

UNIDAD 4: CONFIGURACIÓN DE SISTEMAS OPERATIVOS

Módulo Profesional: Sistemas Informáticos

Índice

RESUMEN INTRODUCTORIO	3
INTRODUCCIÓN	3
CASO INTRODUCTORIO	4
1. GESTIÓN DE USUARIOS	5
1.1 Gestión de usuarios en sistemas Windows	5
1.1.1 A través de la línea de comandos. NET USER.....	5
1.1.2 A través de herramientas gráficas. Cuentas de usuario	8
1.2. Gestión de usuarios en sistemas Linux	16
1.2.1 A través de herramientas gráficas. Cuentas de usuario	17
1.2.2. A través de comandos. USERADD y USERDEL.....	22
1.3 Seguridad en cuentas de usuario.....	24
1.3.1 Seguridad de contraseñas	25
2. GESTIÓN DE GRUPOS.....	27
2.1. Gestión de grupos en sistemas Windows	27
2.2 Gestión de grupos en sistemas Linux	29
3. ACCESO A RECURSOS. PERMISOS LOCALES.....	31
3.1. Permisos locales en Linux	31
3.2 Permisos locales en Windows	34
4. PROCESOS	36
4.1 Tipos de procesos.....	36
4.2 Administración de procesos en Linux.....	38
4.2.1 Mediante Interfaz Gráfica.....	38
4.2.2 Mediante comandos	40
5. SERVICIOS	44
5.1 Administración de servicios en Linux.....	44
5.1.1. Iniciando servicios manualmente, directorio init.d	44
5.1.2. El comando service	45
6. MONITORIZACIÓN Y MANTENIMIENTO DEL SISTEMA	46
6.1 Herramientas propias de los sistemas operativos.....	46
6.2 Herramientas complementarias.	49
RESUMEN FINAL.....	52

RESUMEN INTRODUCTORIO

A lo largo de esta unidad revisaremos con detalle los conceptos de usuario y grupo dentro de un sistema operativo. Veremos los tipos que existen, y el modo en que se realiza su gestión tanto en sistemas operativos libres como propietarios.

También hablaremos de la seguridad en el acceso a los diferentes recursos de nuestro sistema, presentando la noción de permiso y viendo el modo en que se aplica en un sistema operativo.

Estudiaremos a continuación los procesos y los servicios, analizando la importancia que tienen en el funcionamiento cotidiano de nuestros sistemas, y viendo el modo de gestionarlos de forma correcta.

Finalmente revisaremos una serie de herramientas que nos pueden ayudar en la monitorización de nuestros sistemas, algo fundamental a la hora de realizar su mantenimiento para asegurarnos de que funcionan de forma óptima.

INTRODUCCIÓN

En el tema anterior nos centramos en la gestión de la información dentro de nuestros sistemas informáticos, presentando herramientas fundamentales de cualquier sistema operativo como los sistemas de archivos. Pero, ¿basta con tener bien organizados nuestros datos para que el sistema funcione correctamente? Para dar respuesta a esta pregunta basta con pensar qué ocurriría si cualquier persona con acceso a nuestros dispositivos pudiera manipular toda la información que contienen sin ninguna restricción.

Obviamente se necesita algún tipo de control sobre los datos en cualquier sistema, y así aparecen los conceptos de usuario, grupo y permiso. Conceptos que cualquier desarrollador actual debe conocer en una doble vertiente:

- Como usuario de sistemas, ya que por regla general trabajará dentro de un grupo y tendrá permisos de acceso sólo a determinada información de la empresa.
- Como creador de nuevo software, en el que tendrá que tener en cuenta estos aspectos para dotarlo de seguridad.

Por otro lado, hay que tener en cuenta que muchas de las aplicaciones actuales son lo suficientemente complejas para estar compuestas de varios módulos que trabajan en conjunto. Para saber cómo estructurarlas es fundamental tener claros conceptos como los de proceso y servicio.

Y no menos importante es conocer técnicas que nos permitan monitorizar nuestros equipos de trabajo en busca de posibles errores que disminuyan su rendimiento.

CASO INTRODUCTORIO

En nuestro primer día de la FCT la empresa nos pide crear tres nuevas cuentas de usuario en un sistema operativo Windows, e incluirlos en un grupo de trabajo específico para ellos. Los datos de los usuarios se nos han facilitado, pero se nos indica que, por seguridad, las contraseñas tendrán que ser establecidas una vez el usuario acceda al sistema.

Al finalizar la unidad el alumnado identificará los distintos usuarios existentes en sistemas operativos libres y propietarios y será capaz de gestionar de forma correcta, tanto mediante comandos como utilizando herramientas gráficas, será capaz de organizar los usuarios mediante grupos, estará capacitado para proteger el acceso a los recursos de un sistema mediante la asignación de permisos, distinguirá los conceptos de proceso y servicio y será capaz de trabajar con ellos en distintos sistemas operativos, será capaz de utilizar herramientas para la monitorización de los sistemas y conocerá y aplicará utilidades de mantenimiento y optimización del sistema.

1. GESTIÓN DE USUARIOS

En unidades anteriores se han mostrado diferentes técnicas de trabajo partiendo de la base de que un sistema informático es utilizado por un solo usuario. Pero esto no es lo más habitual. Tanto en casa como en nuestro puesto de trabajo es muy común que un ordenador sea usado por diferentes trabajadores o miembros de una misma familia. Para que cada uno de ellos disponga de sus documentos, escritorio y otro tipo de recursos, es necesario que el administrador del sistema cree estos usuarios previamente para que puedan iniciar sesión de forma local (o mediante un dominio en un servidor).

A continuación se revisará cómo llevar a cabo la gestión de estos usuarios tanto en sistemas operativos libres como propietarios.

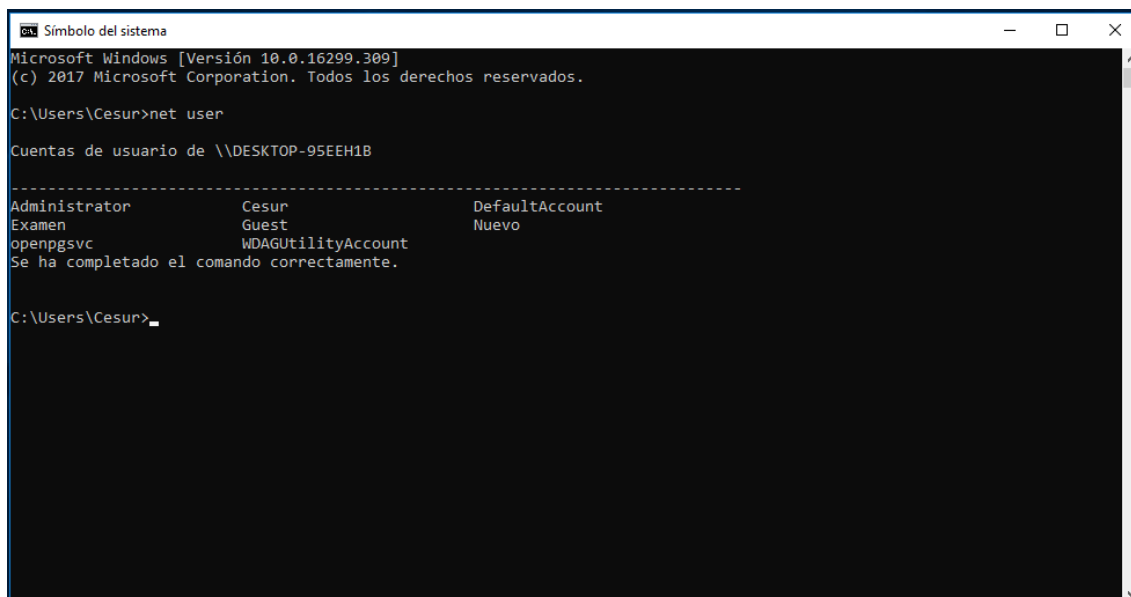
1.1 Gestión de usuarios en sistemas Windows

A continuación se muestra cómo realizar la gestión de usuarios a través de la línea de comandos NET USER y a través de herramientas gráficas y cuentas de usuario de Windows.

1.1.1 A través de la línea de comandos. NET USER

El comando NET USER se utiliza para crear/modificar/eliminar cuentas de usuario desde línea de comando en sistemas operativos Windows. Gracias a él, podemos realizar varias operaciones sobre los usuarios, entre las que destacamos:

- Para ver los usuarios del sistema se usará directamente el comando **NET USER**, sin ningún parámetro adicional.



```

Microsoft Windows [Versión 10.0.16299.309]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Cesur>net user

Cuentas de usuario de \\DESKTOP-95EEH1B

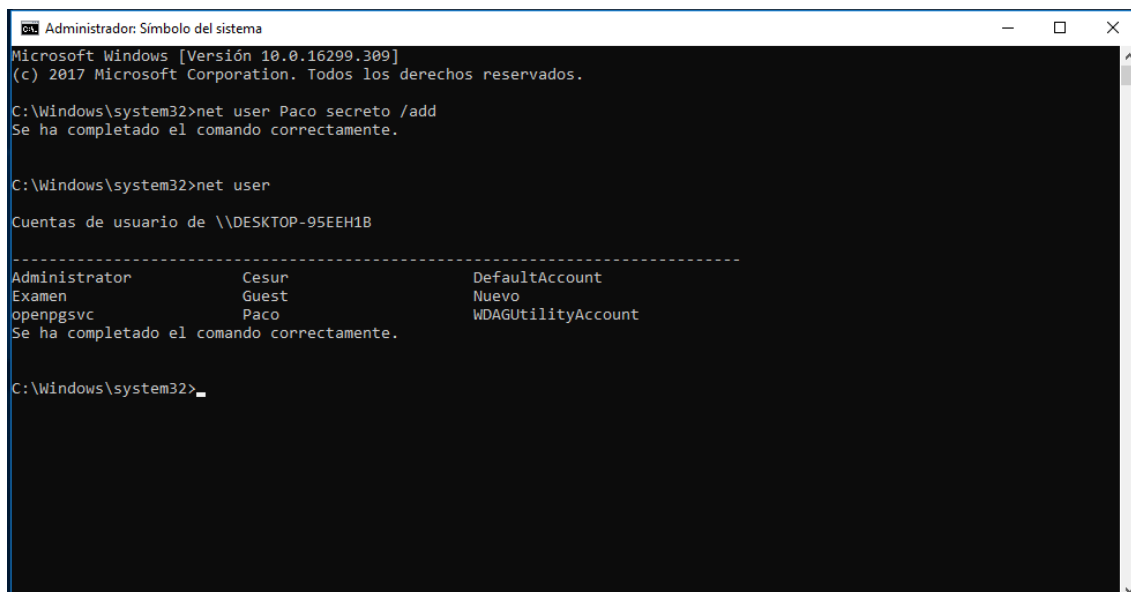
-----
Administrator      Cesur              DefaultAccount
Examen             Guest             Nuevo
openpgsvc          WDAGUtilityAccount
Se ha completado el comando correctamente.

C:\Users\Cesur>

```

Imagen: Listado de usuarios del sistema con NET USER

- Para agregar un nuevo usuario, asignándole directamente su contraseña, se emplea la sintaxis: **NET USER NombreUsuario Contraseña /add**



```

Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.16299.309]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\Windows\system32>net user Paco secreto /add
Se ha completado el comando correctamente.

C:\Windows\system32>net user

Cuentas de usuario de \\DESKTOP-95EEH1B

-----
Administrator      Cesur              DefaultAccount
Examen             Guest             Nuevo
openpgsvc          Paco              WDAGUtilityAccount
Se ha completado el comando correctamente.

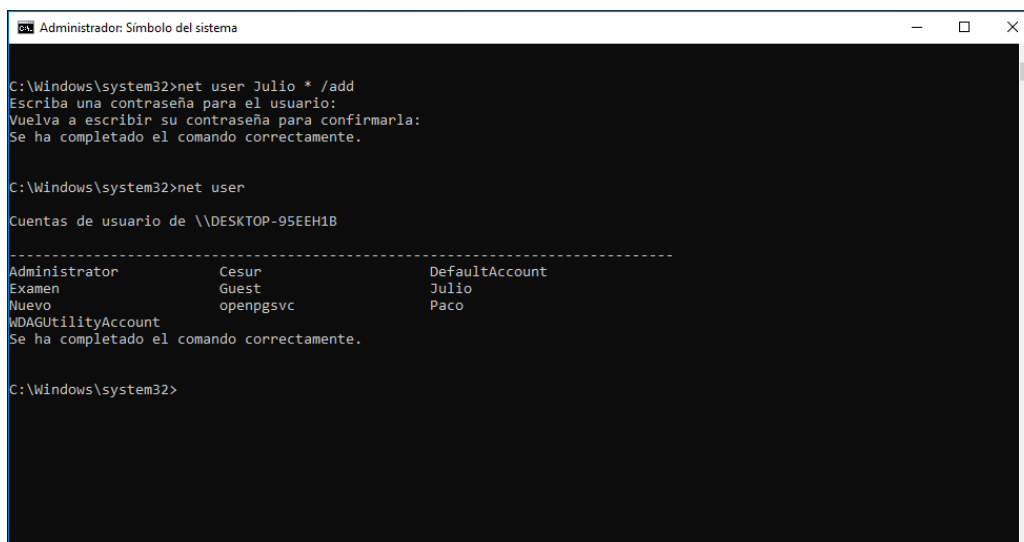
C:\Windows\system32>

```

Imagen: Creación de un nuevo usuario llamado Paco, con contraseña "secreto"

NOTA IMPORTANTE: Hay que reseñar que en este caso se necesitan permisos de Administrador.

- Una variante del caso anterior sería no introducir la contraseña directamente, sino hacerlo por separado. La sintaxis sería: **NET USER NombreUsuario */add**



```

C:\Windows\system32>net user Julio * /add
Escriba una contraseña para el usuario:
Vuelva a escribir su contraseña para confirmarla:
Se ha completado el comando correctamente.

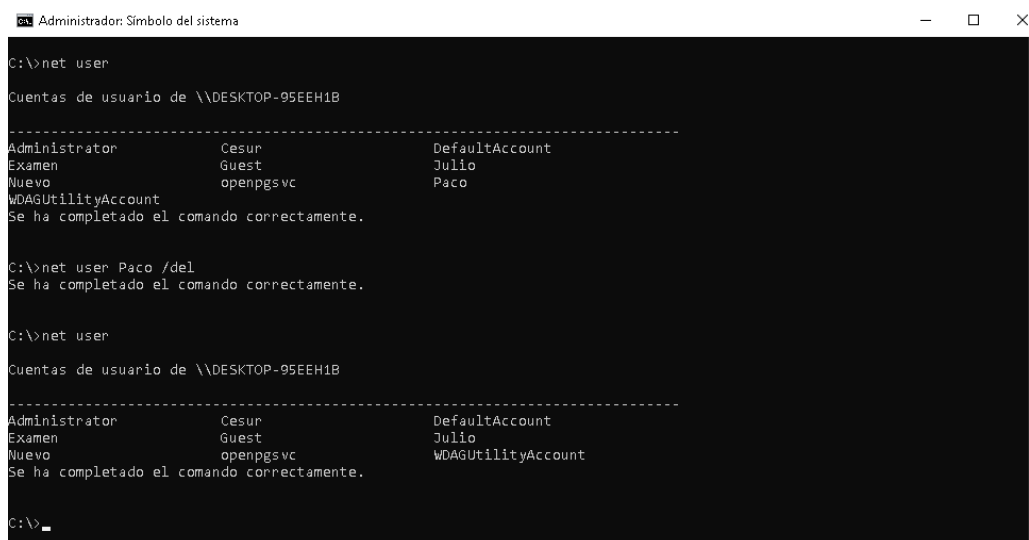
C:\Windows\system32>net user

Cuentas de usuario de \\DESKTOP-95EEH1B
-----
Administrador      Cesur      DefaultAccount
Examen            Guest      Julio
Nuevo             openpgsvc  Paco
WDAGUtilityAccount
Se ha completado el comando correctamente.

C:\Windows\system32>
  
```

Imagen: Creación de un nuevo usuario llamado Julio, con petición independiente de la contraseña

- Para eliminar un usuario previamente creado se emplea la sintaxis: **NET USER NombreUsuario /del**



```

C:\>net user

Cuentas de usuario de \\DESKTOP-95EEH1B
-----
Administrador      Cesur      DefaultAccount
Examen            Guest      Julio
Nuevo             openpgsvc  Paco
WDAGUtilityAccount
Se ha completado el comando correctamente.

C:\>net user Paco /del
Se ha completado el comando correctamente.

C:\>net user

Cuentas de usuario de \\DESKTOP-95EEH1B
-----
Administrador      Cesur      DefaultAccount
Examen            Guest      Julio
Nuevo             openpgsvc  WDAGUtilityAccount
Se ha completado el comando correctamente.

C:\>
  
```

Imagen: Eliminación del usuario Paco

1.1.2 A través de herramientas gráficas. Cuentas de usuario

Método básico.

La forma más sencilla de gestionar usuarios locales en sistemas Windows, disponible en todas sus versiones, es acceder al **Panel de control** → **Cuentas de usuario**.

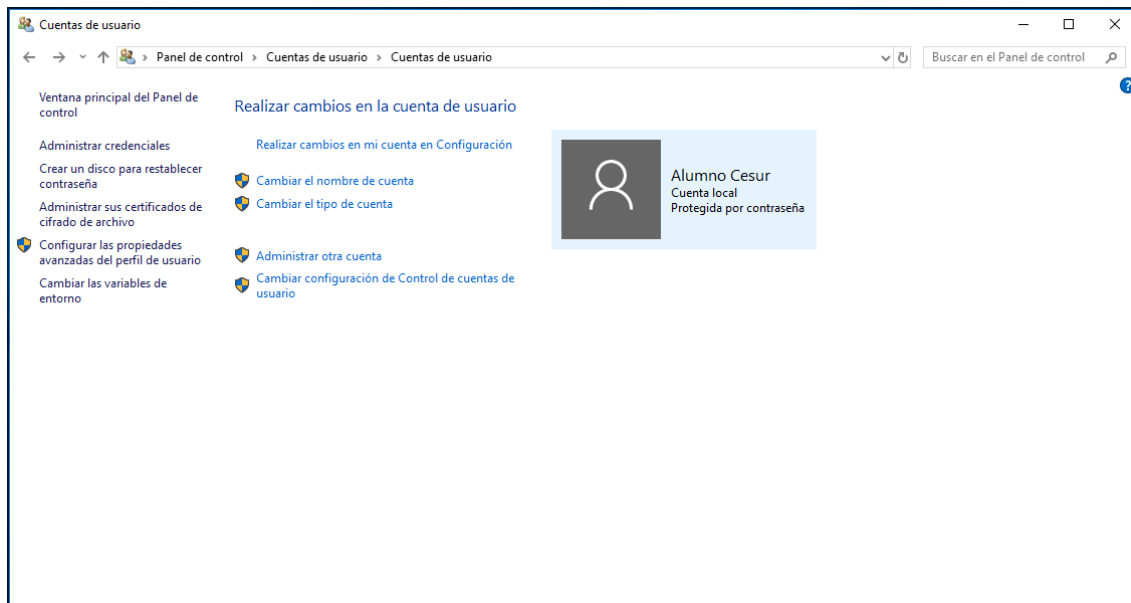


Imagen: Herramienta Cuentas de usuario en Windows 10

En esta herramienta, de entrada, se ofrecen opciones para cambiar el nombre de la cuenta y su tipo. Y, utilizando la opción “Realizar cambios en mi cuenta en Configuración”, cambiar la imagen por otra predefinida o añadida por nosotros. Todas estas opciones son relativas al usuario local que ha iniciado sesión.

Para **crear una cuenta nueva**, se pulsa sobre la opción “Administrar otra cuenta”. Se muestran las cuentas de todos los usuarios habilitados, así como una opción de **Agregar una cuenta de usuario** en la parte inferior, que habrá que pulsar para crear el nuevo usuario.



Imagen: Administrar otra cuenta

Mediante el siguiente formulario que se nos muestra se creará una cuenta nueva de usuario, introduciendo su nombre, una contraseña y un indicio para recordarla.

Agregar un usuario

Elige una contraseña que sea fácil de recordar para ti pero difícil de adivinar para otros. Si la olvidas, te mostraremos el indicio.

Windows no se puede conectar a Internet en estos momentos. Comprueba la conexión a Internet y vuelve a intentarlo más tarde si quieres agregar una cuenta Microsoft.

Nombre de usuario

Contraseña

Vuelve a escribir la contraseña

Indicio de contraseña

Siguiente

Cancelar

Imagen: Agregar nuevo usuario

Una vez creado el usuario, se podrá cambiar su nombre, la contraseña y el tipo de la cuenta.

Realizar cambios en la cuenta de Nuevo

[Cambiar el nombre de cuenta](#)

[Cambiar la contraseña](#)

[Cambiar el tipo de cuenta](#)

[Eliminar la cuenta](#)

[Administrar otra cuenta](#)



Imagen: Configuración del nuevo usuario

Lo habitual es elegir un tipo de cuenta estándar, ya que por tener permisos limitados lo que se haga en dicha cuenta no afecta a otros usuarios ni a la seguridad del equipo informático al que pertenezca.

En esta misma ventana, como se aprecia en la imagen anterior, tenemos una opción para **eliminar una cuenta** de usuario. Se puede acceder a ella en cualquier momento, si deseamos borrar un usuario y tenemos permisos para ello, a través de **Cuentas de usuario → Administrar otra cuenta**.

Método avanzado.

Una alternativa más sofisticada, disponible en la versión profesional de los sistemas operativos Windows, es emplear la herramienta **Administración de equipos**. Se puede acceder a ella desde el **Panel de control → Herramientas administrativas**.

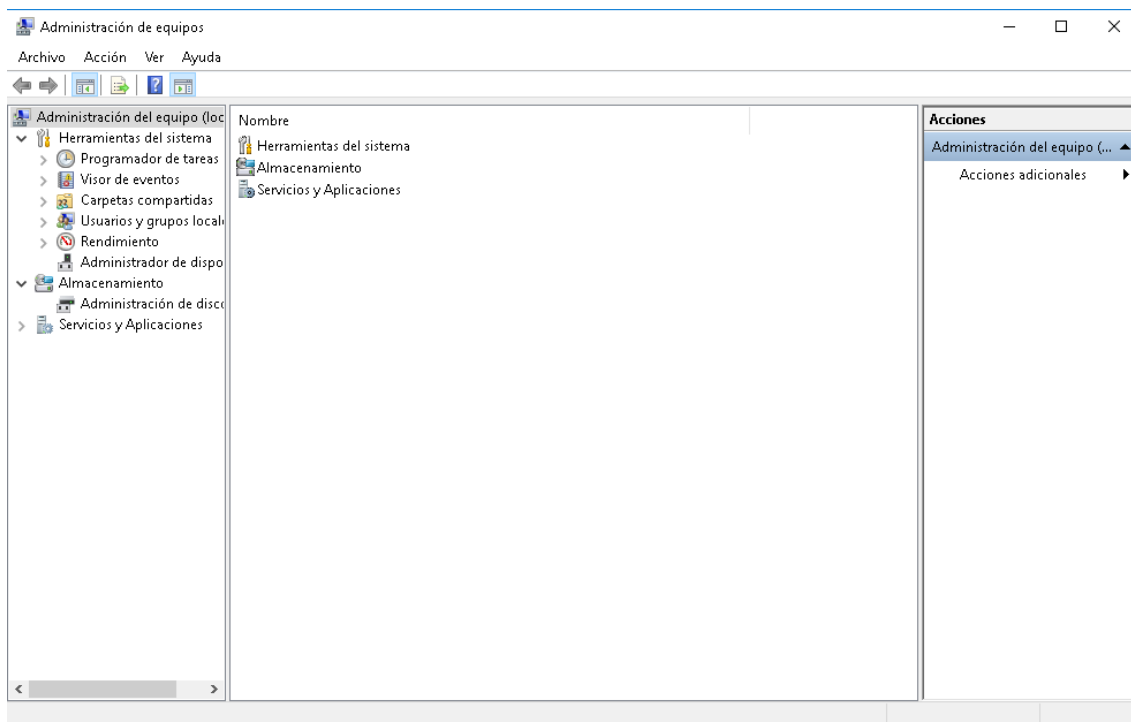


Imagen: Administración de equipos en Windows 10

En esta ventana, se debe acceder a la opción **Usuarios y grupos locales** del menú lateral izquierdo, y a continuación seleccionar **Usuarios**. En ese momento aparecerán en el panel central los usuarios locales que haya creados en el sistema operativo en ese momento.

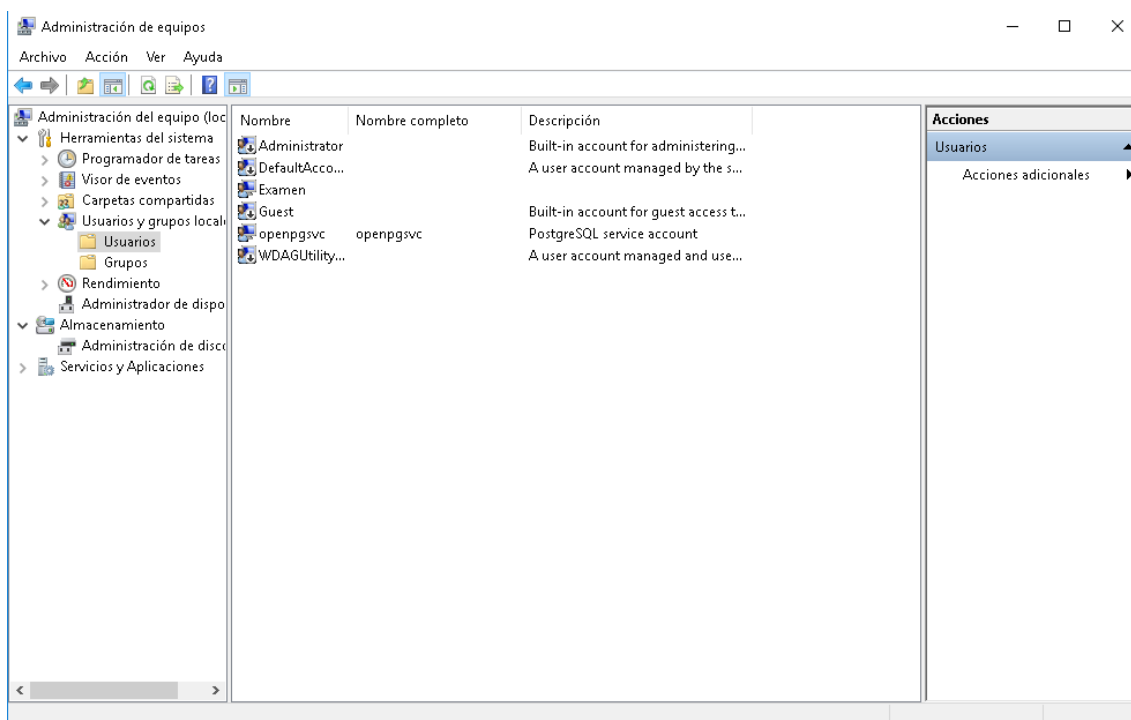


Imagen: Usuarios locales creados en el sistema

Cuando uno de estos usuarios muestra una flecha en su icono, se está indicando que su cuenta está deshabilitada.

Como se aprecia en la imagen anterior, por defecto existe un usuario Administrador (que tiene todos los privilegios), otro Invitado/Guest (que no tiene control total sobre el sistema operativo, por ejemplo no puede crear otras cuentas de usuario ni grupos) y un usuario inicial, en este caso llamado Examen, que es el usuario que se ha añadido durante el proceso de instalación del sistema operativo.

Para **crear un usuario** nuevo con esta herramienta, se pulsa con el botón derecho del ratón en un espacio libre del panel central y se selecciona **Usuario Nuevo**. Otra forma es acceder al menú acción y seleccionar la opción Usuario nuevo. Ambos procedimientos pueden emplearse igualmente para **eliminar** un usuario ya creado.

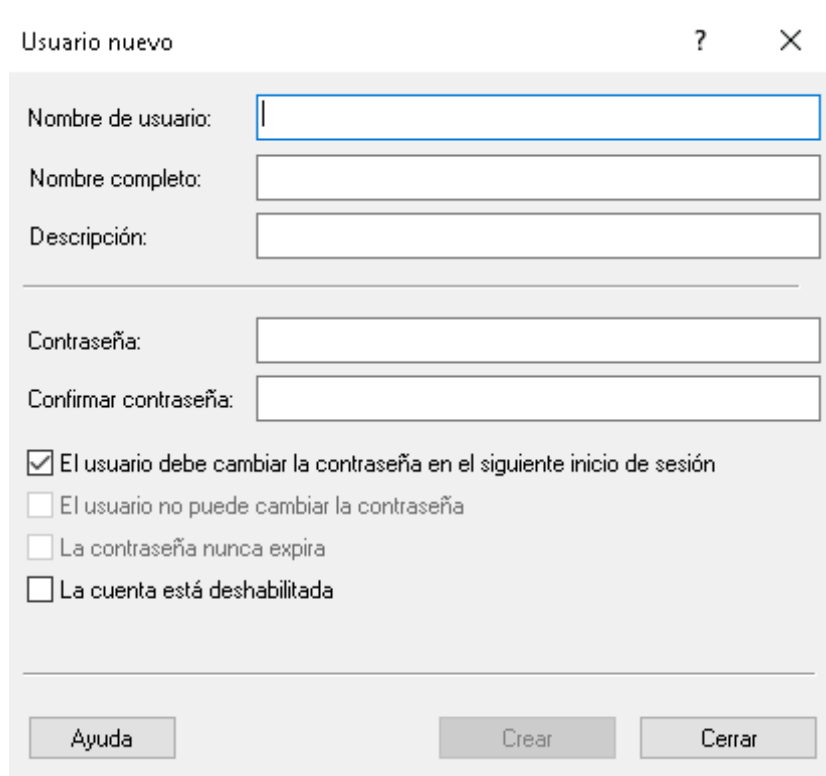


Imagen: Creación de un usuario nuevo

Para dar de alta al usuario habrá que rellenar los campos siguientes:

Nombre de usuario: Nombre que identificará al nuevo usuario (familiar, empleado, amigo...) en el sistema. En este campo hay que tener en cuenta varios aspectos como:

- No puede haber dos nombres de usuarios repetidos.
- No puede coincidir con el de un grupo.
- Longitud máxima de 20 caracteres.
- No puede contener espacios en blanco.
- No puede contener los siguientes caracteres: " / \ [] : ; | = , +*? <> @

Nombre completo: Se escribirá el nombre completo junto a los apellidos del usuario a dar de alta.

Descripción: campo en el cual se proporciona más información sobre el usuario (departamento, planta, negocio, etc.)

Contraseña: permite 14 caracteres como máximo, y se distingue entre mayúsculas y minúsculas.

Una serie de casillas de verificación, que nos permiten gestionar algunos aspectos relativos a la contraseña, como que:

- Se pueda cambiar la contraseña en el siguiente inicio de sesión.
- Cuenta deshabilitada.
- La contraseña nunca expira, el usuario no puede cambiar la contraseña.

Una vez introducidos todos los datos de forma correcta, se puede Crear el usuario nuevo.








Nombre	Nombre completo	Descripción
 Administrator		Built-in account for administering...
 Cesur	Alumno Cesur	Alumno de Cesur
 DefaultAcco...		A user account managed by the s...
 Examen		
 Guest		Built-in account for guest access t...
 openpgsvc	openpgsvc	PostgreSQL service account
 WDAGUtility...		A user account managed and use...

Imagen: Usuario Cesur creado

Cuando se reinicie el equipo aparecerán las cuentas activas en ese momento, no mostrándose las cuentas de Administrador ni Invitado, ya que como se ha visto se encuentran deshabilitadas.

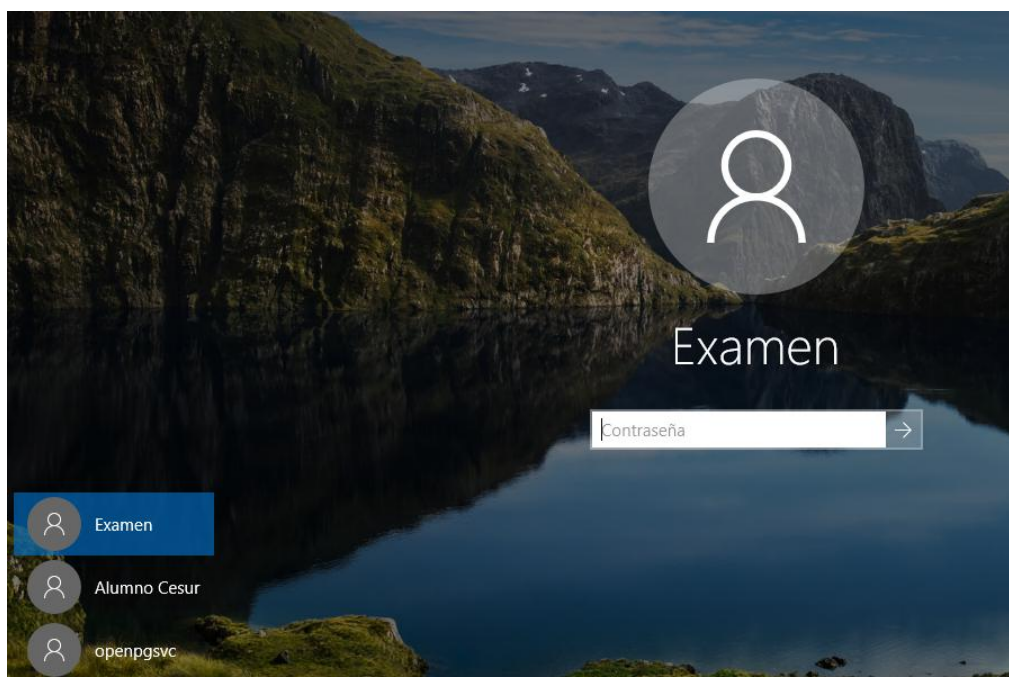


Imagen: Cuentas de usuario en el inicio del sistema

Al seleccionar el nuevo usuario Cesur para iniciar sesión, se informa de que, tal como se había indicado en el momento de crear la cuenta, la contraseña debe de ser cambiada, procediéndose en ese instante a crear una nueva contraseña.

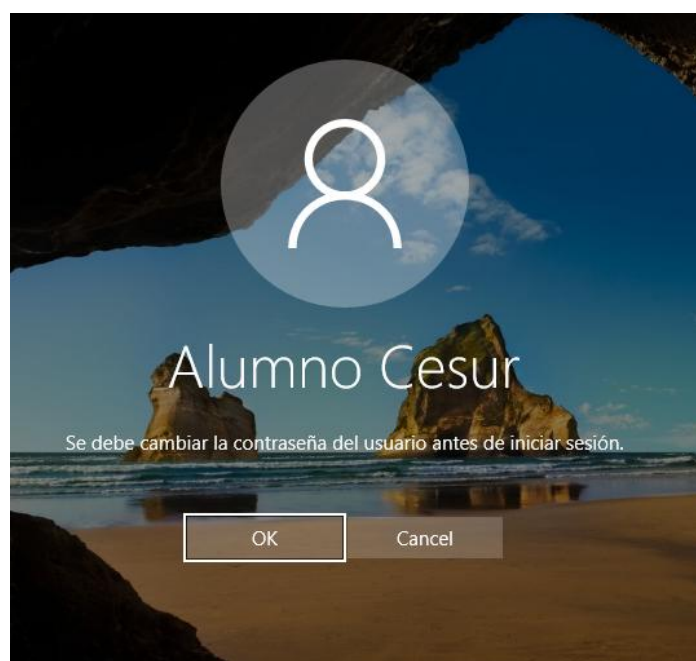


Imagen: Solicitud de cambio de contraseña

Como se puede apreciar en la siguiente imagen, primero hay que introducir la contraseña que se tenía asignada (en nuestro caso se dejó en blanco, ya que se iba a solicitar cambiarla en el primer acceso al sistema), y a continuación se debe facilitar la contraseña nueva.



Imagen: Introducción de contraseñas antigua y nueva

Introducida y confirmada la nueva contraseña, se realiza el cambio, con lo queda completado el proceso de creación del nuevo usuario.

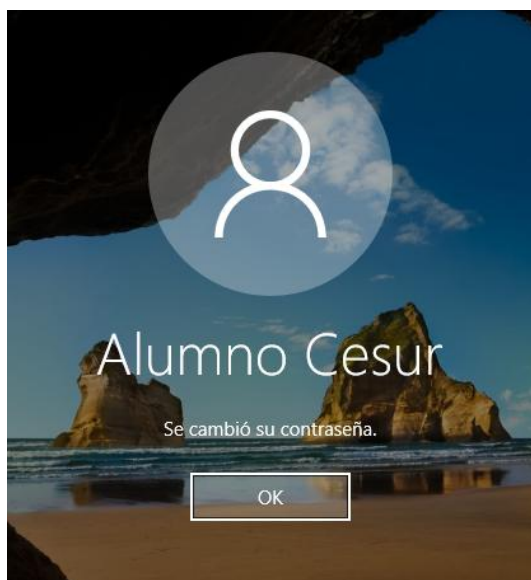


Imagen: Contraseña cambiada de manera correcta

1.2. Gestión de usuarios en sistemas Linux

Linux, como su predecesor Unix, es un sistema operativo multiusuario. Para que múltiples usuarios puedan hacer uso del sistema de una forma segura y ordenada, es necesario que dicho sistema disponga de mecanismos de administración y seguridad para proteger los datos de cada usuario, así como para proteger y asegurar el correcto funcionamiento del sistema en sí.

Para poder utilizar un sistema operativo Linux es necesario disponer de una cuenta de usuario que se compone de **nombre de usuario (login) y de contraseña (password)**. Las cuentas de usuario son creadas por el administrador, que en Linux es un usuario especial llamado **root**.

Al instalar una distribución de Linux, como se vio en la Unidad 2 en el caso de Ubuntu, se crea durante el proceso una cuenta de administrador llamada **root**, que tiene control total sobre el sistema operativo.

El **usuario root** dispone de las siguientes opciones:

- Instalar el software necesario en el sistema operativo.
- Gestionar los usuarios y contraseñas de los mismos.
- Mantenimiento del sistema.
- Control total sobre el sistema de archivos.

Además de root, existen dos tipos más de cuentas de usuario en Linux:

- **Las cuentas de los usuarios estándar o locales:** tienen limitaciones en cuanto a las acciones que pueden iniciar así como a los archivos y carpetas a los que pueden acceder, salvo en su directorio personal en este último caso. Es por ello que son los usuarios recomendados para el uso diario del sistema.
- **Las cuentas del sistema o cuentas de usuarios asociados a servicios:** no pueden iniciar sesiones en el sistema, pero por medio de ellas se pueden establecer los permisos asociados a dichos servicios. Ejemplos son las cuentas de los usuarios apache, bin...



ENLACE DE INTERÉS

Para administrar sistemas Linux es interesante conocer dos ficheros fundamentales para la gestión de usuarios y grupos:

<https://es.ccm.net/contents/317-linux-gestion-de-usuarios>

Por lo expresado anteriormente, y para evitar causar daños involuntarios a nuestro sistema, así como por seguridad, es conveniente que entremos y trabajemos como usuarios estándar, dejando la cuenta de root únicamente para temas de administración que sólo él pueda realizar.

En Linux, la cuenta de usuario root es la dueña de todo. Todo lo que hagamos con esa cuenta desde que iniciemos sesión con ella, repercutirá en todo el sistema, lo hagamos consciente o inconscientemente. **Ello conlleva el tener una responsabilidad que no es necesaria durante nuestro trabajo cotidiano**, para ello están las cuentas estándar o locales, las cuales podemos configurar absoluta y totalmente y sin que ello repercuta en el resto de nuestro Linux. Podemos personalizar nuestro escritorio, programas preferidos, etc, sin riesgos colaterales, ya que los cambios sólo afectarán a nuestro usuario. Las cuentas de usuario fueron creadas específicamente para ello, las de administrador para acciones puntuales.

Para la **administración de usuarios en Linux**, igual que ocurriría en los sistemas Windows, también se dispone de diferentes opciones, que se verán a continuación. Al ser usuarios locales los que se crearán, la gestión de los mismos afecta solamente al equipo local, sin afectar al resto de equipos de la red. Cada usuario creado dispone de una cuenta de acceso y un directorio propio de trabajo.

1.2.1 A través de herramientas gráficas. Cuentas de usuario

En Linux, la gestión de usuarios en modo gráfico es similar a la que ya se ha descrito en otros sistemas operativos.

Para realizar dicha gestión se accede a la Configuración del Sistema → Sistema → Cuentas de usuario, apareciendo la siguiente ventana, desde la que se puede **añadir un nuevo usuario** pulsando sobre el signo '+' situado en la esquina inferior izquierda



Imagen: Herramienta Cuentas de usuario en Ubuntu

Lógicamente, si como es recomendable se ha accedido con un usuario estándar, se necesitan privilegios de root para crear este nuevo usuario. Por ello hay que pulsar el botón **Desbloquear**, y autenticarse.

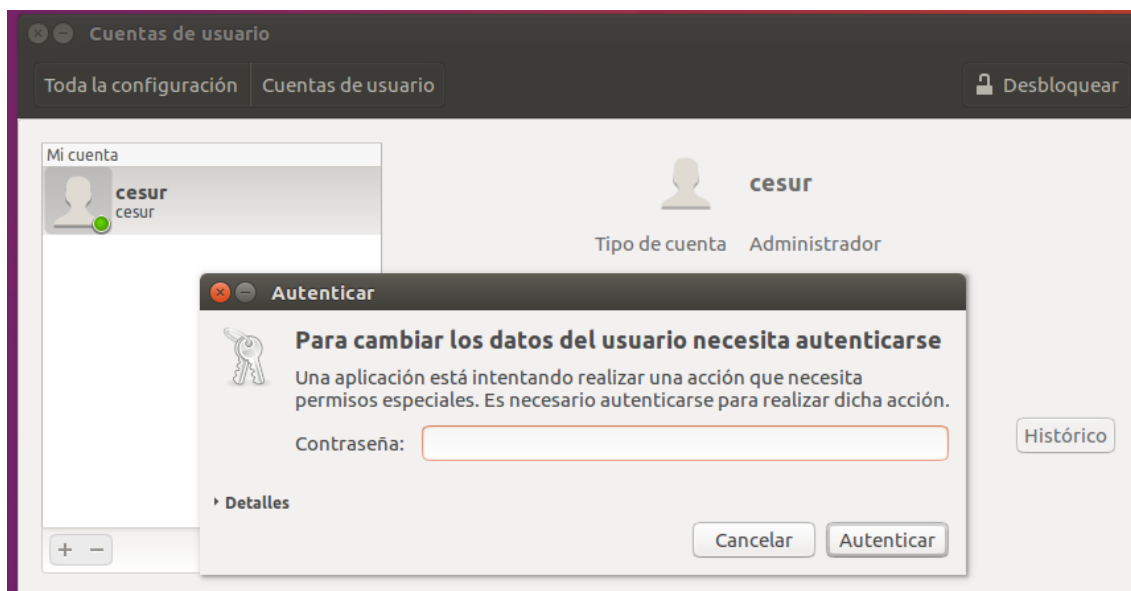


Imagen: Autenticación para desbloquear la creación de usuarios

Hecho esto, se activará el botón "+", que como se indicaba anteriormente debe pulsarse para crear el usuario nuevo

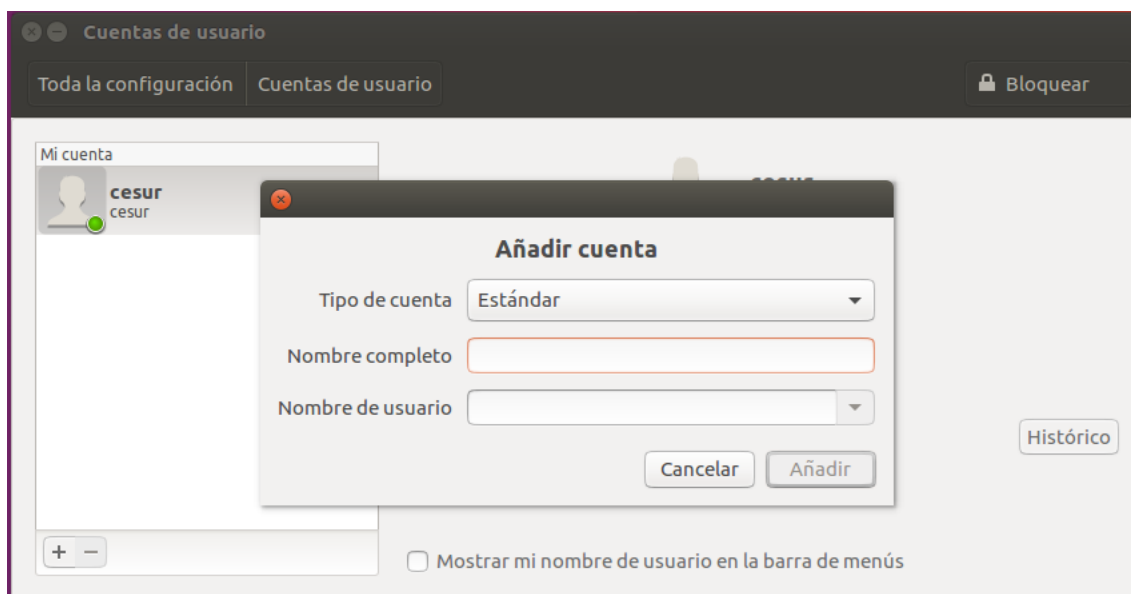


Imagen: Creación de un usuario nuevo

Los campos que se deben rellenar son similares a los que se solicitan en los sistemas Windows: el tipo de cuenta que se quiere crear, el nombre de usuario deseado y su nombre completo. Lo habitual es crear una cuenta de tipo "Estándar".

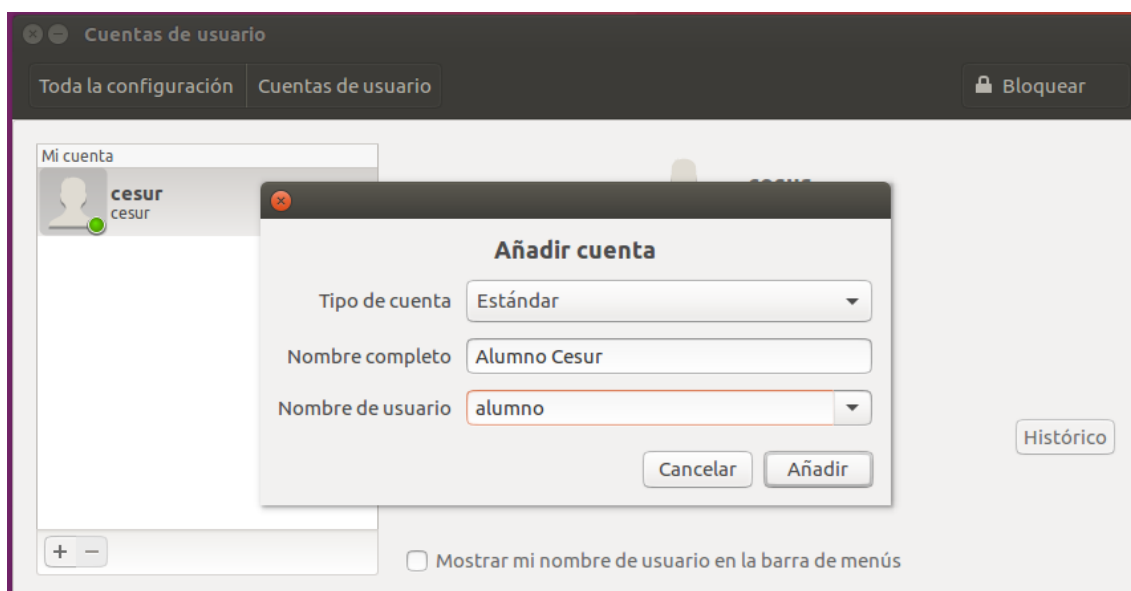


Imagen: Configuración del nuevo usuario

Por defecto la nueva cuenta se crea desactivada. Para hacerla operativa se debe pulsar sobre "**Cuenta desactivada**"

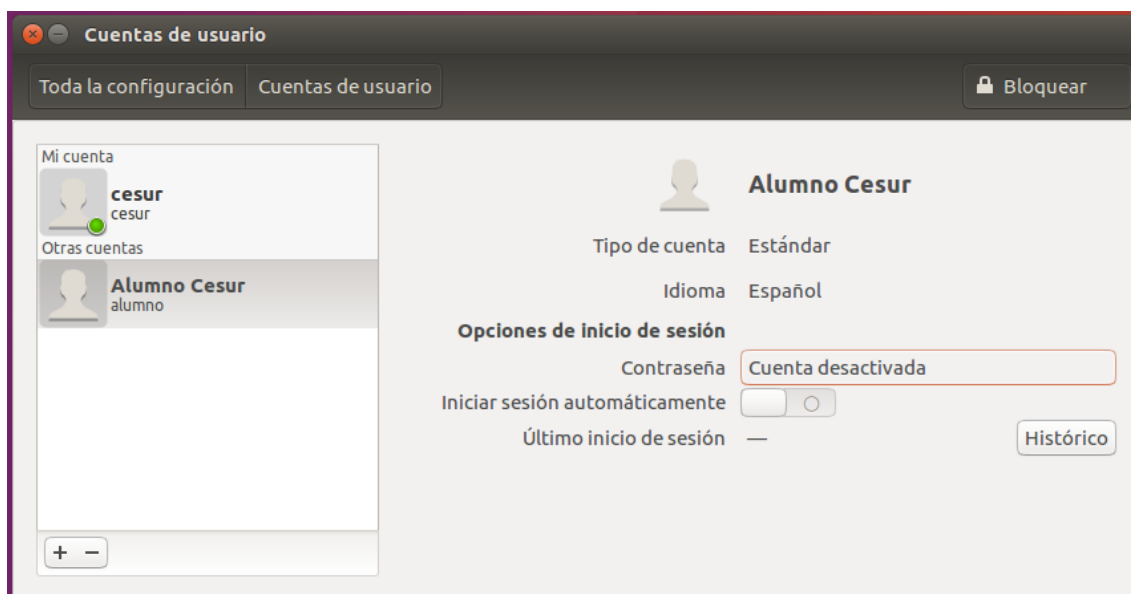


Imagen: Activación de la nueva cuenta de usuario

e introducir una contraseña para ella



Imagen: Asignación de contraseña al nuevo usuario

Terminado el proceso, se dispone de una nueva cuenta de usuario, que estará disponible al iniciar sesión en el equipo.

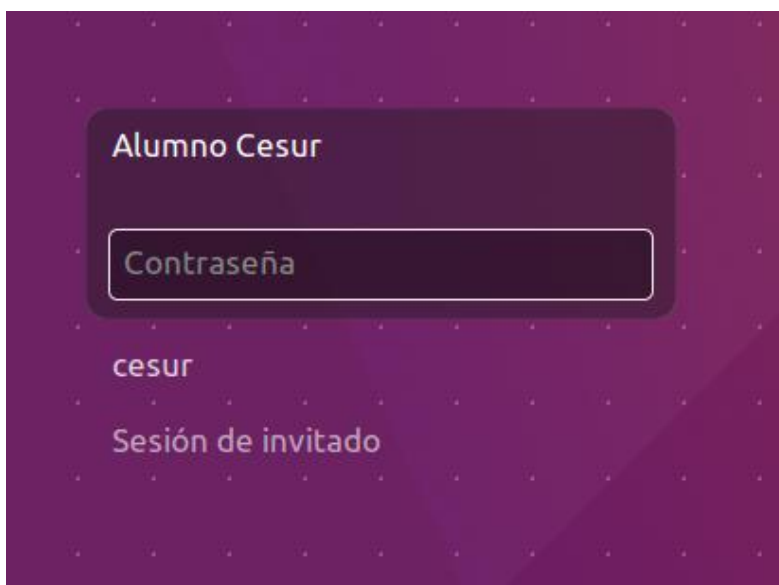


Imagen: Inicio de sesión en el sistema, ya con la nueva cuenta disponible

Si lo que se desea es **eliminar un usuario** ya creado, en la herramienta Cuentas de usuario habrá que seleccionar dicho usuario y utilizar el botón “-”, situado, como el “+” que utilizamos para crear un usuario nuevo, en la esquina inferior izquierda.

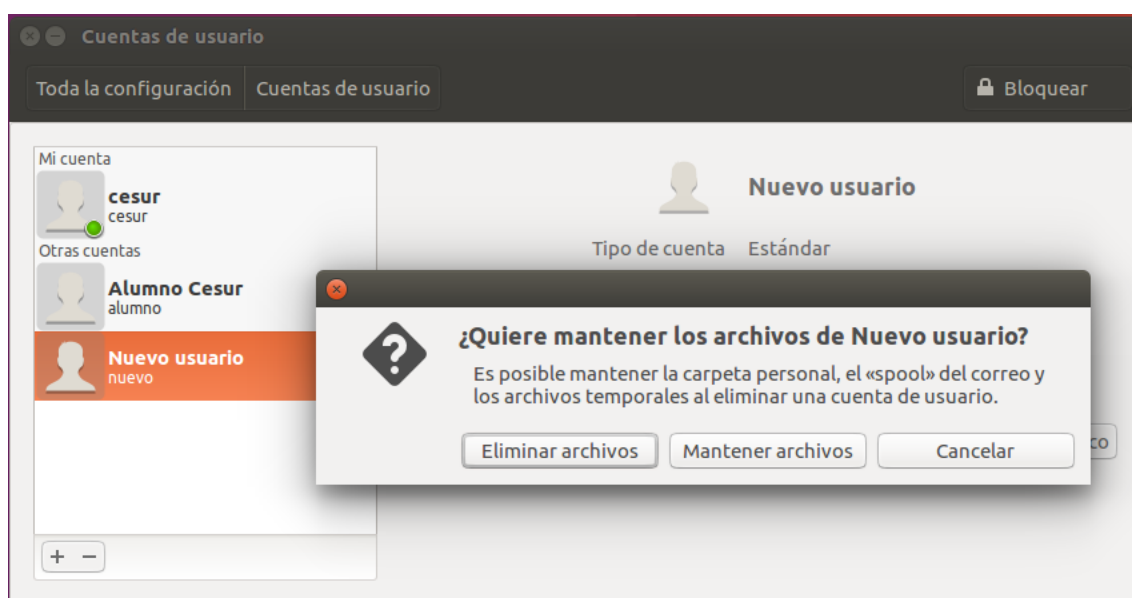


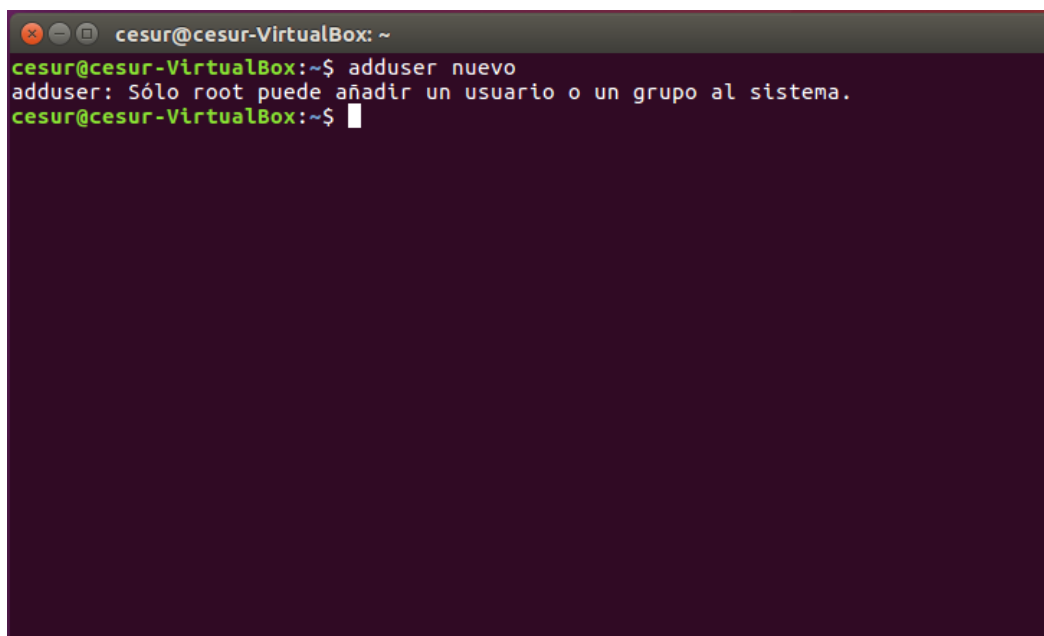
Imagen: Eliminación de un usuario

En el momento de eliminar al usuario seleccionado se nos pregunta, como se aprecia en la anterior imagen, si deseamos mantener o eliminar los archivos personales del usuario en cuestión. Lo habitual será eliminarlos, para liberar espacio en el sistema.

1.2.2. A través de comandos. **USERADD y USERDEL**

Para **añadir un usuario** mediante comandos en Ubuntu se debe abrir la herramienta Terminal y usar el comando **adduser**. Dicho comando nos permite agregar a un usuario de manera más interactiva, aunque internamente se usa el comando **useradd**.

Adduser utiliza como primer parámetro el nombre del nuevo usuario. Pero al ejecutar directamente la sentencia, por ejemplo **adduser nuevo**, se muestra un aviso de que tenemos que emplear el **comando sudo** (que nos permite obtener privilegios de root de forma puntual, para realizar una operación concreta) ya que se necesitan permisos de administrador para crear un nuevo usuario.

A screenshot of a terminal window titled 'cesur@cesur-VirtualBox: ~'. The prompt is 'cesur@cesur-VirtualBox:~\$'. The user has entered the command 'adduser nuevo'. The terminal output shows the command being executed, followed by an error message: 'adduser: Sólo root puede añadir un usuario o un grupo al sistema.' The prompt returns to 'cesur@cesur-VirtualBox:~\$' with a cursor at the end.

```
cesur@cesur-VirtualBox:~$ adduser nuevo
adduser: Sólo root puede añadir un usuario o un grupo al sistema.
cesur@cesur-VirtualBox:~$
```

Imagen: Indicación de la necesidad de privilegios de root para crear un usuario

Se ejecuta el comando precedido de **sudo**, y ahora sí que el proceso se puede completar con éxito. Se van solicitando los datos relativos al usuario a la vez que añaden los directorios y ficheros que usará. Una vez finalizada la operación, se habrá creado el usuario, su grupo, directorio personal, ficheros básicos, contraseña e información adicional relativa a dicho usuario.

```
cesur@cesur-VirtualBox: ~  
cesur@cesur-VirtualBox:~$ sudo adduser nuevo  
[sudo] password for cesur:  
Añadiendo el usuario 'nuevo' ...  
Añadiendo el nuevo grupo 'nuevo' (1002) ...  
Añadiendo el nuevo usuario 'nuevo' (1002) con grupo 'nuevo' ...  
Creando el directorio personal '/home/nuevo' ...  
Copiando los ficheros desde '/etc/skel' ...  
Introduzca la nueva contraseña de UNIX:  
Vuelva a escribir la nueva contraseña de UNIX:  
passwd: contraseña actualizada correctamente  
Cambiando la información de usuario para nuevo  
Introduzca el nuevo valor, o presione INTRO para el predeterminado  
Nombre completo []: Nuevo usuario  
Número de habitación []: 1  
Teléfono del trabajo []: 952102030  
Teléfono de casa []: 952203040  
Otro []:  
¿Es correcta la información? [S/n] s  
cesur@cesur-VirtualBox:~$
```

Imagen: Introducción de datos para el nuevo usuario

Otra alternativa es emplear directamente el comando **useradd**. Esta opción agregará un usuario de manera rápida, sin preguntar ningún dato relativo al mismo. Lógicamente, después de crear el usuario de este modo es recomendable asignarle una contraseña de forma inmediata.

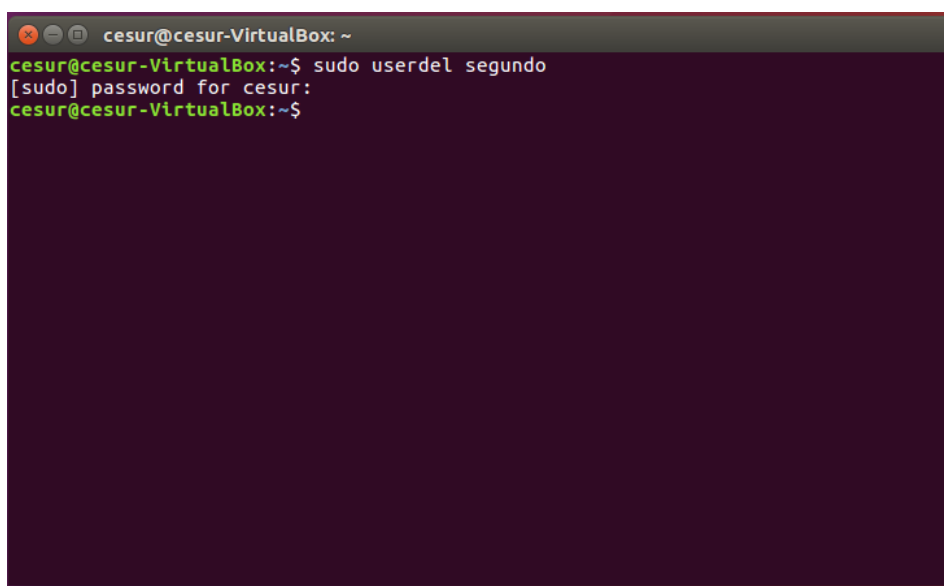
```
cesur@cesur-VirtualBox: ~  
cesur@cesur-VirtualBox:~$ sudo useradd segundo  
[sudo] password for cesur:  
cesur@cesur-VirtualBox:~$
```

Imagen: Creación de usuario mediante useradd

Algunas de las opciones de este comando son las siguientes:

- -c → añade un comentario
- -d → especifica un directorio de trabajo
- -f → establece la cuenta como inactiva
- -p → especifica una contraseña para el usuario

Para **eliminar un usuario** desde el Terminal se hace uso del comando **userdel**, indicando el nombre del usuario que se quiere eliminar



```
cesur@cesur-VirtualBox: ~  
cesur@cesur-VirtualBox:~$ sudo userdel segundo  
[sudo] password for cesur:  
cesur@cesur-VirtualBox:~$
```

Imagen: Eliminación de usuario mediante userdel

1.3 Seguridad en cuentas de usuario

Cuando se almacena información en un sistema de computación, una de las mayores preocupaciones es protegerla de daños físicos (confiabilidad) y del acceso inadecuado (confidencialidad).

La **confiabilidad** generalmente se obtiene creando copias de los archivos. Muchos computadores tienen programas que automáticamente copian archivos a intervalos regulares, para poder usar una de estas copias en caso de que el sistema de archivos se destruya accidentalmente. Un sistema de archivos puede ser dañado por problemas de hardware (errores en la lectura o escritura), cortes o sobrecargas de energía...

La **confidencialidad** o protección puede ofrecerse de varias maneras. La necesidad de protección surge de la capacidad de acceder a los archivos en

aquellos sistemas donde no se permite el acceso a los archivos de otros usuarios.

El S.O. debe asegurar la confidencialidad de la información almacenada en el sistema de ficheros. Para ello, debe establecer qué usuarios pueden acceder a qué ficheros. Se deben tomar medidas para la correcta identificación de los usuarios, como la utilización de passwords. Para acceder a sistemas multiusuario, el interesado debe introducir un nombre de usuario y un password. Cada nombre de usuario tiene siempre una clave de acceso o password asociado.

1.3.1 Seguridad de contraseñas

Las **contraseñas** son el principal método que utilizan los sistemas operativos para validar los usuarios. De ese modo se autentifica que el usuario es quien dice ser. Por este motivo, la seguridad de las contraseñas es la principal manera de proteger al usuario, al equipo y a la red contra el acceso no autorizado; las contraseñas nunca deben ser compartidas, ni siquiera con familiares cercanos, y deben ser seguras en el sentido de ser lo suficientemente complejas como para que nadie pueda adivinarlas o intuir las.



ENLACE DE INTERÉS

En la web oficial de la Oficina de Seguridad del Internauta encontramos información muy útil para la gestión de nuestras contraseñas:

<https://www.osi.es/es/contrasenas>

Las contraseñas seguras deben cumplir una premisa: **su longitud debe ser de 8 caracteres o más**, cada carácter de más que se adicione a la contraseña hace que su seguridad aumente exponencialmente. Una contraseña segura debe contener tres de las cuatro características que se exponen a continuación.

1. Letras minúsculas.
2. Letras mayúsculas.
3. Números.
4. Caracteres no alfanuméricos o símbolos (\$, @, &, #, {, ?, etc...).

Aparte de contener 3 de las 4 características, las contraseñas deben cumplir los siguientes requisitos:

- No debe contener información que sea fácil de averiguar como el nombre del usuario de la cuenta y la información personal de dicho usuario.
- No debe contener palabras existentes en algún idioma, ni siquiera invertidas.
- No debe formarse con números y/o letras que estén adyacentes en el teclado.
- No debe escribirse en ningún sitio, por lo que debe ser fácil de recordar.

**RECUERDA**

TUS CONTRASEÑAS DEBEN SER...


SECRETAS


ROBUSTAS


NO REPETIDAS


CAMBIADAS PERIÓDICAMENTE

2. GESTIÓN DE GRUPOS

En un sistema personal, doméstico, es habitual disponer de un número reducido de usuarios, cuya gestión se puede realizar de forma manual, del modo descrito en el apartado anterior. Pero en sistemas más complejos, como puede ser el caso de una empresa, el número de usuarios será mucho más elevado, y esta gestión manual de sus características se puede convertir en inviable.

Para solucionar este problema surge el concepto de **grupo**, que puede englobar a diferentes usuarios. Mediante los grupos se mejora la gestión de los usuarios, ya que se podrán realizar modificaciones (por ejemplo, añadir/eliminar una serie de permisos) sobre todos los usuarios del grupo de manera simultánea.

2.1. Gestión de grupos en sistemas Windows

Se utiliza la herramienta, ya citada en el apartado anterior, de **Administración de equipos**. Se accede a ella, recordamos, desde el Panel de control → Herramientas administrativas.

En este caso se seleccionará la opción **Grupos** del menú lateral izquierdo, que aparece justo a continuación de la de Usuarios.

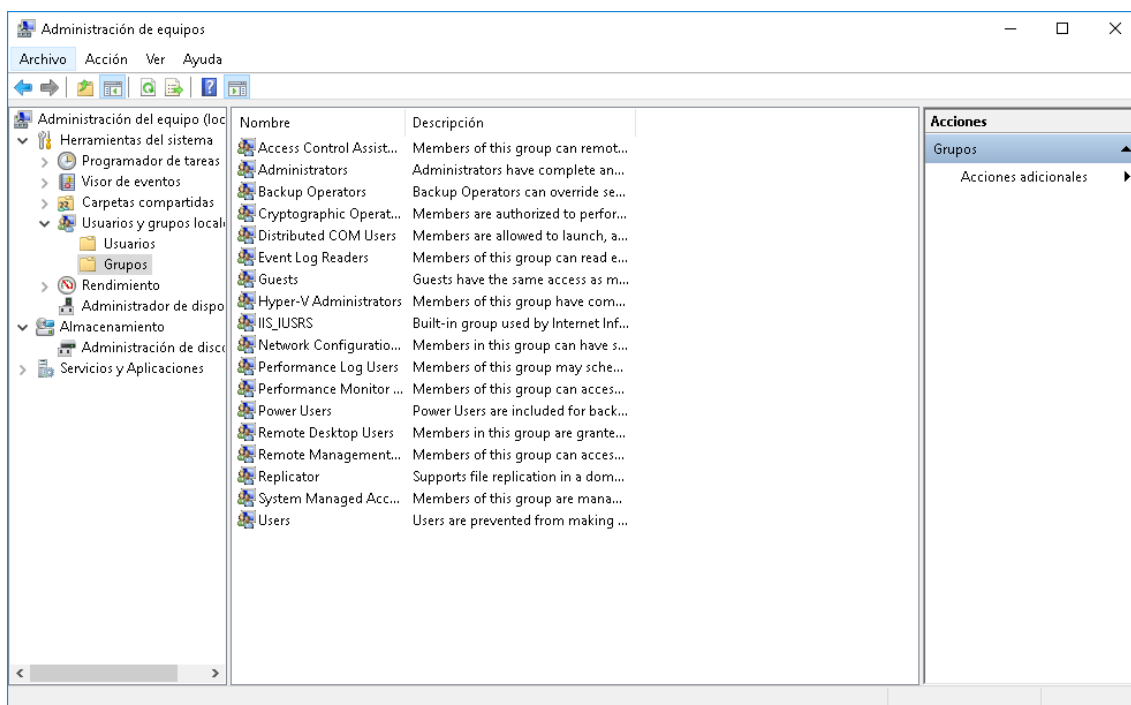


Imagen: Administración de grupos en Windows 10

Como se aprecia en la imagen anterior, algunos grupos de usuario que aparecen por defecto al realizar una instalación de Windows son:

- Administradores. El administrador que se crea en la instalación del sistema operativo, así como todas las cuentas creadas con este perfil, pertenecen a este grupo. Tienen acceso sin restricciones a todo el sistema.
- Usuarios. El resto de usuarios con el simple hecho de darse de alta ya pertenecen a este grupo. Tienen privilegios reducidos.
- Invitados. Similar a la de usuarios, pero cuenta con más restricciones

La gestión de grupos se realiza de forma muy similar a la de usuarios. Por ejemplo, se puede **crear un grupo nuevo** tanto con el menú contextual del botón derecho como con el menú Acción, eligiendo en ambos casos la opción **Grupo nuevo**.

Imagen: Creación de un grupo nuevo

Como puede verse en la imagen anterior, para crear un nuevo grupo debe facilitarse su nombre y una descripción, y agregar como miembros a los distintos usuarios que se desee que formen parte del grupo.

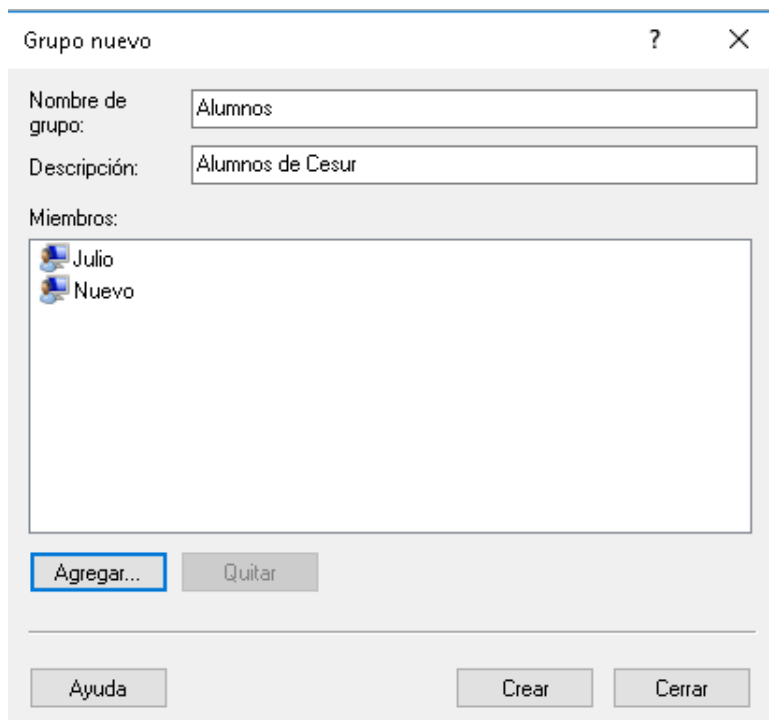


Imagen: Grupo al que se han agregado los alumnos creados previamente

2.2 Gestión de grupos en sistemas Linux

Para poder administrar los permisos de los usuarios de una forma más flexible, los sistemas Linux permiten la organización de usuarios en grupos y establecer permisos a dichos grupos. Por ejemplo, si en un centro educativo el grupo "profesores" tiene acceso a ciertas carpetas, cuando demos de alta un profesor nuevo, tan solo tendremos que añadirle al grupo "profesores" para que pueda acceder a todas esas carpetas. Es lo que se denomina administración de permisos por grupos.

En Linux, todos los usuarios pertenecen al menos a un grupo que es el **grupo principal del usuario**, también llamado grupo primario del usuario, pero pueden pertenecer a más grupos. En caso de que pertenezcan a más grupos, éstos serán **grupos secundarios**.

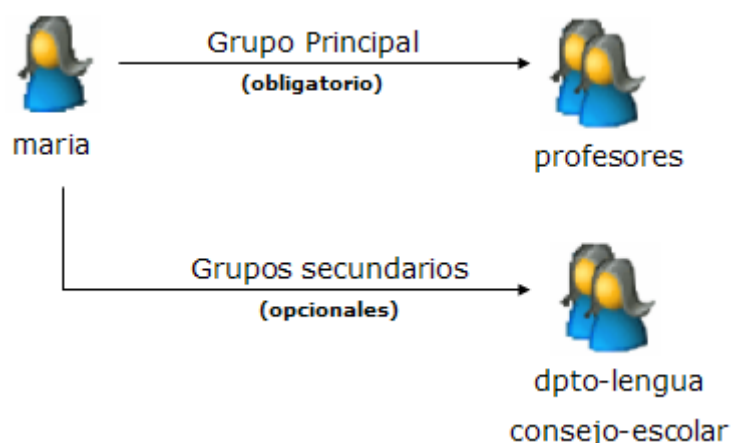


Imagen: Usuario perteneciente a varios grupos



ENLACE DE INTERÉS

En el siguiente enlace se explica el significado de los principales grupos existentes en Linux:

https://www.linuxtopia.org/online_books/espaniol/debian_linux_guides/debian_linux_reference_guide/ch-tune.es_007.html

Linux codifica los grupos de usuarios asignando un número diferente a cada uno, que es el **identificador de grupo (gid = *Group Identifier*)**. Internamente el sistema trabaja con el gid, no con el nombre del grupo. Normalmente a los grupos que creamos se les asignan gids desde 1000 en adelante. Los números gid menores que 100 se reservan para grupos especiales del sistema.

En Linux, por defecto, la información de los grupos de un sistema se guarda en el archivo **/etc/group**. Es un archivo de texto que puede visualizarse con cualquier editor. Cada línea del archivo /etc/group almacena los parámetros del grupo y los usuarios que contiene. Solo puede modificarlo el administrador (root). Las contraseñas de los grupos se guardan encriptadas con un sistema de codificación irreversible, en el archivo **/etc/gshadow** que también es un archivo de texto.

3. ACCESO A RECURSOS. PERMISOS LOCALES

Los permisos de cada fichero son la protección más básica de estos objetos en el sistema operativo: definen quién puede acceder a cada uno de ellos, y de qué forma puede hacerlo. A continuación se describe el modo en que se trabaja con estos permisos tanto en sistemas operativos libres como propietarios.

3.1. Permisos locales en Linux

Cada uno de los elementos de un sistema de ficheros Linux posee permisos de acceso de acuerdo a tres tipos de usuarios: Usuario, Grupo y Otros.

El usuario es el dueño del fichero, el grupo es un identificador de un conjunto de usuarios al que pertenece el archivo y otros es el resto de los usuarios.

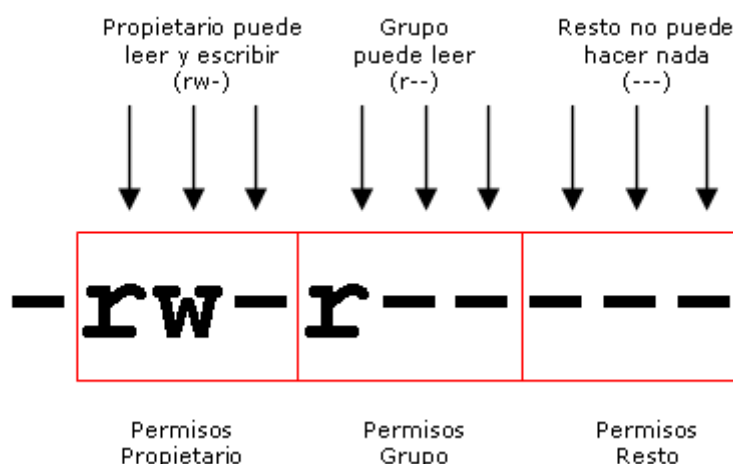
Para cada uno de estos tres grupos de usuarios existen tres tipos de permisos fundamentales:

r: read (lectura).	<ul style="list-style-type: none"> El usuario que tenga este permiso podrá si es un directorio, listar los recursos almacenados en él, y si es cualquier otro tipo de fichero podrá leer su contenido.
w: write (escritura).	<ul style="list-style-type: none"> Todo usuario que posea este permiso para un fichero podrá modificarlo. Si se posee para un directorio se podrán crear y borrar ficheros en su interior.
x: execute (ejecución).	<ul style="list-style-type: none"> Este permiso para el caso de los ficheros permitirá ejecutarlos desde la línea de comandos y para los directorios, el usuario que lo posea tendrá acceso para realizar el resto de las funciones permitidas mediante los otros permisos (lectura y/o escritura).

Por lo tanto, un determinado archivo en el sistema de ficheros tiene especificados sus tres permisos para cada uno de los tres tipos de usuarios que hemos citado.

Así pues, los permisos especifican si el dueño del fichero tiene permiso de lectura, escritura y ejecución, así como también especifican los permisos para el grupo dueño del archivo. Todo usuario que pertenezca al grupo al que pertenece el archivo, tendrá los permisos dados a ese grupo por el archivo en cuestión. Si el usuario que pretende manipular el archivo no es

su dueño ni pertenece a su grupo, le serán aplicados los permisos especificados en el archivo para otros usuarios.



Para determinar los permisos siempre se deben tener en cuenta los siguientes aspectos:

- Para poder realizar operaciones sobre cualquier directorio (leer o escribir) será necesario siempre tener otorgado además el permiso de ejecución.
- Para acceder a un recurso de cualquier forma (ejecución, lectura o escritura) se deben tener permisos de ejecución para todos los directorios que contienen al recurso directa e indirectamente.

Los tres tipos de permisos mencionados poseen una representación numérica basada en el sistema octal que parte de representar como "1" los bits de los permisos otorgados y "0" para los negados. Luego se transforma la representación binaria así obtenida en octal.

La combinación de los tres tipos de permisos para un tipo de usuario oscila desde cero (ningún permiso) hasta siete (todos los permisos).

Por ejemplo:

- "-rwxr-xr-x" se representa como 755 en notación octal.
- "-rw-rw-r--" se representa como 664 en notación octal.
- "-r-x-----" se representa como 500 en notación octal.

En cualquier momento se pueden **comprobar los permisos** que tiene asignados un recurso en un sistema Linux. Basta con utilizar el comando ls.


```
cesur@cesur-VirtualBox: ~/pruebas
cesur@cesur-VirtualBox:~/pruebas$ ls -l
total 0
-rw-rw-r-- 1 cesur cesur 0 sep  3 17:40 carta.txt
cesur@cesur-VirtualBox:~/pruebas$
```

Imagen: Permisos de un archivo en Ubuntu

Como puede verse en la imagen anterior, justo al inicio de la descripción de un archivo mostrada por `ls -l`, se encuentran los permisos que tiene asignados, con la notación explicada con anterioridad.

También es bastante simple **modificar** dichos permisos. Para ello se emplea el comando **chmod**. Por ejemplo, para asignar todos los permisos al archivo `carta.txt` se hace: **chmod 777 carta.txt**

```
cesur@cesur-VirtualBox: ~/pruebas
cesur@cesur-VirtualBox:~/pruebas$ chmod 777 carta.txt
cesur@cesur-VirtualBox:~/pruebas$ ls -l
total 0
-rwxrwxrwx 1 cesur cesur 0 sep  3 17:40 carta.txt
cesur@cesur-VirtualBox:~/pruebas$
```

Imagen: Asignación de permisos con chmod



ENLACE DE INTERÉS

Tal como hemos visto, los permisos se pueden expresar en notación octal y simbólica. Para ampliar información se recomienda visitar la web:

https://es.wikipedia.org/wiki/Permisos_de_acceso_a_archivos

3.2 Permisos locales en Windows

En los sistemas Windows, los permisos se asignan a los ficheros y directorios a través de las **Propiedades** de dichos recursos. Se puede acceder a dichas propiedades mediante el menú contextual del botón derecho, pulsando sobre el archivo o directorio deseado.

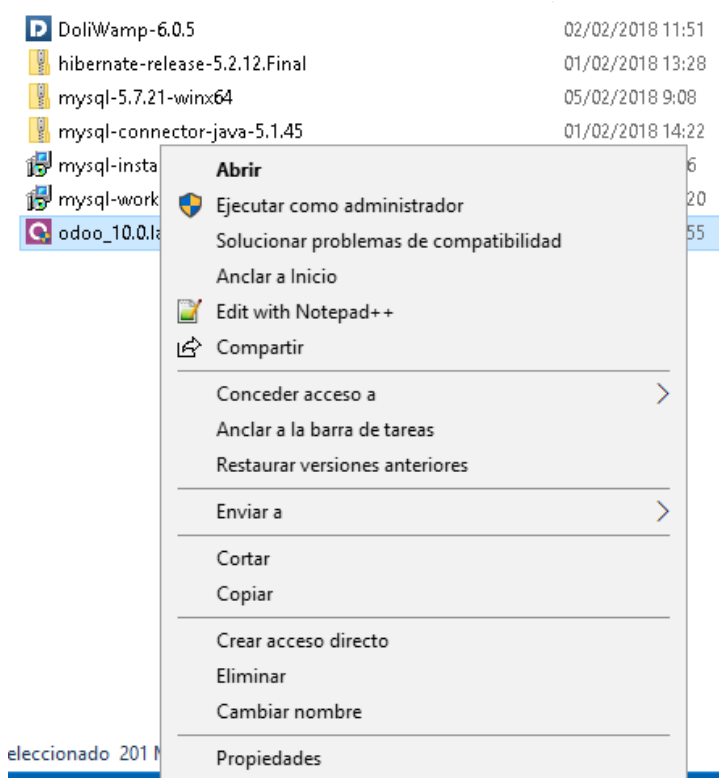


Imagen: Acceso a las propiedades de un archivo en Windows 10

Dentro de la ventana de Propiedades del objeto, para la gestión de los permisos hay que acceder a la pestaña Seguridad.

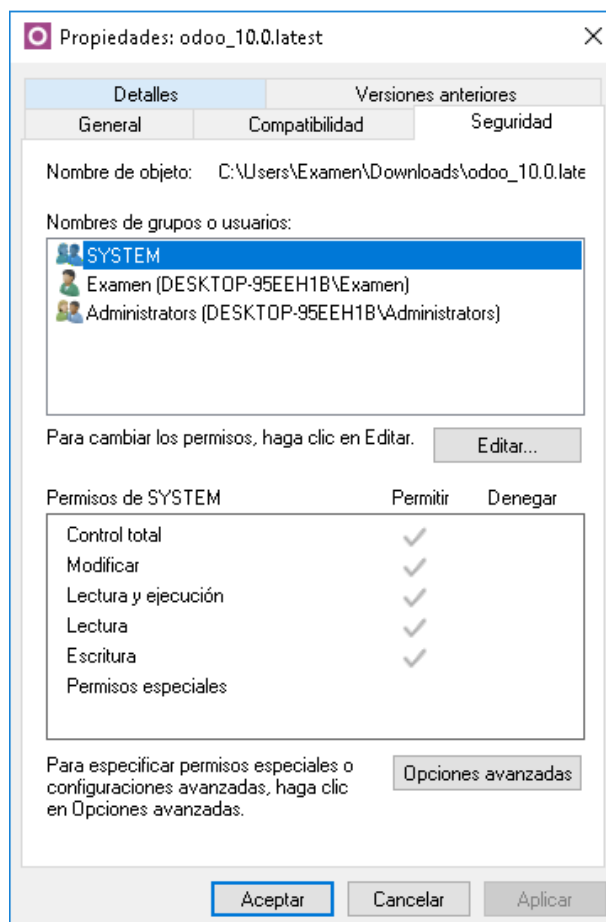


Imagen: Propiedades de Seguridad de un archivo

Esta pestaña, como se aprecia en la imagen anterior, se encuentra dividida en dos zonas: la superior, que muestra los usuarios/grupos que tienen permisos sobre el recurso; y la inferior, que indica qué permisos son estos exactamente.

Si se desea modificar estos permisos hay que hacer uso del botón **Editar**, que nos permitirá agregar o quitar usuarios a los que asignar permisos; así como indicar exactamente cuáles de estos permisos se quiere permitir o denegar a cada uno de esos usuarios.

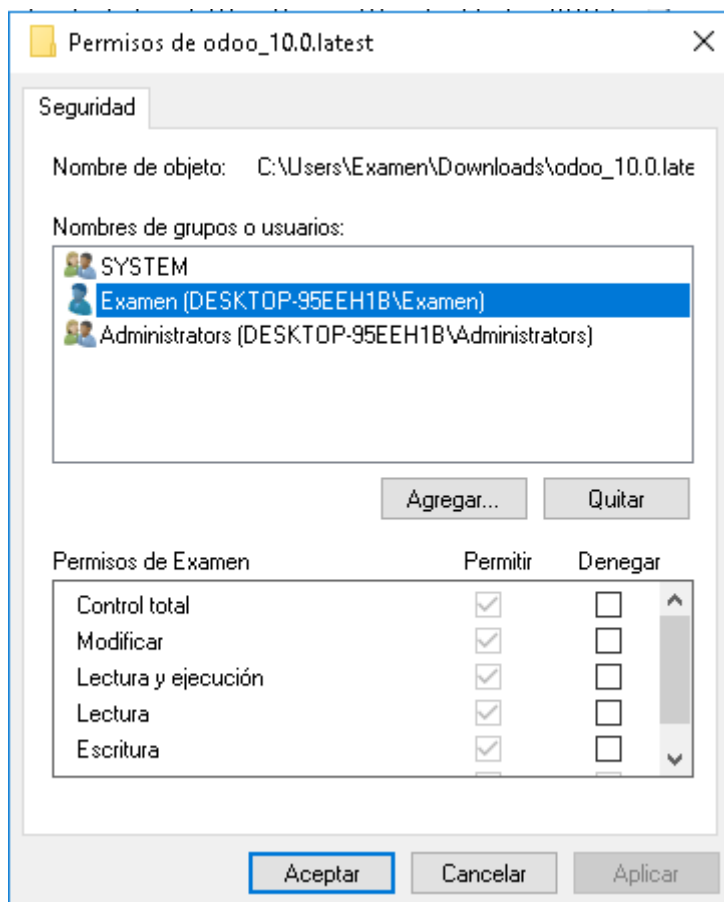


Imagen: Edición de permisos para el usuario Examen

4. PROCESOS

Un proceso simplemente **es un programa en ejecución**. Los procesos, además de la información propia del programa, contienen la información necesaria para que el programa interactúe con el sistema.

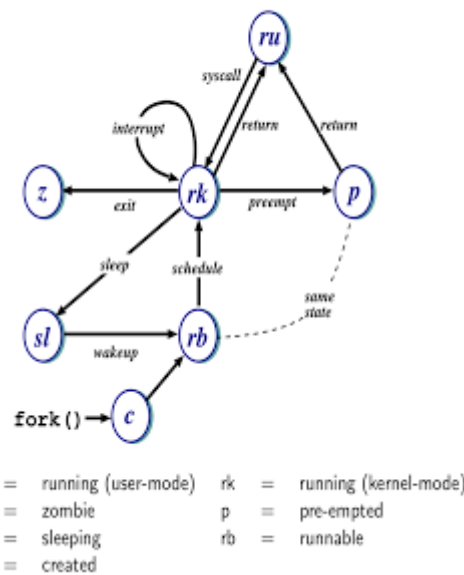
4.1 Tipos de procesos

En sistemas Unix se suelen distinguir los siguientes tipos de procesos:

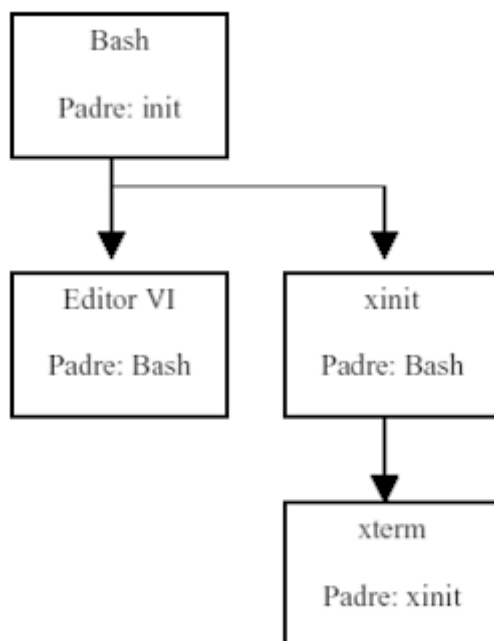
- **Child (hijos).** **Un proceso hijo es un proceso creado por otro proceso.** Se crean mediante la llamada al sistema `fork()` y en realidad, todos los procesos en algún momento son hijos, todos menos el proceso `init`. En el caso de que un proceso sea creado mediante la shell (ejecutado desde esta), la shell será el padre.

-

Unix Process States



- **Orphan** (huérfanos). Normalmente un proceso hijo termina antes que un proceso padre, pero se puede dar la situación de que **se mate a un proceso padre (killed)** y el hijo se quede sin padre. Entonces el proceso init lo adoptará como hijo, pero como su padre original no existe, es considerado huérfano.
- **Daemon** (demonios). Es un tipo especial de proceso que se ejecuta en segundo plano y no está asociado a ninguna shell. Esto se consigue matando la shell que crea el proceso, de esta forma el padre de este proceso pasa a ser el proceso init (queda huérfano). Estos corren con permisos de root y su cometido es proveer servicios a otros procesos.
- **Zombie**. Cuando un proceso hijo termina, **el sistema guarda el PID (Identificador) y su estado (un parámetro)** para dárselo a su padre. Hasta entonces el proceso finalizado entra en estado zombie. Cuando un proceso finaliza toda la memoria y recursos asociados con dicho proceso son liberados, pero la entrada del mismo en la tabla de procesos aún existe, para cuando su padre llame a la función wait() devolverle su PID y estado.

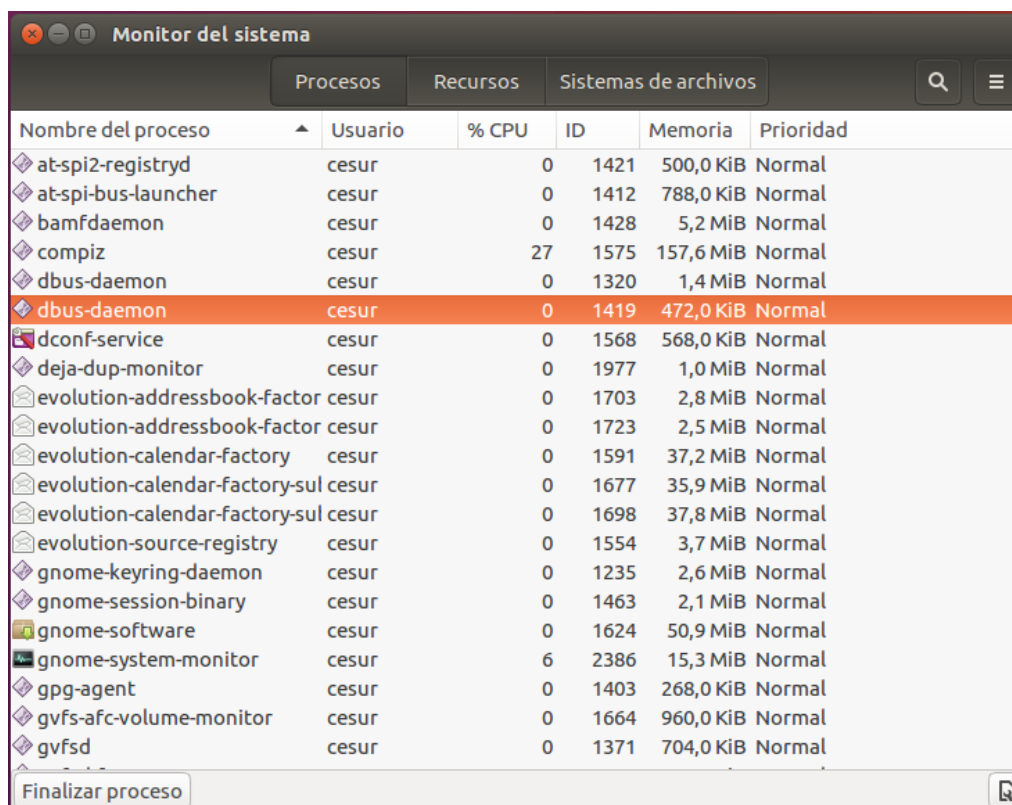


4.2 Administración de procesos en Linux.

Una vez más, la administración de los procesos en Linux se puede hacer bien mediante interfaz gráfica, bien mediante comandos.

4.2.1 Mediante Interfaz Gráfica

En el caso de Ubuntu la gestión de procesos de forma gráfica se realiza mediante el **Monitor del sistema**, al que se puede acceder, por ejemplo, utilizando la herramienta de búsqueda de Unity.



Nombre del proceso	Usuario	% CPU	ID	Memoria	Prioridad
at-spi2-registryd	cesur	0	1421	500,0 KiB	Normal
at-spi-bus-launcher	cesur	0	1412	788,0 KiB	Normal
bamfd daemon	cesur	0	1428	5,2 MiB	Normal
compiz	cesur	27	1575	157,6 MiB	Normal
dbus-daemon	cesur	0	1320	1,4 MiB	Normal
dbus-daemon	cesur	0	1419	472,0 KiB	Normal
dconf-service	cesur	0	1568	568,0 KiB	Normal
deja-dup-monitor	cesur	0	1977	1,0 MiB	Normal
evolution-addressbook-factor	cesur	0	1703	2,8 MiB	Normal
evolution-addressbook-factor	cesur	0	1723	2,5 MiB	Normal
evolution-calendar-factory	cesur	0	1591	37,2 MiB	Normal
evolution-calendar-factory-sul	cesur	0	1677	35,9 MiB	Normal
evolution-calendar-factory-sul	cesur	0	1698	37,8 MiB	Normal
evolution-source-registry	cesur	0	1554	3,7 MiB	Normal
gnome-keyring-daemon	cesur	0	1235	2,6 MiB	Normal
gnome-session-binary	cesur	0	1463	2,1 MiB	Normal
gnome-software	cesur	0	1624	50,9 MiB	Normal
gnome-system-monitor	cesur	6	2386	15,3 MiB	Normal
gpg-agent	cesur	0	1403	268,0 KiB	Normal
gvfs-afc-volume-monitor	cesur	0	1664	960,0 KiB	Normal
gvfsd	cesur	0	1371	704,0 KiB	Normal

Finalizar proceso

Imagen: Monitor del sistema en Ubuntu

El monitor nos informa del PID (Identificador del proceso), el espacio que ocupa y el porcentaje del procesador que está usando. Además, pulsando con el botón derecho sobre un proceso podemos matarlo, terminarlo, detenerlo o cambiar su nivel de prioridad.

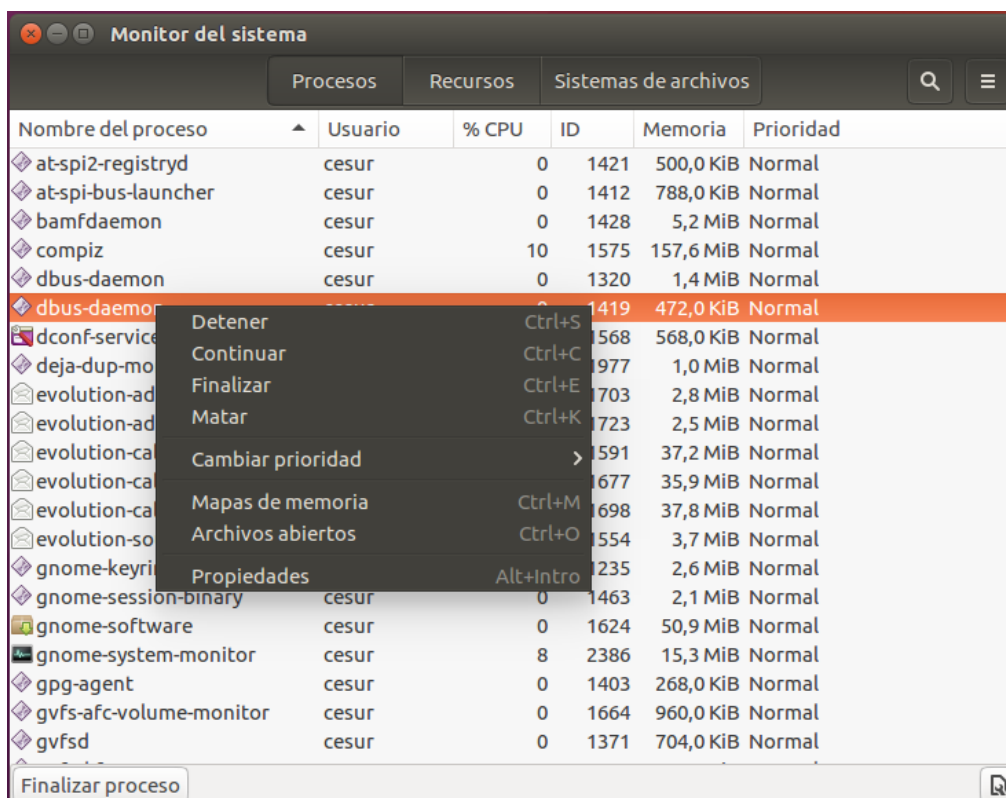


Imagen: menú contextual para un proceso

4.2.2 Mediante comandos

Para la administración de procesos mediante la línea de comandos se dispone de una serie de instrucciones que nos van a ayudar con este cometido. Los comandos están divididos en cuatro categorías: visualización, terminación de procesos, cambio de prioridad y ejecución en segundo plano.



ENLACE DE INTERÉS

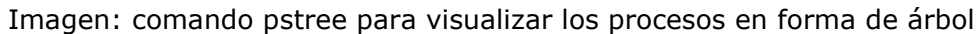
En el siguiente enlace hay un manual completo sobre los comandos usados para la gestión de procesos en Linux:

http://www.linuxtotal.com.mx/?cont=info_admon_012

- **Visualización de procesos.** Para visualizar el estado del proceso seleccionado o de todos los procesos se utiliza **ps**



Pero si lo que se desea es **monitorizar** los procesos en tiempo real se dispone del comando **top**, que muestra una lista de procesos que se actualiza cada 3 segundos (por defecto). Los procesos están ordenados por el uso de CPU y muestran PID, usuario, %CPU, %MEM.



```
cesur@cesur-VirtualBox: ~
top - 19:06:14 up 31 min, 1 user, load average: 0,03, 0,11, 0,18
Tareas: 157 total, 1 ejecutar, 156 hibernar, 0 detener, 0 zombie
%Cpu(s): 2,0 usuario, 0,3 sist, 0,0 adecuado, 97,6 inact, 0,0 en espera, 0,0
KiB Mem : 2045960 total, 779568 free, 731960 used, 534432 buff/cache
KiB Swap: 2095100 total, 2095100 free, 0 used. 1135656 avail Mem

  PID  USUARIO  PR  NI  VIRT  RES  SHR  S  %CPU  %MEM  HORA+  ORDEN
1575  cesur    20   0 1295396 250860 79404 S   1,7 12,3 2:07.21 compiz
821   root     20   0 423556 85080 32468 S   0,7 4,2 0:24.87 Xorg
2534  cesur    20   0 670008 38076 28708 S   0,3 1,9 0:01.39 gnome-termin+
2562  cesur    20   0 48952 3796 3236 R   0,3 0,2 0:00.09 top
1     root     20   0 119776 5920 3980 S   0,0 0,3 0:01.77 systemd
2     root     20   0 0 0 0 S   0,0 0,0 0:00.00 kthreadd
4     root     0 -20 0 0 0 S   0,0 0,0 0:00.00 kworker/0:0H
6     root     20   0 0 0 0 S   0,0 0,0 0:00.09 ksoftirqd/0
7     root     20   0 0 0 0 S   0,0 0,0 0:00.29 rcu_sched
8     root     20   0 0 0 0 S   0,0 0,0 0:00.00 rcu_bh
9     root     rt    0 0 0 0 S   0,0 0,0 0:00.00 migration/0
10    root     0 -20 0 0 0 S   0,0 0,0 0:00.00 lru-add-drain
11    root     rt    0 0 0 0 S   0,0 0,0 0:00.01 watchdog/0
12    root     20   0 0 0 0 S   0,0 0,0 0:00.00 cpuhp/0
13    root     20   0 0 0 0 S   0,0 0,0 0:00.00 kdevtmpfs
14    root     0 -20 0 0 0 S   0,0 0,0 0:00.00 netns
15    root     20   0 0 0 0 S   0,0 0,0 0:00.00 khungtaskd
16    root     20   0 0 0 0 S   0,0 0,0 0:00.00 oom_reaper
17    root     0 -20 0 0 0 S   0,0 0,0 0:00.00 writeback
18    root     20   0 0 0 0 S   0,0 0,0 0:00.00 kcompactd0
```

Imagen: comando top para monitorizar los procesos

- **Terminación de procesos.** Para matar un proceso se utiliza la instrucción **kill**, que a pesar de su nombre, no es una instrucción para matar procesos. Esta instrucción lo que hace es enviar una señal al proceso, que por defecto, es una señal **SIGTERM** que solicita al proceso limpiar su estado y salir (matar).

```
cesur@cesur-VirtualBox: ~/pruebas
cesur@cesur-VirtualBox:~/pruebas$ ps
  PID TTY          TIME CMD
 2258 pts/4        00:00:00 bash
 4361 pts/4        00:00:00 gedit
 4373 pts/4        00:00:00 ps
cesur@cesur-VirtualBox:~/pruebas$ kill 4361
[1]+  Terminado                  gedit
cesur@cesur-VirtualBox:~/pruebas$ ps
  PID TTY          TIME CMD
 2258 pts/4        00:00:00 bash
 4374 pts/4        00:00:00 ps
cesur@cesur-VirtualBox:~/pruebas$
```

Imagen: comando kill para terminar un proceso

- **Cambio de prioridad.** Los procesos por defecto tienen una prioridad de 0, pero es posible cambiar este valor para que el sistema trate con más o menos prioridad al proceso en cuestión. **Una prioridad de**

-10 es mayor que una prioridad de 10. Es decir, valores más bajos indican mayor prioridad.

Para cambiar la prioridad de un proceso se dispone de la instrucción **nice**.

```
cesur@cesur-VirtualBox: ~/pruebas
cesur@cesur-VirtualBox:~/pruebas$ nice -5 gedit &
[1] 4449
cesur@cesur-VirtualBox:~/pruebas$ ps -l
F S  UID  PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
0 S  1000  2258  2251  0  80   0 - 7440 wait  pts/4    00:00:00 bash
0 S  1000  4449  2258  4  85   5 - 165870 poll_s pts/4    00:00:00 gedit
0 R  1000  4456  2258  0  80   0 - 9009 -      pts/4    00:00:00 ps
cesur@cesur-VirtualBox:~/pruebas$
```

Imagen: comando nice para cambiar la prioridad de un proceso

- **Ejecutar procesos en segundo plano.** Si queremos ejecutar un proceso en segundo plano (background) se utiliza el **operador &**.

Si queremos que la terminal quede liberada de un proceso utilizamos **el operador &**. Esto se utiliza mucho cuando ejecutamos un programa con interfaz gráfica mediante la línea de comandos. Si no utilizamos el operador & (ampersand) después de la llamada al programa la terminal queda inutilizada (está ejecutando el programa), pero si utilizamos & el programa se ejecuta en segundo plano y podemos seguir ejecutando nuevos comandos.

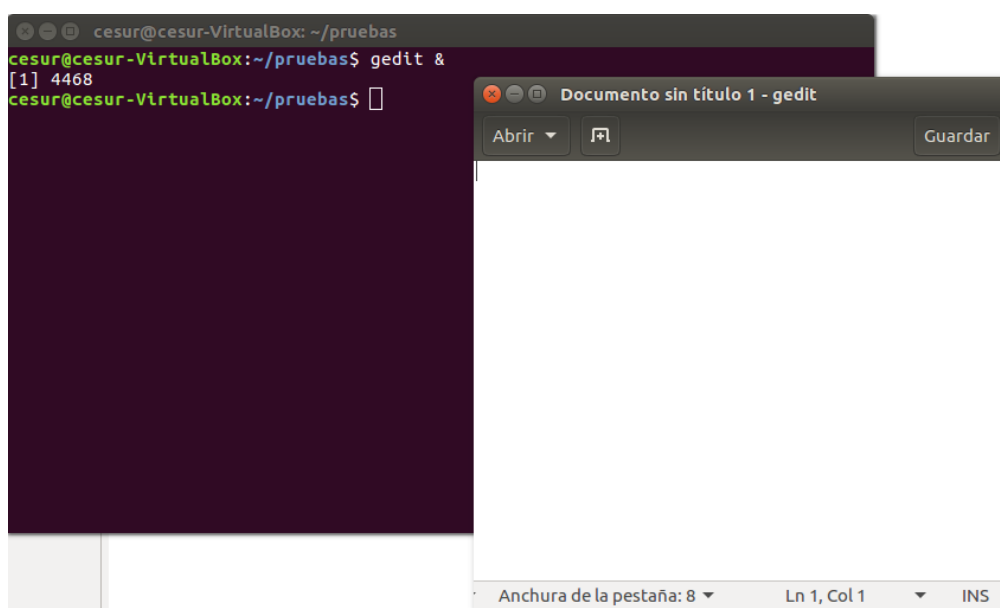


Imagen: operador & para abrir un proceso en segundo plano

5. SERVICIOS

Los servicios son procesos que se ejecutan en segundo plano, y por tanto no tienen interacción directa con el usuario.

Linux ofrece multitud de servicios, a los que denomina **daemon**, que se pueden iniciar o arrancar junto con la carga del sistema o pueden ser puestos a funcionar más adelante, cuando se requieran. Parte esencial de la administración de sistemas Linux es continuamente trabajar con los servicios que este proporciona, cosa que es bastante sencilla.

5.1 Administración de servicios en Linux

Linux nos permite iniciar o detener la mayoría de los servicios instalados en el equipo mediante dos opciones diferentes:

Iniciando servicios manualmente, escribiendo la ruta completa mediante una serie de scripts, el llamado directorio `init.d`.

Utilizando el comando `service`.

5.1.1. Iniciando servicios manualmente, directorio `init.d`

Dentro de esta carpeta, ubicada en `/etc` o en `/etc/rc.d` dependiendo de la distribución, se encuentran una serie de scripts que permiten iniciar/detener la gran mayoría de los servicios que estén instalados en el equipo. Estos scripts están programados de tal manera que la mayoría reconoce los siguientes argumentos:

- `start`
- `stop`
- `restart`
- `status`

Los argumentos son autodescriptivos y tienen permisos de ejecución, por lo que siendo `root` es posible iniciar un servicio de la siguiente manera:



Imagen: iniciando un servicio de forma manual

Solo que hay que cambiar start por stop | restart | status para detenerlo, reiniciarlo (releer archivos de configuración) o comprobar su estado.



ENLACE DE INTERÉS

En este enlace se explica cómo reiniciar servicios en Linux:

<http://es.wikihow.com/reiniciar-servicios-en-Linux>

5.1.2. El comando service

En varias distribuciones, como Fedora, RedHat o Ubuntu, existe el comando service, que permite iniciar y/o detener servicios exactamente igual que si escribiéramos la ruta completa hacia el directorio init.d, como hacíamos en el ejemplo anterior.

```
cesur@cesur-VirtualBox: ~/pruebas
cesur@cesur-VirtualBox:~/pruebas$ sudo service cups start
cesur@cesur-VirtualBox:~/pruebas$ service cups status
● cups.service - CUPS Scheduler
   Loaded: loaded (/lib/systemd/system/cups.service; enabled; vendor preset: ena
   Active: active (running) since lun 2018-09-03 18:44:49 CEST; 1min 11s ago
     Docs: man:cupsd(8)
    Main PID: 4838 (cupsd)
    CGroup: /system.slice/cups.service
            └─4838 /usr/sbin/cupsd -l
              └─4839 /usr/lib/cups/notifier/dbus dbus://
                └─4840 /usr/lib/cups/notifier/dbus dbus://

sep 03 18:44:49 cesur-VirtualBox systemd[1]: Started CUPS Scheduler.
sep 03 18:45:55 cesur-VirtualBox systemd[1]: Started CUPS Scheduler.
lines 1-12/12 (END)
```

Imagen: iniciando un servicio mediante el comando service

6. MONITORIZACIÓN Y MANTENIMIENTO DEL SISTEMA

Para conocer y mejorar el comportamiento de un sistema es necesario obtener información sobre las prestaciones de los diferentes subsistemas que lo componen. Todos los sistemas operativos incorporan herramientas que ayudan en este proceso de monitorización, a las que hay que añadir un sinnúmero de herramientas adicionales, tanto gratuitas como de pago, que complementan a las anteriores.

6.1 Herramientas propias de los sistemas operativos

Dentro de esta categoría se puede citar, por ejemplo, el **Monitor de rendimiento** de los sistemas Windows. Se puede acceder a él mediante el Panel de control, dentro de las Herramientas administrativas.

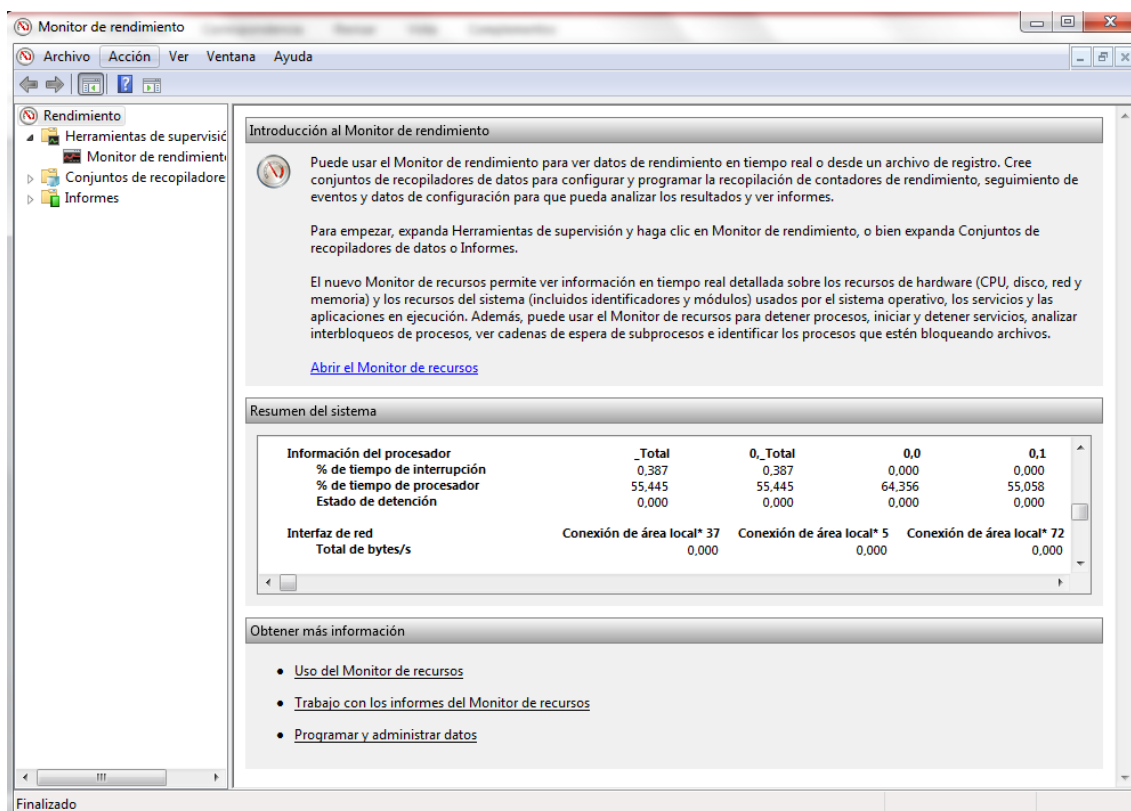


Imagen: monitor de rendimiento de Windows

Esta herramienta proporciona una serie de informes acerca de los principales subsistemas de nuestro equipo (CPU, Disco Duro, Red...). Además, proporciona acceso al Monitor de recursos, una interesante herramienta que permite comprobar el rendimiento de dichos subsistemas en tiempo real.

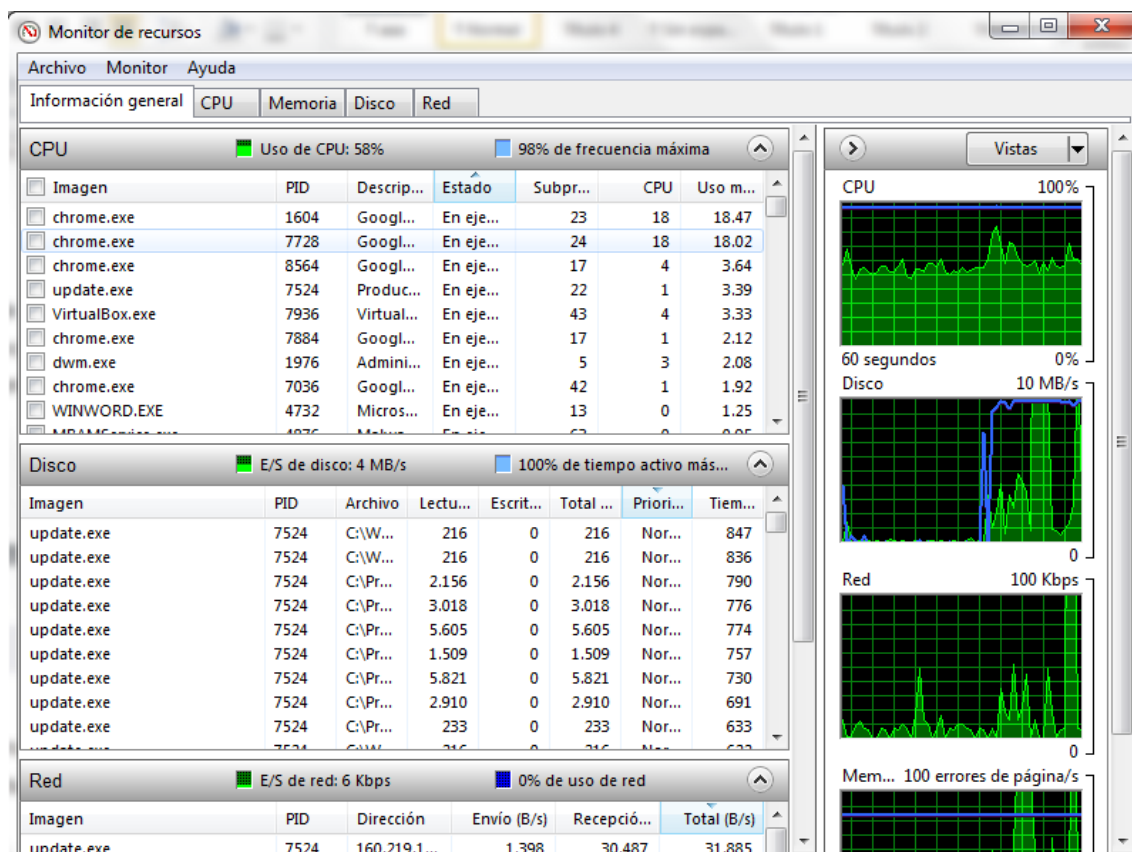


Imagen: monitor de recursos de Windows

En sistemas Linux tenemos herramientas similares, como el ya nombrado en esta unidad **Monitor del sistema**.

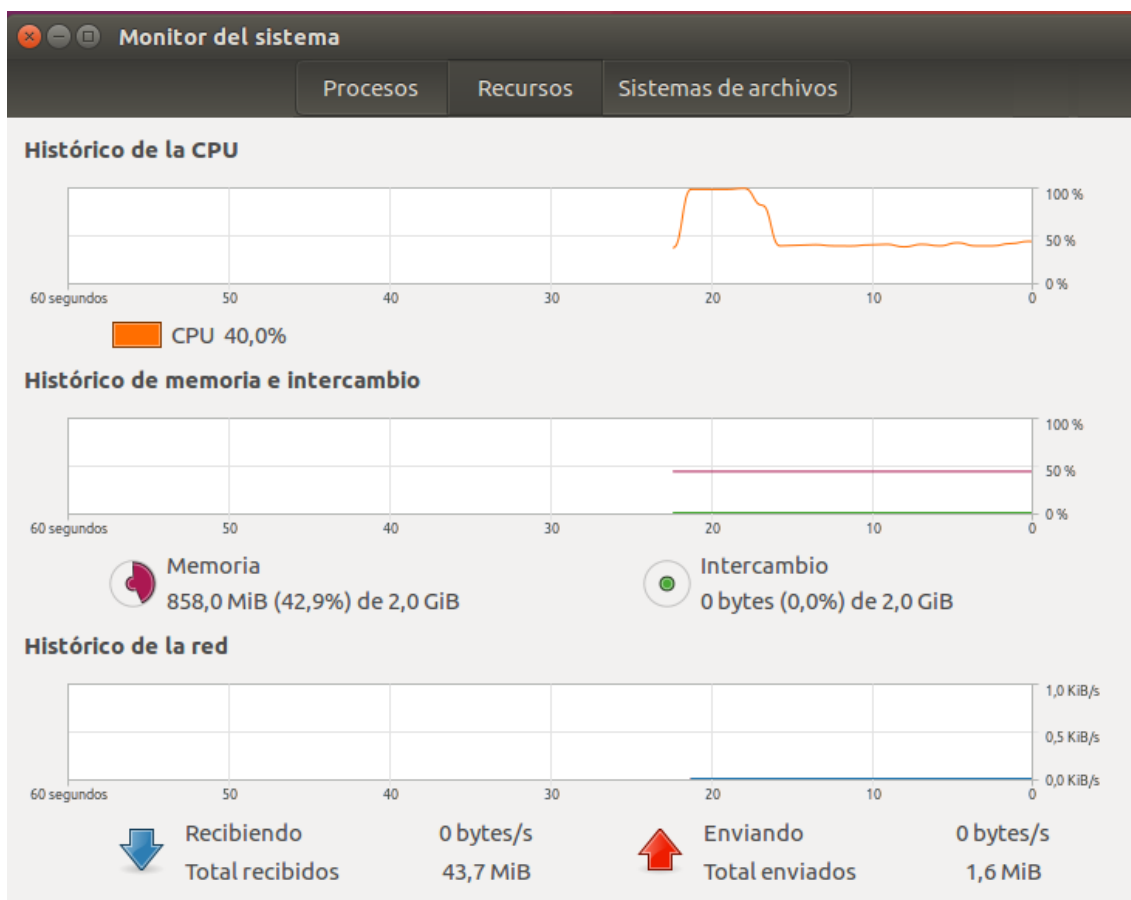


Imagen: monitor del sistema en Ubuntu

6.2 Herramientas complementarias.

HWInfo

Se trata de una aplicación diseñada para recoger y presentar al usuario la máxima cantidad de información posible sobre el hardware de su equipo.

Monitoriza en tiempo real todos los componentes del sistema, para ayudar a conocer su estado actual y de ese modo prevenir posibles fallos.

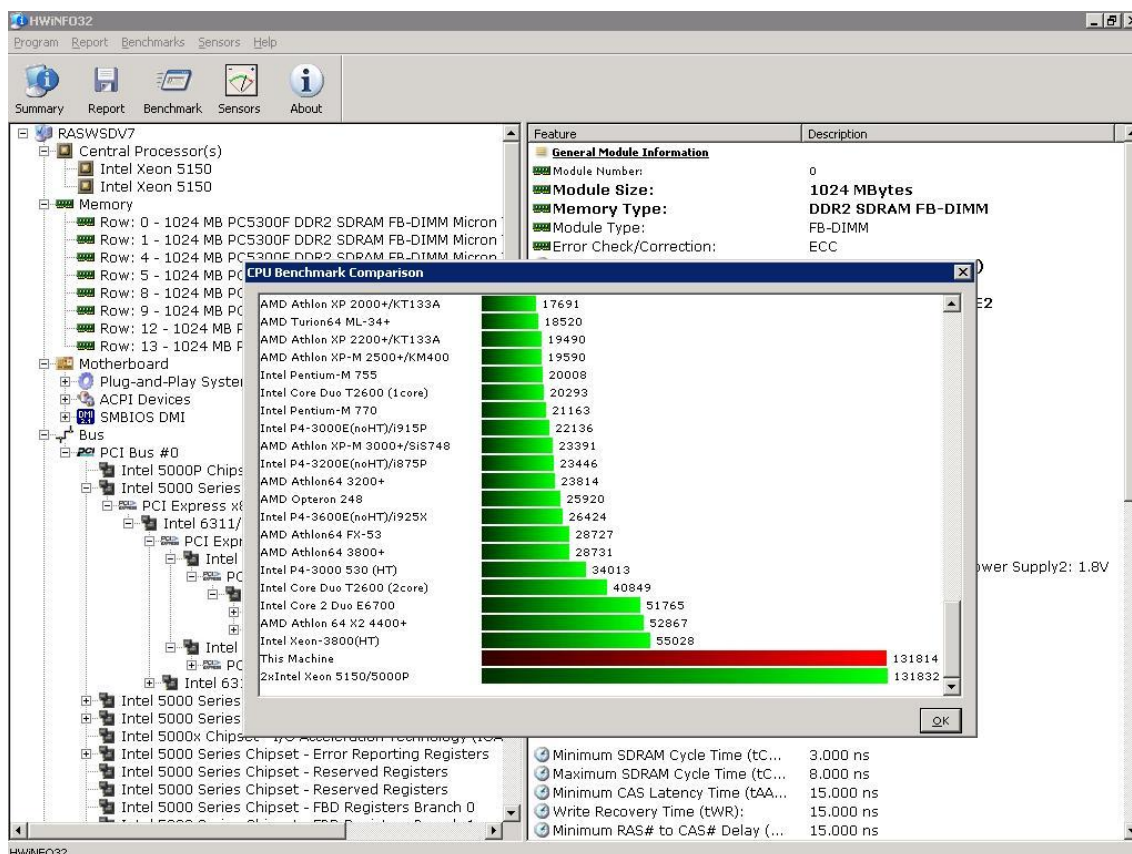


Imagen: aplicación HWInfo



ENLACE DE INTERÉS

Este es el enlace a la web oficial de la aplicación, en la que se puede ampliar información sobre sus prestaciones, y realizar su descarga.

<https://www.hwinfo.com/>

Nagios

Nagios es un sistema de monitorización ampliamente utilizado, de código abierto, que controla los equipos (hardware) y servicios (software) que se especifiquen, alertando cuando el comportamiento de los mismos no sea el deseado. Entre sus características principales figuran la monitorización de servicios de red (SMTP, POP3, HTTP, SNMP...) y la monitorización de los recursos de sistemas hardware (carga del procesador, uso de los discos, memoria, estado de los puertos...).

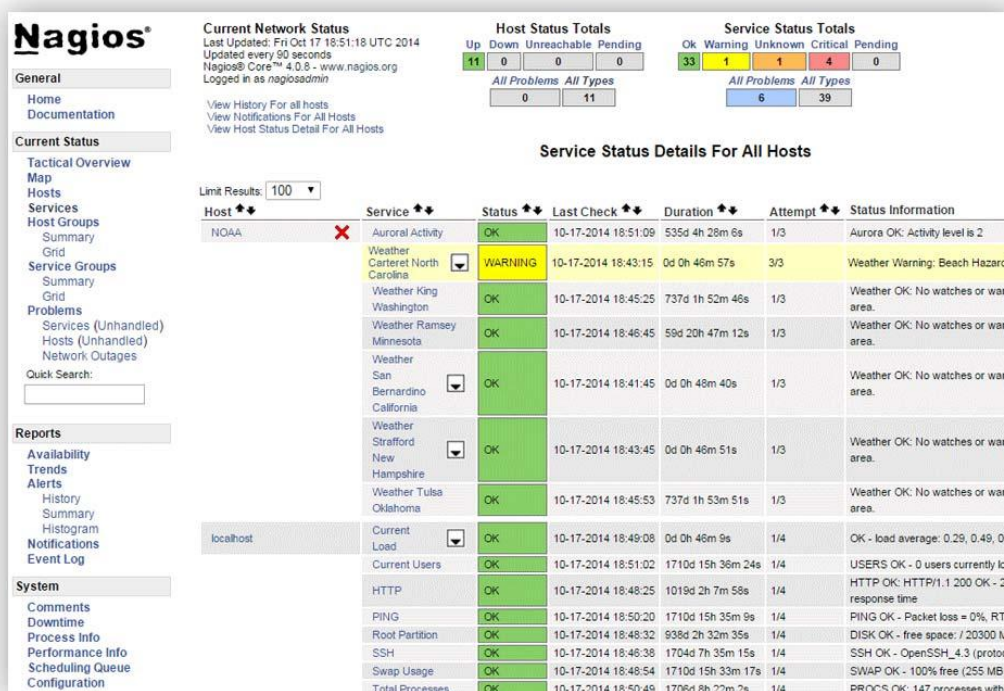


Imagen: aplicación Nagios



ENLACE DE INTERÉS

Este es el enlace a la web oficial de la aplicación, en la que se puede ampliar información sobre sus prestaciones, y realizar su descarga.

<https://www.nagios.org/>

RESUMEN FINAL

En esta unidad se ha comenzado por revisar el modo en que se gestionan los usuarios locales tanto en sistemas operativos libres como propietarios, y la importancia que dicha tarea tiene en la seguridad e integridad de los sistemas informáticos.

Seguidamente se ha visto cómo se pueden organizar los usuarios en grupos, simplificando de este modo el trabajo con estos.

Se ha hablado de los permisos de acceso a recursos, que permiten asegurar que sólo los usuarios habilitados para ello puedan acceder a los distintos objetos que se almacenan en nuestros equipos.

Se han presentado los conceptos de proceso y servicio, se ha visto su importancia dentro de un sistema operativo, y se mostrado cómo trabajar con ellos en los diferentes sistemas.

Por último, se han presentado una serie de herramientas de ayuda para la monitorización de sistemas, que nos permiten controlar el rendimiento de estos, evitando así posibles fallos futuros.