pueden llevar a cabo tareas de manera cómoda. Si ponemos un ejemplo, al copiar, mover un fichero de una carpeta a otra podremos utilizar el “arrastre” del documento. ¿Cómo se hace? Estamos situados sobre el área donde está el documento, abrimos otro explorador de documentos y nos situamos sobre la carpeta que queremos mover o copiar el documento, “pinchamos” con el botón izquierdo del ratón sobre el documento objeto y, sin soltar el botón del ratón, lo movemos al otro explorador y “soltamos” el botón.

También podemos utilizar siempre el sistema de atajos. El sistema de atajos es combinando teclas del teclado. Por ejemplo, si sobre el documento seleccionado presionamos CTRL-X el documento reduce la intensidad de su color y lo marca de manera que se está cortando. Para pasarlo a otra parte presionamos CTRL-V.

En caso de copiar utilizaremos CTRL-C + CTRL-V.

Bibliografía

Publicación de páginas Web MF0952_2 – Autor: José Talledo San Miguel – Certificado de Profesionalidad IFCD0110. Confección y publicación de páginas web. MF0952_2

