Tabla de Contenidos



#### Administrador de Tareas

1. **Automatización de Tareas del sistema**

#### Scripting en PowerShell

1. **Optimización del Arranque del Sistema Operativo**

#### Visor de Eventos

**1**

# 1. Administrador de Tareas



## Administrador de Tareas

* Herramienta que permite **controlar la actividad del sistema y su rendimiento** en tiempo real.
* Muestra información sobre el procesador, la memoria, los procesos, programas y servicios, la red y los usuarios que se encuentran conectados.
* Está compuesto por los administradores de **procesos**,

**rendimiento** del sistema, **usuarios**, **detalles** y **servicios**.

* Para ejecutarlo se puede usar una de las siguientes opciones:
  + **Ctrl+Shift+Esc.**
  + **Ctrl+Alt+Supr** y seleccionamos **Administrador de tareas.**
  + Ejecutar el programa **taksmgr.exe**.
  + Clic con el botón derecho en la barra de tareas  **Administrador 2**

**de tareas.**

# 1. Administrador de Tareas

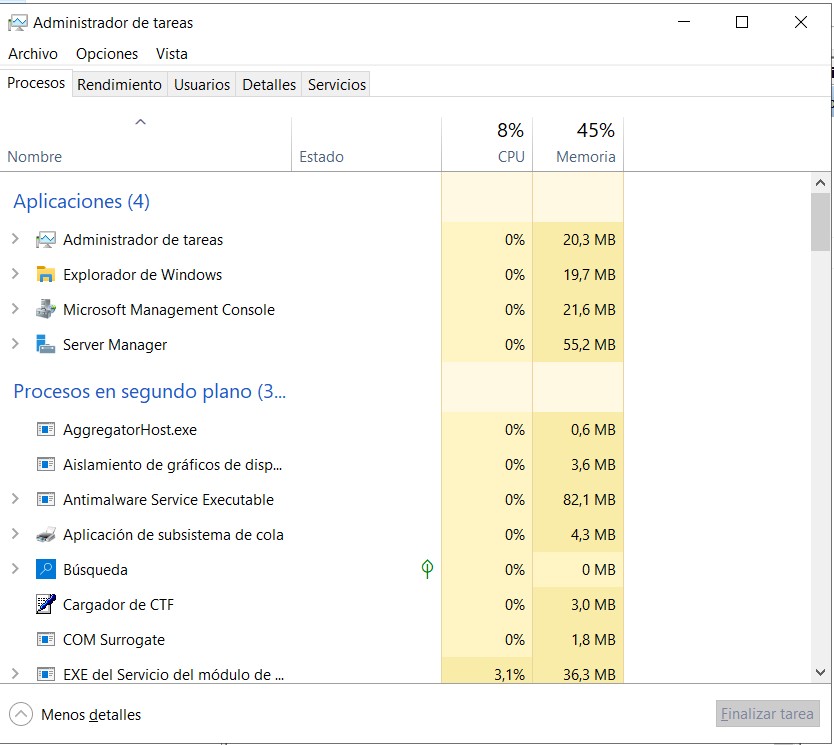


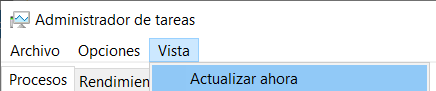
## Administrador de Tareas

#### El Administrador de Tareas permite:

* + Parar un proceso cuando se está ejecutando. Es decir, si al

intentar cerrar el programa, este no responde.

* + Ver el estado de la red.
  + Ver los usuarios conectados.
  + Los servicios.
* Para cada concepto, podemos actualizar la información mostrada en ese instante en **Vista > Actualizar ahora**

**3**

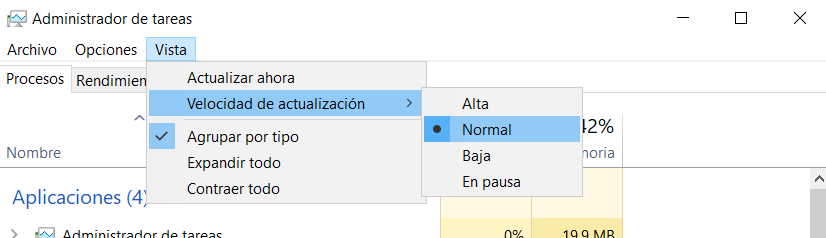
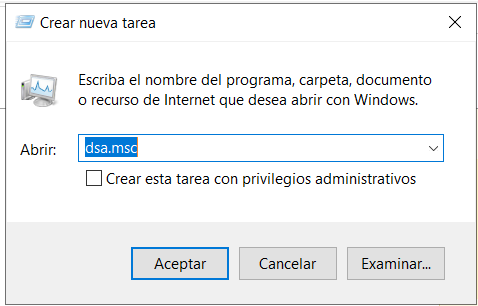
# 1. Administrador de Tareas

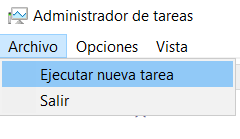


## Administrador de Tareas

* La **velocidad de actualización** se puede configurar para que

la información de la pantalla se actualice cada:

* + **Alta:** dos veces por segundo.
  + **Normal:** cada dos segundos.
  + **Baja:** cada cuatro segundos.
  + **En pausa:** no se actualiza.
* La opción **Archivo > Ejecutar nueva tarea** abre una ventana para poder ejecutar un programa y si queremos podemos marchar que se ejecute con privilegios de administrador.

**4**

# 1. Administrador de Tareas



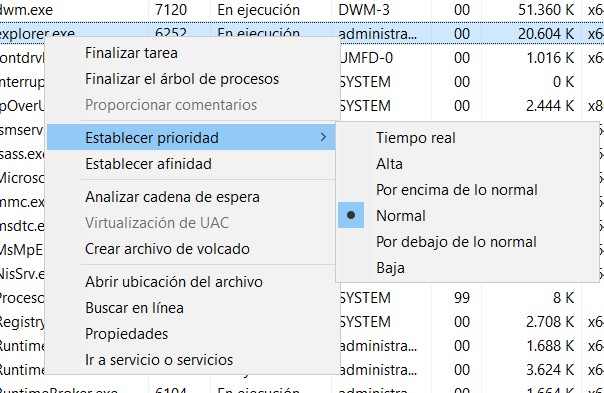
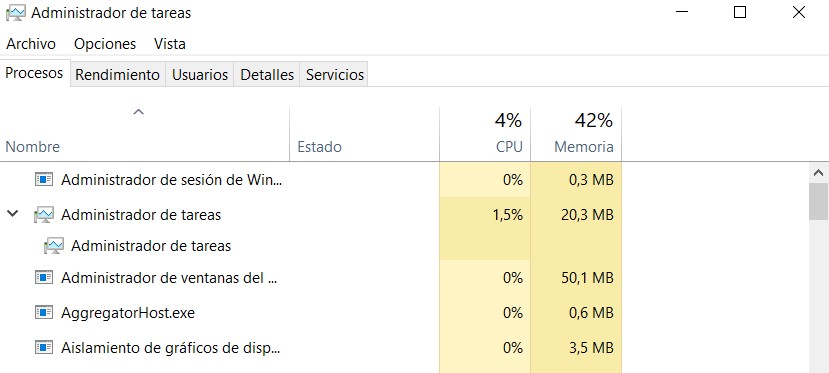
## Procesos

* Muestra información detallada de todos los **procesos en**

**ejecución**: PID, %CPU, tiempo CPU, % uso de la memoria,…

* La **prioridad** determina cuántos recursos del sistema se

asignan a un proceso.

* + Posibles **valores**: tiempo real, alta, por encima de lo normal, normal, por debajo de lo normal, baja.
* Permite finalizar un proceso, finalizar un proceso y sus “subprocesos” (finalizar el árbol de procesos) o cambiar la prioridad de un proceso.

**5**

# 1. Administrador de Tareas



## ACTIVIDADES

* Abre las siguientes actividades: Calculadora, Visor de eventos, Configuración del sistema y Directivas de seguridad local. Cambia de una a otra mediante el Administrador de tareas.
* Cierra las aplicaciones abiertas en la actividad anterior con el

Administrador de tareas.

* Ejecuta desde el Administrador de tareas las herramientas Información del sistema y Administración de discos.
* Cambia la velocidad de actualización del Administrador de tareas para que no se actualice. Abre el Bloc de notas y comprueba que no aparece como tarea. Cambia la velocidad de actualización a alta y observa cómo sale inmediatamente

como una tarea.

# 1. Administrador de Tareas

## Procesos

* Un programa es una lista de instrucciones que se almacenan en ficheros.

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamenteUn proceso** es la ejecución de un programa (ente activo), que requiere que se cargue previamente

en memoria principal, que consta tanto del programa como de los datos y los registros necesarios

para llevar a cabo la ejecución.

Un proceso es cada una de las distintas acciones que pueden estar ejecutándose simultáneamente

en un S.O.

Además, el procesador debe poder atender a varios procesos de forma simultánea, para ello

el SO se encarga de:

* + - * + Creación y eliminación de procesos.
        + Comunicación y sincronización de procesos.
        + Ejecución concurrente.
        + Planificación de procesos.
* ¿Quién crea los procesos y como se crean?

Todo sistema operativo cuando arranca lanza un proceso inicial que mantiene vivo el sistema.

Este proceso es el encargado de lanzar el resto de los procesos que sean necesarios, los cuales a su vez pueden

lanzar nuevos procesos.

A los procesos que crean a otros procesos se les denomina procesos padres, y a los creados procesos hijo. Cuando un

proceso padre es eliminado, todos sus procesos hijo son eliminados. (documento / impresión) Obviamente deben existir

restricciones en la eliminación de procesos, no pudiendo un usuario eliminar los procesos de otro usuario.

Cuando el usuario solicita la ejecución de un programa (proceso), es decir cuando un programa se convierte en proceso, el

SO realiza los siguientes pasos:

* + - * + Crea el proceso y le asigna un identificador único (PID).
        + Carga el código del proceso en memoria principal.
        + Crea el Bloque de Control de Proceso (BCP).
        + Crea el segmento de datos.
        + Crea el segmento de pila.
* El BCP es una estructura de datos que, junto con los segmentos de código, datos y pila de ejecución de un proceso, contienen toda la información de un proceso en un momento dado durante su ejecución. Existe un BCP por cada proceso en memoria y su contenido es el siguiente:
* PID.
* Prioridad del proceso.
* Estado del proceso. (listo, ejecución, bloqueado, terminado )
* CP. (Nos indica por donde va ejecutándose el programa)
* registros de la CPU.
* Identificador de usuario (UID). Identifica que usuario a lanzado ese proceso
* Tabla de distribución de páginas (TPD).
* Información de los dispositivos de E/S.
* Información de archivos asignados.
* Estadísticas:
  + Hora de inicio.
  + Tiempo de ejecución.
  + Porcentaje de uso de CPU.
  + Tiempo de espera.
* Apuntador al BCP padre.
* Apuntador a BCP hijos.
* Información del algoritmo de planificación.
* Si el proceso es interrumpido o abandona la CPU, el contenido del BCP es necesario para que el procesador pueda continuar en el mismo instante que dejo el proceso en el momento que se interrumpió.

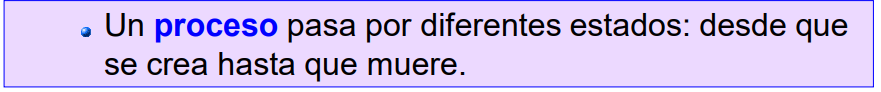
Texto

Descripción generada automáticamente

* CICLO DE VIDA DE UN PROCESO

Los procesos durante su ejecución pasan por distintos estados. En un instante dado, un proceso sólo puede estar en un estado.

El SO se encarga de controlar las transacciones de estado de un proceso. El ciclo de vida típico es el de cinco estados.



|  |  |  |
| --- | --- | --- |
| Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto  Descripción generada automáticamente | Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto  Descripción generada automáticamente | Listo ó preparado  Interfaz de usuario gráfica, Texto, Aplicación  Descripción generada automáticamente |

Diagrama

Descripción generada automáticamente

Transacciones de un proceso:

1. El SO crea un proceso **NUEVO** y le asigna un PID.
2. El SO asigna memoria principal al proceso, pero no la CPU se queda **LISTO o preparado**.
3. El SO asigna la CPU y el proceso pasa a **EJECUCIÓN**. Cuando está usando el procesador esta activo
4. La ejecución termina en el tiempo de CPU asignado y el proceso pasa a estado **TERMINADO**, donde se le libera todos los

recursos (memoria, CPU, etc.).

1. El proceso consume el Quamtiun de tiempo asignado y pasa a estado **LISTO**, el planificador elige otro proceso de la cola de

procesos listos y le asigna la CPU.

1. Se produce una operación de E/S que obliga al proceso a esperar, por lo que el SO lo pasa a estado **BLOQUEADO o en espera**, el planificador elige otro proceso de la cola de procesos listos y le asigna la CPU.
2. Cuando la operación de E/S finalice el proceso pasa a estado **LISTO**.

Interfaz de usuario gráfica, Texto, Chat o mensaje de texto

Descripción generada automáticamenteEl **planificador de procesos (DISPATCHER)**  es la parte del S.O que se encarga de controlar

el estado de los procesos e indicar que proceso es el que se está ejecutando o pasa a ejecutarse,

para lo cual utiliza un algoritmo de planificación.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

* Para indicar como es el algoritmo de planificación, medir su eficacia, deberemos tener en cuenta los siguientes puntos o apartados:
  + **COLA DE PROCESADOR**: Contiene los programas que tienen asignados todos los recursos y le falta asignarle el tiempo de procesador o ejecución.
  + **COLA DE E/S**: Contiene procesos que necesitan realizar operaciones de E/S. Cuando un proceso en ejecución pide una E/S, se comprueba si el dispositivo esta libre. Si es así pasa a realizar la operación de E/S. Pero si no está libre, el B.C.P (bloque de control de programas) solicitante quedará ligado a una cola asociada a ese dispositivo de E/S.

-La Equidad  asignar el procesador a cada proceso de forma equitativa.

-La Eficacia  mantener ocupado el procesador el 100% del tiempo.

-El Rendimiento  el número de trabajos ejecutados en unidad de tiempo ha de ser máximo. THROUGHPUT.

Tenemos algoritmos de planificación de la cola de procesos.

**.- Planificación no expulsora**.

El primer proceso que llega se pasa a ejecutar y no deja el procesador hasta que no termina su ejecución.

**.- Planificación expulsora**.

Si llega un proceso con mayor prioridad que el que está ejecutándose en el procesador, el que llega es capaz de echar,

expulsar, (al que se estaba ejecutando) del procesador y ya se terminará después de completar su ejecución.

* Cambiar la prioridad de un proceso. Es importante tener en cuenta que el propio sistema operativo

asigna recursos a un proceso tomando su prioridad como referencia: cuanto mayor sea el nivel de

prioridad, más recursos se asignarán. El nivel de prioridad normal es el predeterminado.

* Establecer afinidad de un proceso. En microprocesadores de varios núcleos o en equipos multiprocesador

es posible indicar para cada proceso en qué núcleos o microprocesador se debe ejecutar.

* Terminar un proceso. Detener un proceso y sus subprocesos. Esta acción fuerza la finalización del mismo.
* Ubicación del ejecutable. Permite conocer dónde se encuentra almacenado el ejecutable que inició el proceso.

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

Administrar procesos desde el Símbolo del sistema (CMD) es posible mediante el comando tasKlist.

Por ejemplo, si queremos mostrar información detallada sobre todos los procesos que se están

ejecutando actualmente, hay que escribir: tasklist /v /fi “status eq running”

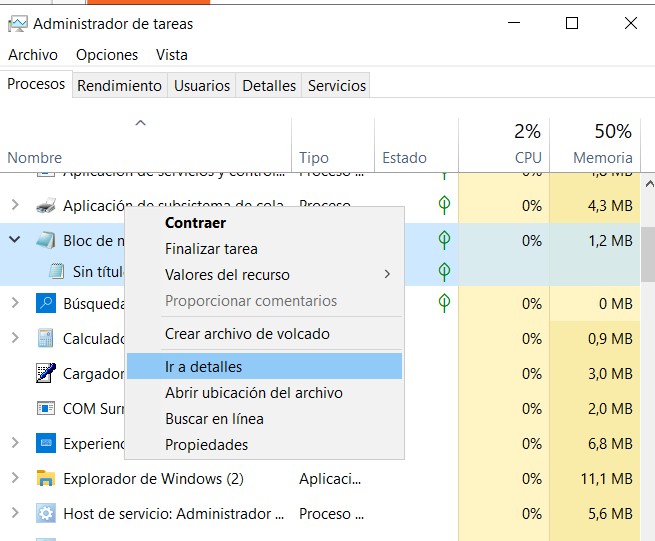
Tasklist a secas nos da la lista de procesos.

Help tasklist o tasklist /? nos da ayuda del comando.

# 1. Administrador de Tareas



## Procesos

* Un **proceso** es un programa en ejecución.
* Si nos situamos sobre cualquier tarea abierta podemos ver su **proceso asociado** eligiendo la opción **Ir a detalles** en el menú contextual que se despliega.
* Desde la pestaña **Detalles** podemos ver todos los procesos que se están ejecutando:
  + **PID**, estado, nombre del usuario al que pertenece cada proceso,

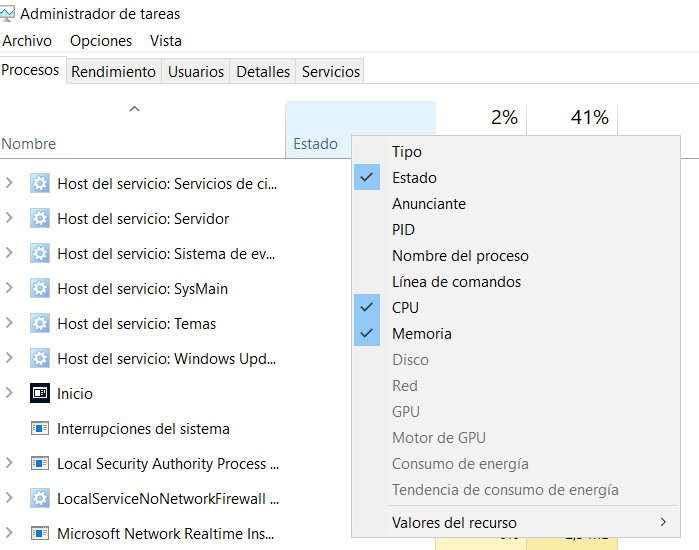
cantidad de memoria que está **9**

usando, etc.

# 1. Administrador de Tareas



## Procesos

* **Proceso inactivo del sistema**: proceso especial con PID (0), que hace referencia al porcentaje de tiempo de inactividad del procesador.
* **Finalizar tarea**: permite terminar o finalizar un proceso.
* **Ir a servicio o servicios**: permite ver si un servicio se está ejecutando mediante un proceso. Los servicios asociados al proceso aparecerán resaltados en la pestaña **Servicios**.
* Podemos elegir las columnas de información sobre los procesos que queremos que aparezcan, marcando

la casilla correspondiente.

# 1. Administrador de Tareas



Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

Administrar procesos desde el Símbolo del sistema (CMD) es posible mediante el comando tasKlist.

Por ejemplo, si queremos mostrar información detallada sobre todos los procesos que se están

ejecutando actualmente, hay que escribir: tasklist /v /fi “status eq running”

Tasklist a secas nos da la lista de procesos.

Help tasklist o tasklist /? nos da ayuda del comando.

**10**

# 1. Administrador de Tareas



## ACTIVIDADES

* Abre el Explorador de archivos de Windows. Después abre el Administrador de tareas. Comprueba el proceso al que está asociada la tarea.
* Abre el Bloc de notas y, a continuación, ciérralo matando el

proceso.

* Mira el servicio o servicios asociados a un determinado proceso.

**11**

# 1. Administrador de Tareas



## Servicios

* Un **servicio** es proceso que se ejecuta en segundo plano en el

sistema, de forma transparente, los usuarios no deberemos interactuar, pero son totalmente necesarios para el funcionamiento del sistema.

* A diferencia de los procesos, los servicios no se pueden cerrar o eliminar, lo único que podemos hacer es detenerlos o reiniciarlos, pero en ningún caso "matarlos".
* Por poner ejemplos de servicios con los que cuenta Windows y que están trabajando de forma constante en segundo plano y no somos conscientes de ello, podemos poner como ejemplo: cliente DNS, cliente DHCP, servicios de escritorio remoto etc.
* Aplicaciones que pueden estar en ejecución o no, pero son aplicaciones destinadas al funcionamiento del sistema operativo y no al manejo de un usuario, por ejemplo, cuando nuestro sistema operativo detecta una conexión de red y tenemos la red configurada en modo automático, el equipo lanzará una solicitud de asignación de IP a través de la red hasta encontrar el servidor de DHCP que se la proporcione, pues bien, toda la tarea e intercambio de información lo está realizando el servicio de DHCP de Windows, que al final no será más que una aplicación que se ejecutará en un momento dado.
* Conclusión

Podríamos decir que un proceso es una aplicación con la que podremos trabajar de forma consciente y sobre la que tendremos el control, mientras que un servicio es una aplicación con la que trabajará el propio sistema operativo y sobre la que no tenemos más control que únicamente detenerla o ponerla en ejecución, los procesos los percibimos y trabajamos con ellos a diario, mientras que los servicios están en la sombra trabajando para el correcto funcionamiento de Windows.

Proporcionan varias funcionalidades del sistema operativo, por ejemplo, servicios web, registro de eventos, servicios de archivos, impresión, criptografía, informe de errores, etc.

La ficha de servicios proporciona información detallada sobre los servicios del sistema: nombre, PID, descripción, el estado y el grupo de cada servicio.

Windows de forma predeterminada:

Inicia diversos servicios al inicio del sistema.

Otros están a la espera en modo automático y son activados por aplicaciones del sistema o por otros servicios.

Cada usuario puede decidir qué servicios desactivar completamente.

# 1. Administrador de Tareas



## Servicios

* Los **servicios proporcionan funciones clave** en las estaciones de trabajo y servidores con Windows Server:
  + Autenticación de los accesos de usuario.
  + Registros y alertas de rendimiento.
  + Instalación y configuración de dispositivos.
  + Impresión.
  + Registro de eventos que generan las aplicaciones y el sistema

operativo.

* + Servicios para las comunicaciones y conexiones de red.
  + Cumplimiento de las licencias.
  + Cliente DNS, Telnet, RPC, etc. **16**

# 1. Administrador de Tareas



## Servicios

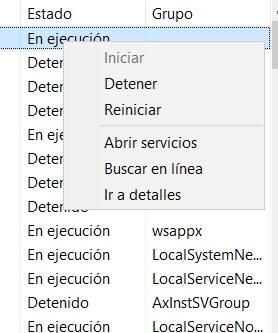
**17**

# 1. Administrador de Tareas



## Servicios

* Con el botón derecho sobre un servicio se puede:
  + **Iniciar, Detener, Reiniciar.**
  + **Ir a detalles**: va al proceso asociado en la ficha Procesos*.*
  + **Abrir servicios**: permite acceder a la consola de

servicios locales.

* Seleccionando un servicio podemos:
  + **Iniciar** un servicio que está detenido.
  + **Detener** un servicio que está en ejecución.
  + **Reiniciar**

**18**

# 1. Administrador de Tareas



## Servicios

* El **tipo de inicio de un servicio** puede ser:
  + **Manual**: se ejecuta solo cuando se solicita.
  + **Automático**: se ejecuta al iniciar el sistema operativo.
  + **Automático (inicio retrasado)**: el servicio se inicia después de que Windows se haya iniciado completamente.
  + **Deshabilitado**: no se ejecuta a no ser que se cambie el tipo de

inicio.

* + En manual y automático existe la opción de **desencadenar inicio**, y por tanto el servicio se ejecutará cuando sea necesario

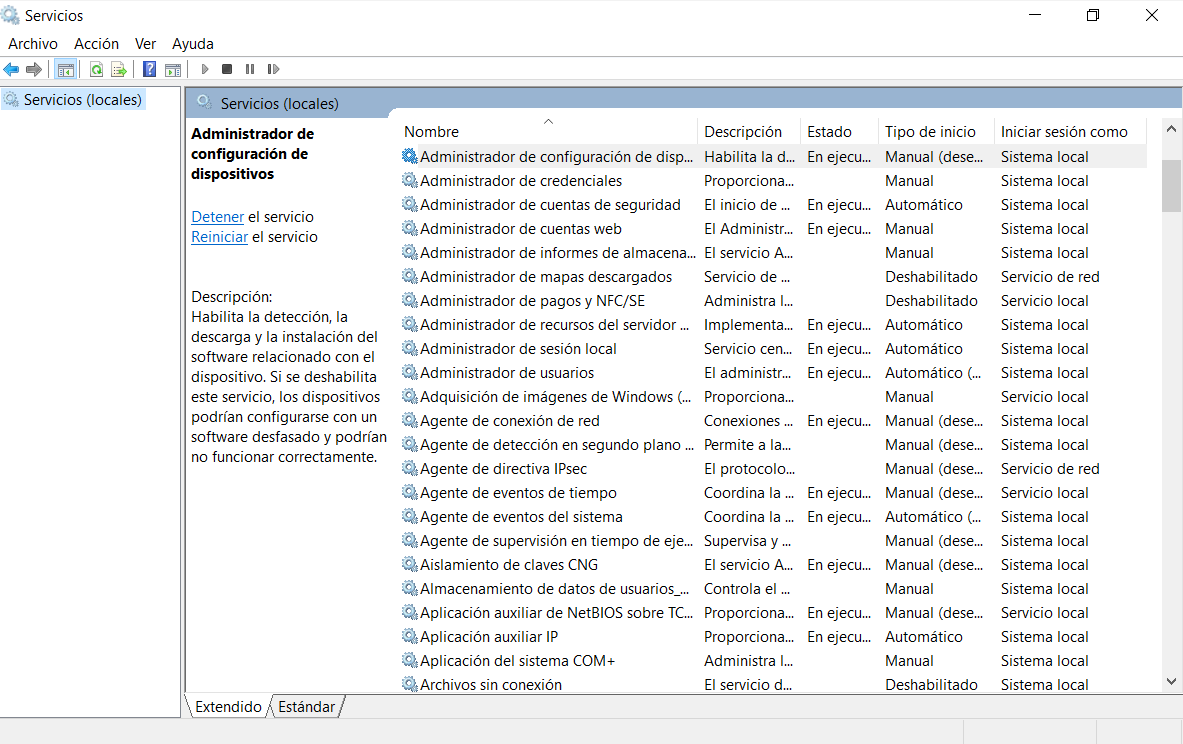
**19**

# 1. Administrador de Tareas



## Servicios: Administración

* Para **administrar los servicios instalados** desde:
  + El **administrador del servidor** > Herramientas > Servicios.
  + **Herramientas administrativas** > Servicios.
  + Ejecutar el comando

**services.msc**

* + **Administración**

**de**

**equipos**.

**20**

# 1. Administrador de Tareas



## Servicios: Administración

* Para cada servicio tenemos la siguiente información:
  + **Nombre** del servicio.
  + **Descripción** del servicio.
  + **Estado**: en ejecución, pausado o detenido.
  + **Tipo de inicio**: configuración del inicio del servicio (automático, manual o deshabilitado).
  + **Iniciar sesión como**: cuenta con la que tiene que ser lanzado ese servicio.

**21**

# 1. Administrador de Tareas



## Servicios: Administración

#### Inicio, parada y pausa de servicios

* + Sobre el servicio, clic con el botón derecho del ratón, podemos: iniciar, detener, pausa, reanudar (después de una pausa) o reiniciar.

#### Configuración del inicio del servicio.

* + Sobre el servicio, clic con el botón derecho del ratón, opción Propiedades, en la **ficha General** se puede cambiar el Tipo de inicio: automático (inicio retrasado), automático, manual o deshabilitado.

#### Configuración de inicio de sesión

* + Opción Propiedades, en la **ficha Iniciar Sesión**, seleccionar la

cuenta de usuario con la que iniciar el servicio: cuenta del

sistema local (por defecto) o bien de un usuario concreto. **22**

# 1. Administrador de Tareas



## Servicios: Administración

**23**

# 1. Administrador de Tareas



## Servicios: Administración

* Configuración del servicio de **recuperación**:
  + Los servicios se pueden configurar para que se realicen ciertas acciones cuando un servicio falla.
* Las posibles acciones a realizar ante un fallo son: no realizar ninguna acción, reiniciar el servicio, ejecutar un programa o reiniciar el equipo.
* Se pueden configurar las distintas opciones de recuperación a realizar ante el primer error, el segundo o los siguientes errores.
* La ficha **dependencias** muestra un listado de los componentes

del sistema de los que depende el servicio. **24**

# 1. Administrador de Tareas



## Servicios: Administración

**25**

# 1. Administrador de Tareas



## Rendimiento

#### Ofrece un resumen del uso de memoria, de la CPU y de la interfaz de red, mostrando gráficos y estadísticas.

* Muestra los siguientes **gráficos**:
  + **Uso de CPU**: porcentaje usado de los recursos del procesador.
  + Historial de **uso de CPU en el tiempo**.
  + **Uso de memoria**: cantidad de memoria que se usa en el sistema.
  + Historial de **uso de memoria en el tiempo**.
  + **Uso de la interfaz de red**.
* Muestra las siguientes **estadísticas**:
  + **Totales**: Identificadores de E/S en uso, subprocesos, procesos en

ejecución, tiempo en uso desde que se arrancó el sistema y

archivo de paginación (la cantidad de memoria RAM y de **26**

memoria virtual usada y disponible).

# Administrador de Tareas



## Rendimiento

* + **Memoria del núcleo**: cantidad total de memoria usada por el núcleo, cuánta está paginada y cuánta no está paginada
  + **Memoria física**: información sobre la RAM del sistema, disponible y usada como caché.
* Información sobre las **funciones de red**:
  + **Rendimiento de la red**, mostrando el estado de las redes que están funcionando en el sistema.
  + **Gráfico por cada tarjeta de red del sistema**, con el historial del

mismo (bytes totales transferidos, bytes enviados y recibidos)

* + Tabla descriptiva del dispositivo (velocidad, porcentaje de uso,

estado, etc.). **27**

# 1. Administrador de Tareas



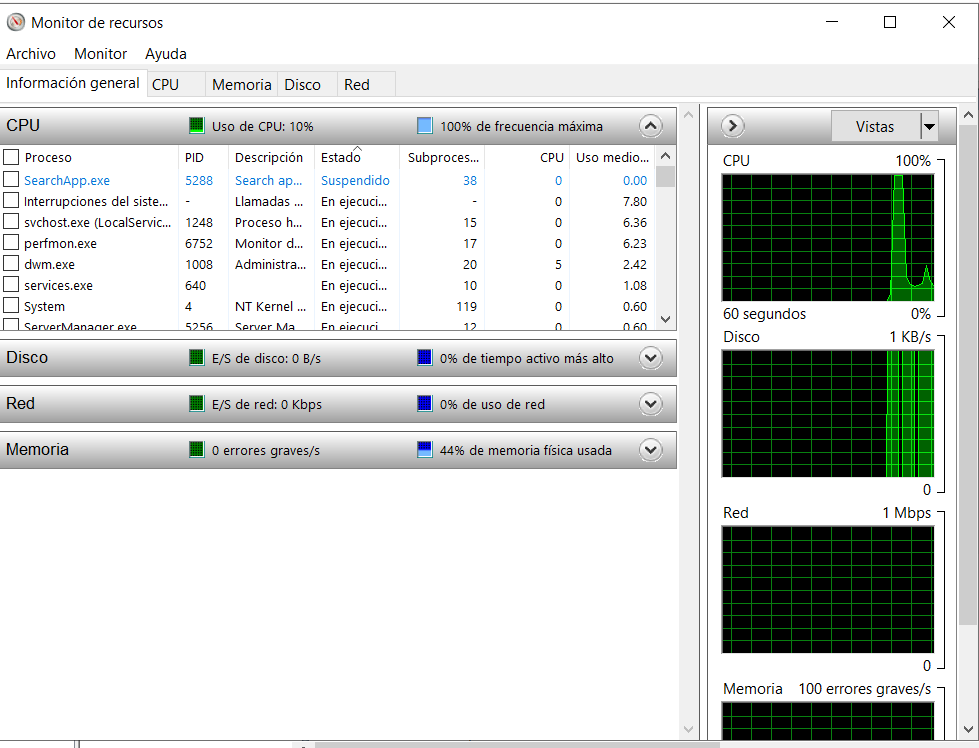
## Rendimiento

**28**

# 1. Administrador de Tareas



## Rendimiento

* El **Monitor de recursos** nos permite supervisar gráficamente la actividad del sistema con respecto al uso de la **CPU**, del **disco**, de la **red** y de la **memoria**.

**29**

# 1. Administrador de Tareas



## Rendimiento

**ACTIVIDAD**

* Desactiva los adaptadores de red que tengas instalados y mira en el Monitor de recursos su funcionamiento. Vuelve a activar los adaptadores.

**30**

# 1. Administrador de Tareas

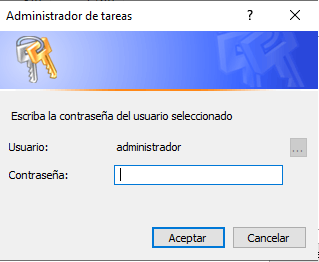
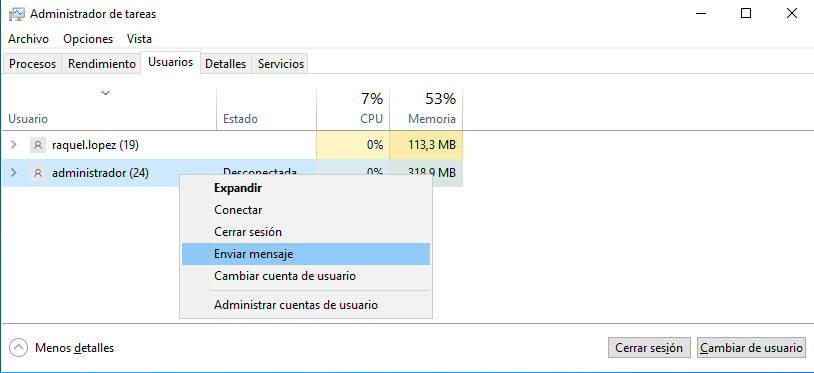


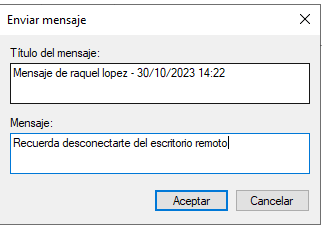
## Usuarios

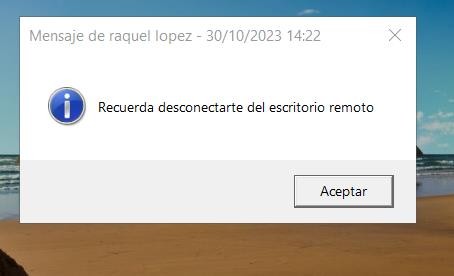
* Muestra los **usuarios conectados** al sistema, su identificador y

nombre de usuario, su estado y el tipo de sesión.

* Podemos conectar, cerrar sesión, enviar mensaje, cambiar cuenta de usuario y administrar cuentas de usuario.





**32**

# 1. Administrador de Tareas



## Comandos para Gestionar Procesos y Servicios

* En los cmdlets **Process** se asocia con la **gestión de procesos**. Se pueden realizar una serie de acciones sobre ellos utilizando estos verbos: **Get**, **Start**, **Stop**, **Wait**, **Debug**.
  + **Get-Process**: obtiene los procesos que se están ejecutando.
    - **Alias: gps, ps**
  + **Stop-Process**: detiene uno o varios procesos que estén en

ejecución.

* + - **Alias: spps,kill**
  + **Start-Process**: inicia uno o más procesos.
    - **Alias: saps, start**
  + **Wait-Process**: espera a que uno o más procesos terminen.
  + **Debug-Process**: adjunta un depurador a uno o varios procesos. **33**

# 1. Administrador de Tareas



## Comandos para Gestionar Procesos y Servicios

* En los cmdlets **Service** se asocia con la **gestión de servicios**. Se pueden realizar una serie de acciones sobre ellos utilizando estos verbos: **Get, Set, New, Restart, Resume, Start, Stop, Suspend, remove.**
  + **Get-Service**: obtiene los servicios.
    - **Alias: gsv**
  + **Stop-Service**: detiene un servicio.
    - **Alias: spsv**
  + **Restart-Service**: reinicia un servicio.
* **Tasklist**: muestra todas las tareas en ejecución, incluidos los servicios, que se ejecutan en un equipo local o remoto.
* **Taskkill**: termina o interrumpe un proceso o aplicación que se **34**

está ejecutando.

# Automatización de Tareas del Sistema



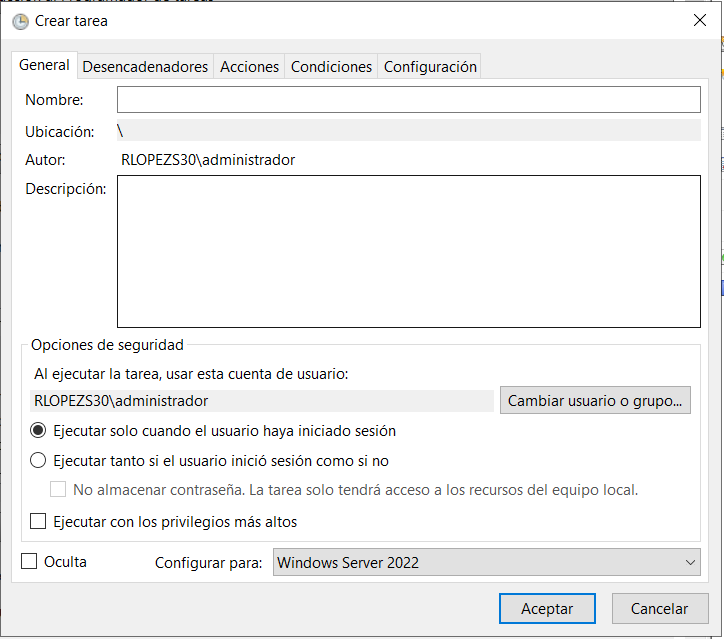
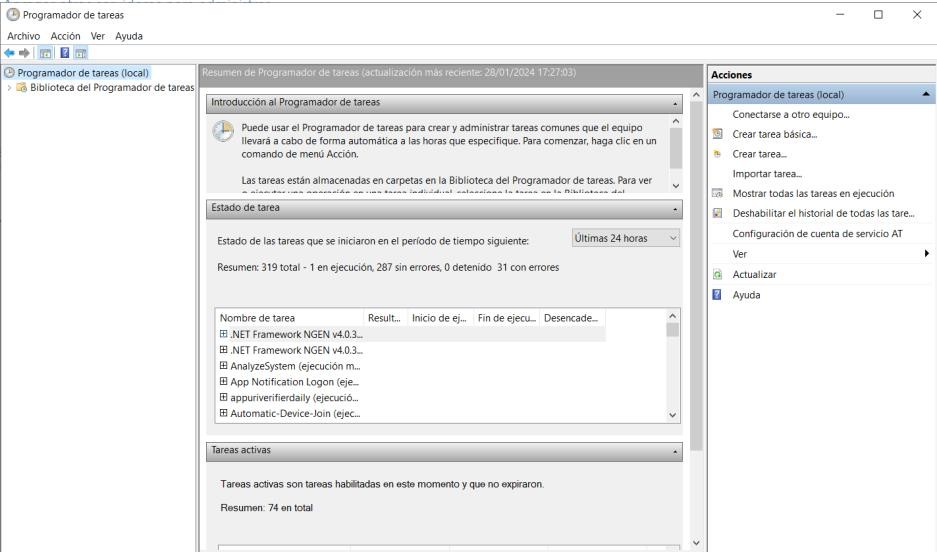
## Programación de Tareas

* Mediante la programación de tareas se pueden **automatizar tareas que realizan acciones a una hora concreta, con cierta periodicidad o cuando se produce un evento determinado**.
* El desencadenante de la tarea puede ser:
  + La hora.
  + La producción de un evento.
* El desencadenante puede estar condicionado:
* Eventos que pueden hacer que se ejecute una acción:
  + Iniciar un programa.
  + Enviar un mensaje por correo.
  + Mostrar un mensaje.
* Se pueden crear dos **tipos de tareas**: tarea básica y tarea. **38**

# 2. Automatización de Tareas del Sistema



## Programación de Tareas

* Para **programar tareas**:
  + **Administrador del Servidor**  **Herramientas**  **Programador de tareas**.
  + Ejecutando el comando **taskschd.msc**
  + Dentro de **Administración de equipos**.

**39**

# Scripting en PowerShell



## PowerShell ISE

* Para realizar tareas de administración que haya que programar o que sean un conjunto de tareas podemos escribir ficheros con los comandos para realizarlas y después ejecutar dichos ficheros.
* PowerShell dispone del **entorno de ejecución PowerShell ISE** que permite escribir y ejecutar comandos y scripts, proporcionando utilidades que facilitan la escritura y la depuración del código.
* Para poder ejecutar scripts de PowerShell es necesario tener activadas las **directivas de ejecución de scripts**.
* Para ver el **estado** de la política de ejecución de scripts:
  + **Get-ExecutionPolicy 45**



# 3. Scripting en PowerShell



## PowerShell ISE

* Los **valores posibles** son:
  + **Restricted**: no permite ningún script.
  + **Allsigned**: permite ejecutar los scripts firmados por un editor de confianza.
  + **Remotesigned**: permite ejecutar los firmados locales y remotos.
  + **Unrestricted**: permite ejecutar cualquier script.
  + **Bypass**: permite la ejecución de cualquier script.
  + **Undefined**: ninguna restricción establecida.
* Para **modificar** el tipo de política establecida:
  + **Set-ExecutionPolicy**

**46**



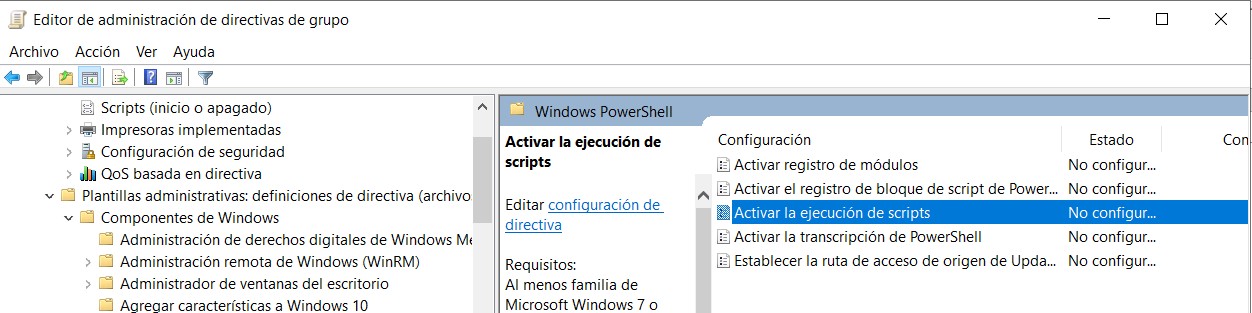
# 3. Scripting en PowerShell



## PowerShell ISE

* Es posible utilizar una **directiva de Grupo para administrar la**

#### directiva de ejecución de scripts.

* La configuración de la directiva de grupo invalida las directivas de ejecución establecidas en PowerShell.
* Las directivas establecidas en el nodo de Configuración del equipo tienen prioridad sobre las directivas establecidas en el nodo Configuración de usuario.
  + Se configura desde: **Plantillas administrativas\Componentes de Windows\Windows PowerShell**

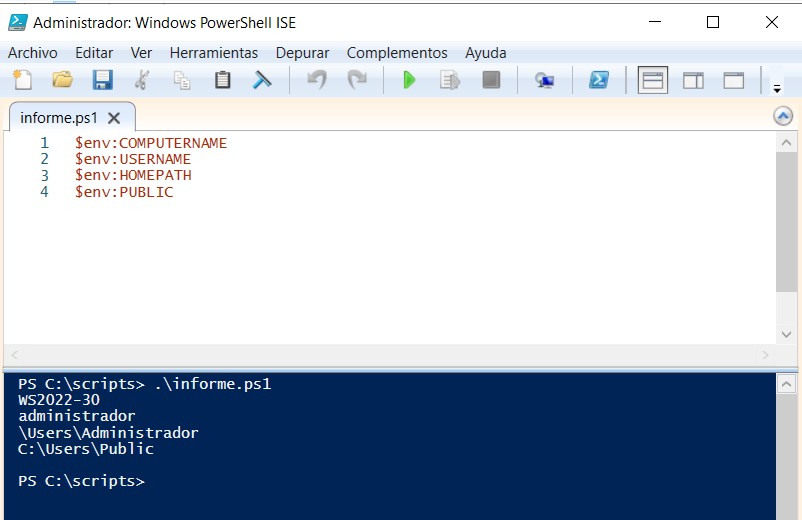
**47**

# 3. Scripting en PowerShell

## PowerShell ISE

* Algunos cmd-lets y elementos del lenguaje que se usan para

los scripts son:

* + **Write-Host:** escribe cadenas o valores en la pantalla.
  + **Read-Host**: pide un dato a través del teclado.
  + **Variables**
  + **Objetos**
  + **Alias**
  + **Funciones**
  + **Instrucciones**

**repetitivas,**

**condicionales,… 48**

# Scripting en PowerShell



## PowerShell ISE

### ACTIVIDAD

#### ¿Qué hace el siguiente script de PowerShell?

#### Interfaz de usuario gráfica, Texto Descripción generada automáticamente

**49**

# Optimización del Arranque



## Introducción

* Al arrancar el sistema operativo se inician una serie de

servicios y además se ejecutan una serie de aplicaciones.

* En la memoria del equipo se cargan programas residentes que realizan funciones necesarias, y consumen recursos del sistema.
* Para que el inicio sea óptimo es conveniente no iniciar

servicios que no sean necesarios.

* La herramienta **Configuración del sistema** (**msconfig.exe**)

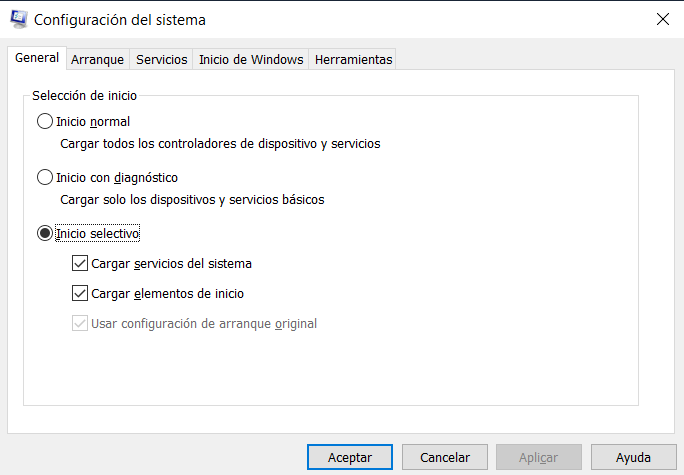
permite:

* + **Configurar el inicio del sistema operativo.**
  + **Personalizar el arranque 51**

# 4. Optimización del Arranque



## Introducción: General

* En la pestaña General podemos elegir:
  + **Inicio normal**: carga todos los controladores de dispositivos y servicios.
  + **Inicio con diagnóstico**: carga únicamente los dispositivos y servicios básicos.
  + **Inicio selectivo**: permite

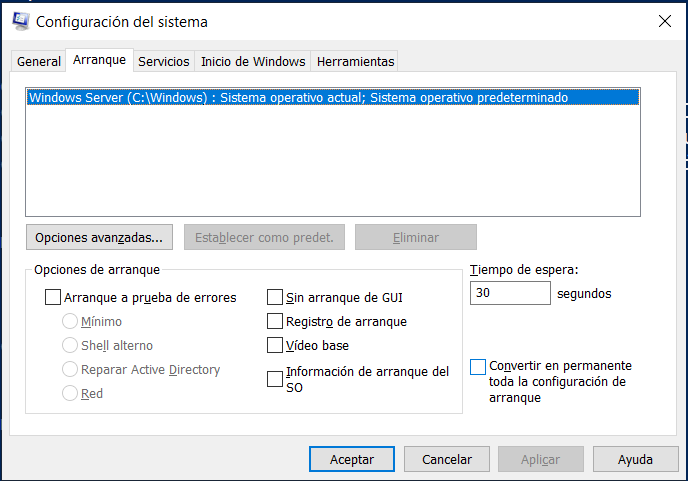
elegir que se carga.

**52**

# 4. Optimización del Arranque



## Introducción: Arranque

* En la pestaña Arranque permite configurar el modo en que se inicia Windows. Nos permite elegir qué sistema operativo queremos que arranque, en caso de tener instalado más de uno, y distintas opciones de arranque.
  + Arranque a prueba de

errores:

* + Tiempo de espera.
  + Convertir en permanente toda la configuración de arranque.

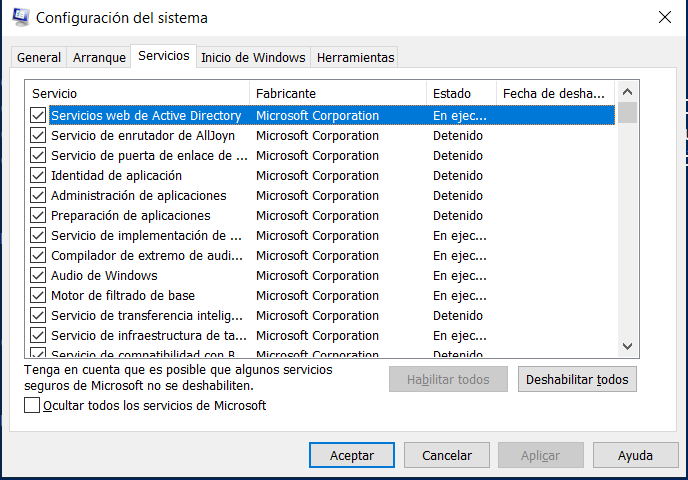
**5**

# 4. Optimización del Arranque



## Introducción: Servicios

* Muestra los servicios que se cargan al inicio del sistema y que

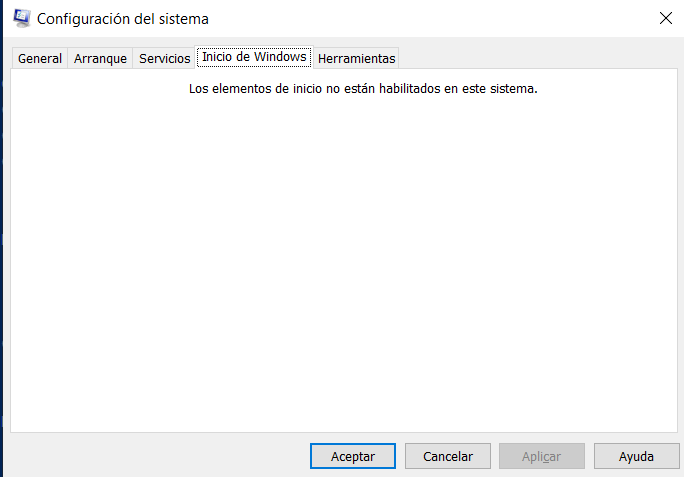
podemos evitar que se carguen desmarcando su casilla.

**56**

# 4. Optimización del Arranque



## Introducción: Inicio de Windows

* Muestra los programas que se cargan al arrancar el sistema operativo, si los elementos de inicio están habilitados en el sistema.
* Si no queremos que se cargue al inicio algún programa concreto, desmarcamos la casilla del programa correspondiente.

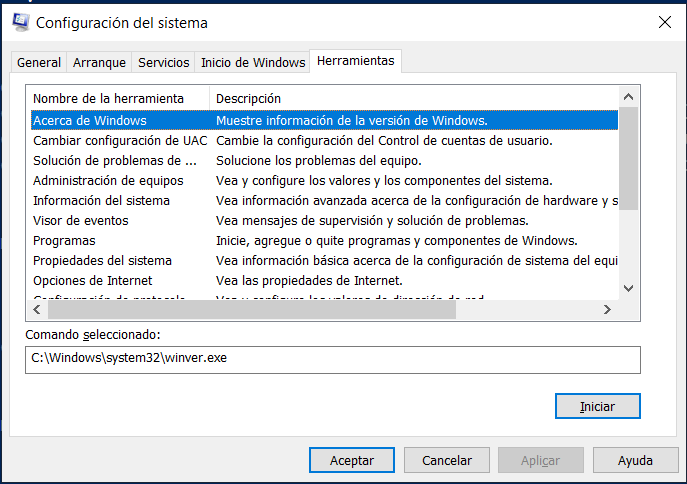
**57**

# 4. Optimización del Arranque



## Introducción: Herramientas

* Muestra herramientas avanzadas y de diagnóstico que se

pueden ejecutar desde la pestaña Herramientas.

**58**

# 4. Optimización del Arranque



## Memoria Virtual

* Windows Server usa **ficheros de paginación** como espacio de

paginación.

* De forma automática se crea un fichero de paginación en la unidad que contiene el SO.
* Para el resto de las unidades se pueden crear manualmente.
* Microsoft recomienda crear un archivo de paginación por cada volumen físico del sistema (salvo en unidades extraíbles).
* Los ficheros de paginación se crean en el volumen con el

nombre: **pagefile.sys**

**59**

# 4. Optimización del Arranque



## Memoria Virtual

* Desde Sistema > Acerca de > Configuración avanzada del sistema > Opciones avanzadas (Rendimiento) > Configuración…> Opciones avanzadas > **Memoria Virtual**: se configuran los detalles de la memoria virtual:
  + Incluye la opción **Administrar automáticamente el tamaño del archivo de paginación para todas las unidades** (Windows realiza toda la gestión, bloqueando las opciones siguientes).
  + Para cada volumen disponible muestra si hay o no fichero de paginación creado y sus características (en caso de que lo haya).
  + Se puede crear un nuevo fichero de paginación.
  + Fijar un tamaño inicial y otro máximo para cada uno de los ficheros

de paginación existentes.

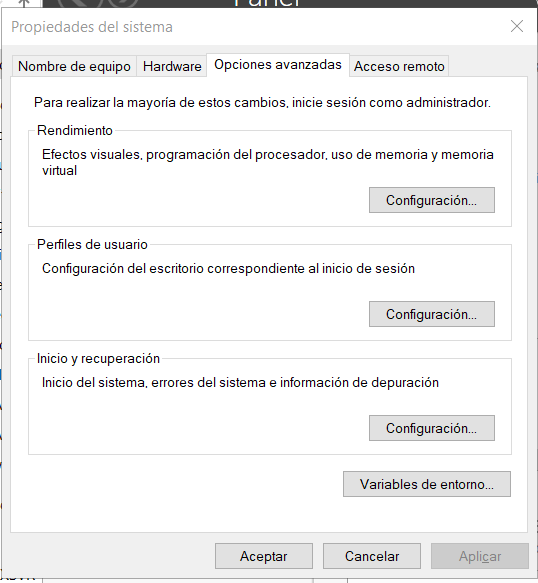
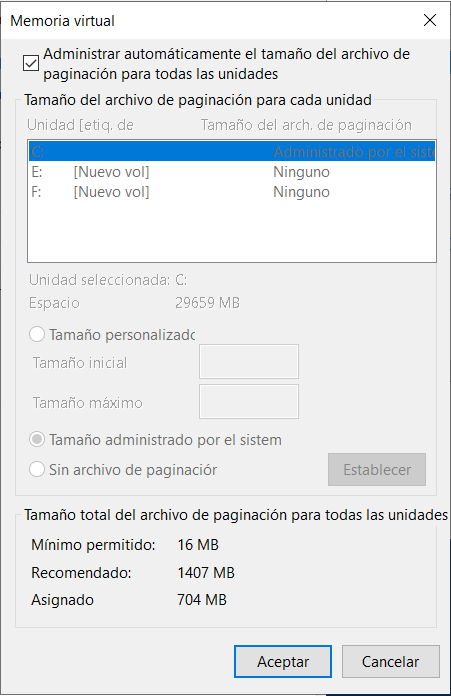
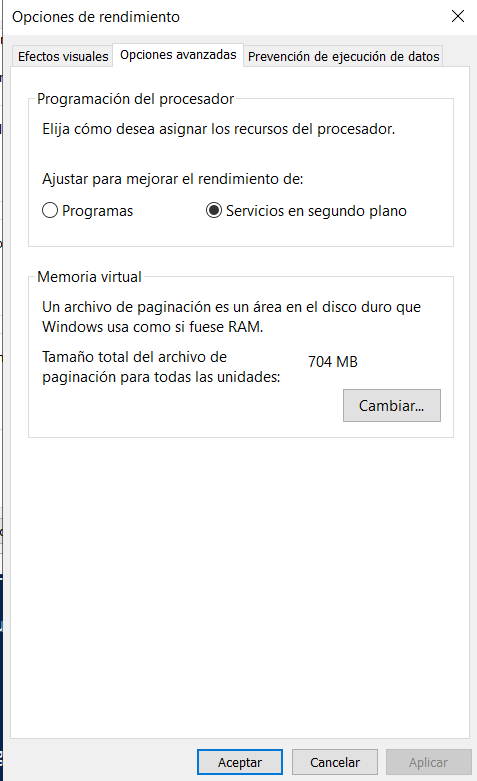
* + Informa del tamaño total del archivo de paginación para todas las **60**

unidades.

# Optimización del Arranque



## Memoria Virtual

  **61**

# Visor de Eventos



## Introducción

* **Un evento es cualquier ocurrencia significativa en el sistema operativo o en una aplicación que requiera ser notificada** a los usuarios (particularmente al administrador).
* Windows Server permite recopilar todos estos eventos en un conjunto de archivos de registro.
* Durante el funcionamiento de un equipo ocurren gran

cantidad de eventos.

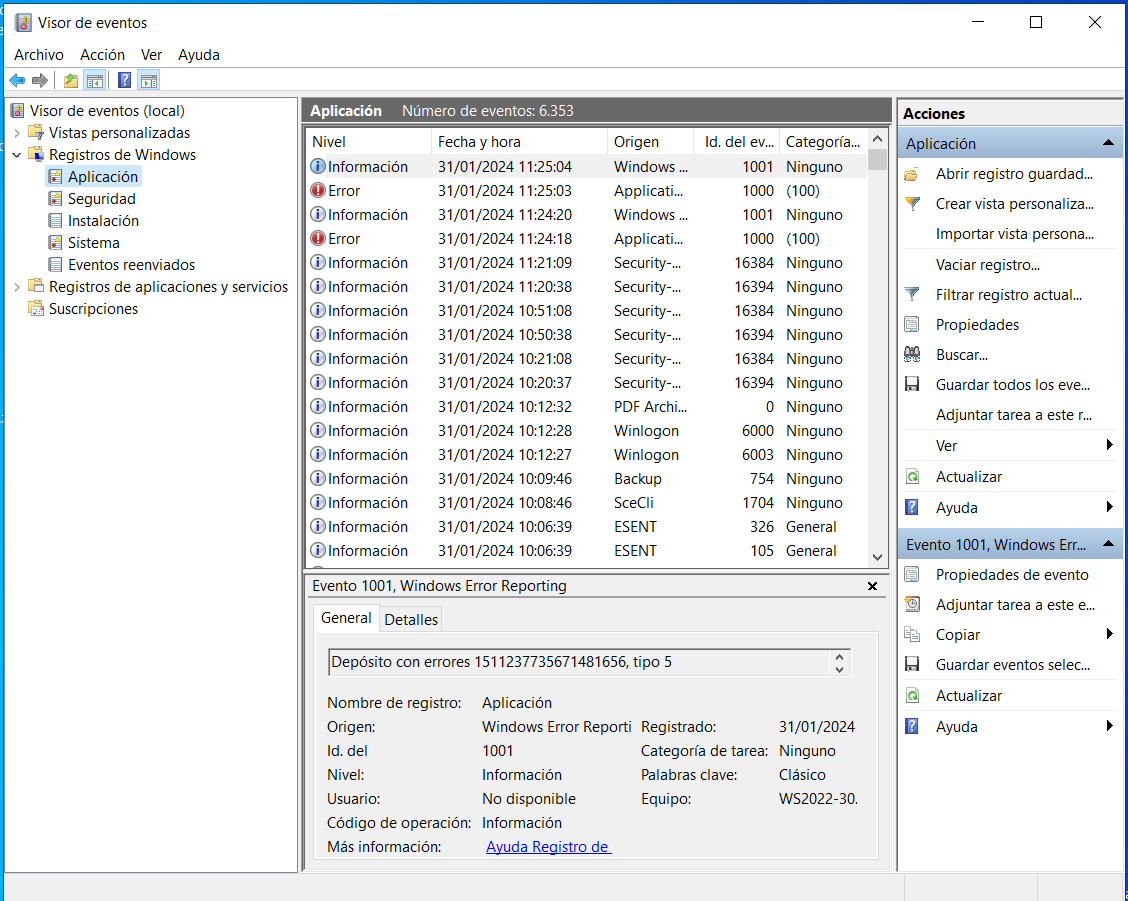
#### El visor de eventos es un complemento de *Microsoft Management Console (mmc)* que permite examinar y administrar de forma centralizada la información contenida en los múltiples registros de eventos (logs) de las

**aplicaciones y servicios de Windows**. **62**

# 5. Visor de Eventos



## Introducción

* Es una herramienta indispensable para supervisar el mantenimiento de los sistemas y solucionar los problemas que surjan.

**63**

# 5. Visor de Eventos



## Introducción

* **El visor de eventos permite** realizar las siguientes tareas:
  + **Ver eventos desde varios registros de eventos**.
  + **Guardar filtros** de eventos útiles como **vistas personalizadas** que se pueden volver a usar.
  + **Programar una tarea** para que se ejecute **como respuesta a un**

**evento**.

* + Crear y administrar **suscripciones a eventos**: mediante el servicio **Recopilador de eventos de Windows** se pueden recopilar eventos de equipos remotos en una red.
* El **registro de eventos** proporciona un histórico que permite

detectar posibles problemas de seguridad y del sistema. **64**

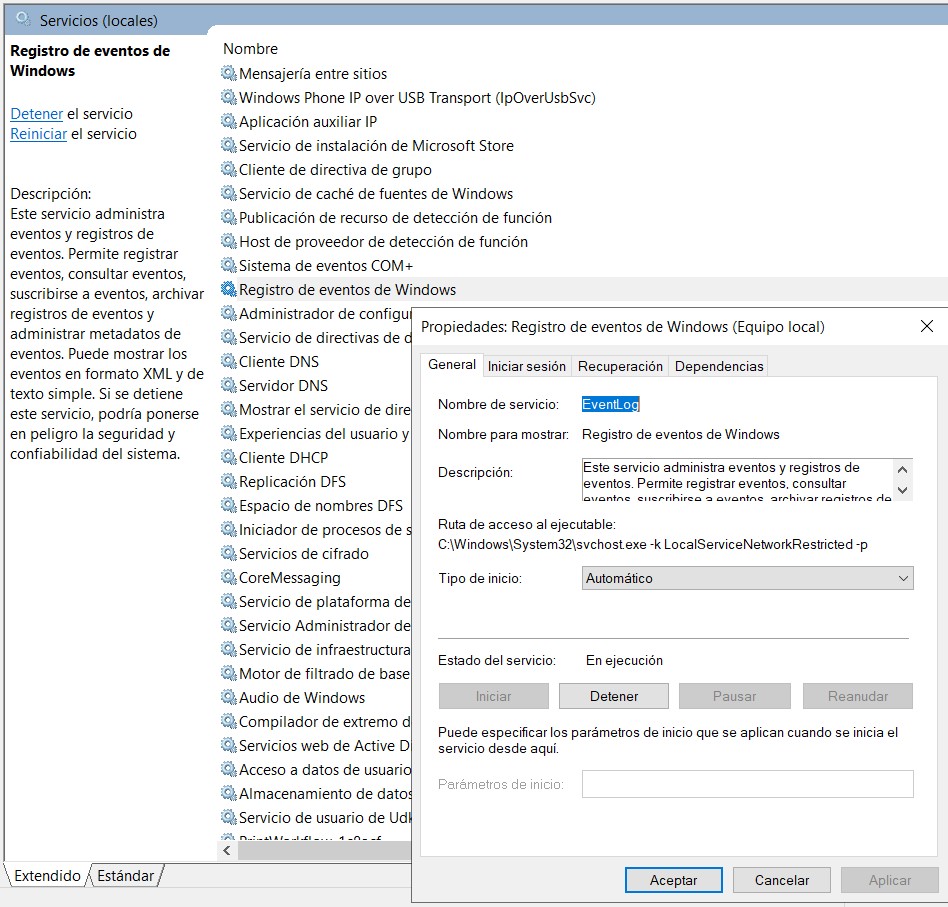
# 5. Visor de Eventos



## Introducción

#### El servicio: Registro de eventos de Windows (EventLog)

determina si los eventos del sistema se han de controlar.

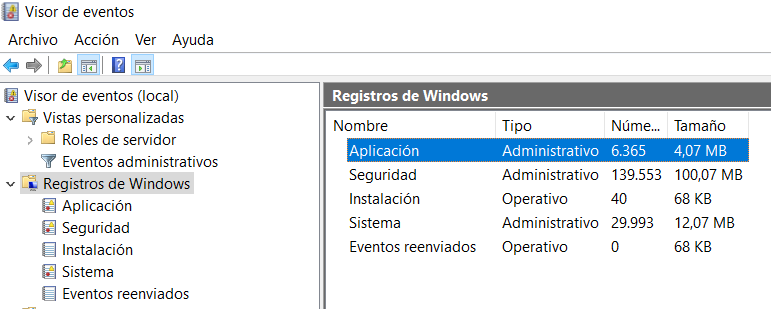
* + Si este servicio está iniciado, los eventos del sistema se capturarán para poder visualizarlos a posteriori.
* Se pueden supervisar tanto los eventos de uso de los recursos del sistema como las acciones de los usuarios.

**65**

# 5. Visor de Eventos



## Registros de Windows

* **Registro de Aplicación**: registros provenientes de las aplicaciones o **programas instaladas**.
* **Registro de Seguridad**: registros provenientes de un fallo o problema de la seguridad, como intentos de **inicio de sesión** válidos y no válidos, además de registros relacionados con el **uso de recursos**, como la **creación, apertura o eliminación de archivos** u otros objetos.
  + **Han de estar configurados para su auditoría** con

las directivas de **66**

grupo.

# 5. Visor de Eventos



## Registros de Windows

* **Registro del Sistema**: registros que genera el sistema operativo y sus componentes, como un **fallo de inicio de servicio**.
* **Registro de Instalación**: **r**egistros **durante el proceso de instalación** del sistema operativo o de sus componentes.
* **Registro de Eventos reenviados**: si el reenvío de eventos está funcionando, este archivo guarda los eventos reenviados a otros servidores.

**67**

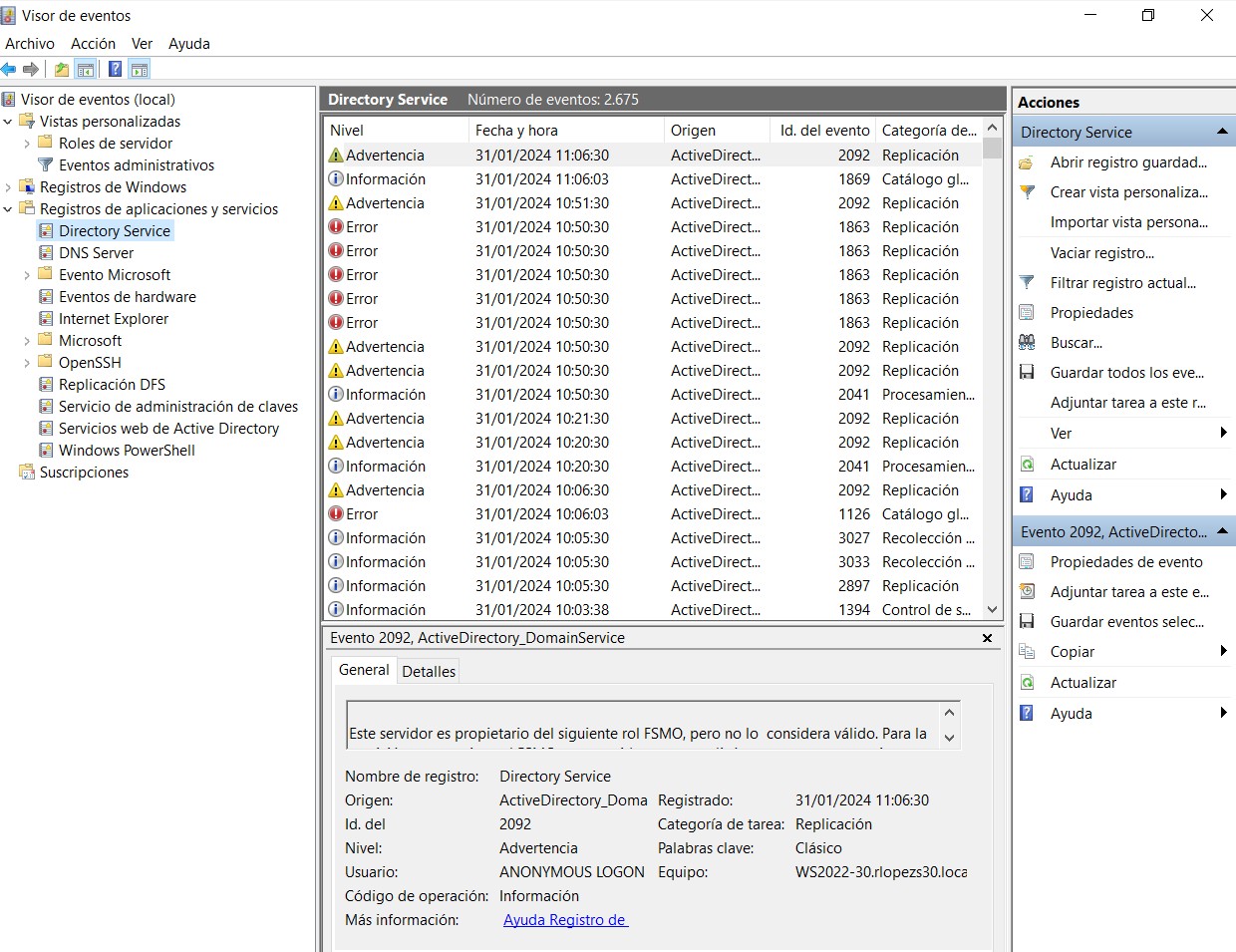
# 5. Visor de Eventos



## Registros de Aplicaciones y Servicios

* Estos registros almacenan eventos de una única aplicación o un

componente en concreto..

* + Internet Explorer, Servidor DNS, Windows PowerShell, Eventos de Hardware, Servicio de Directorio (específico de un

controlador de dominio)

.

**68**

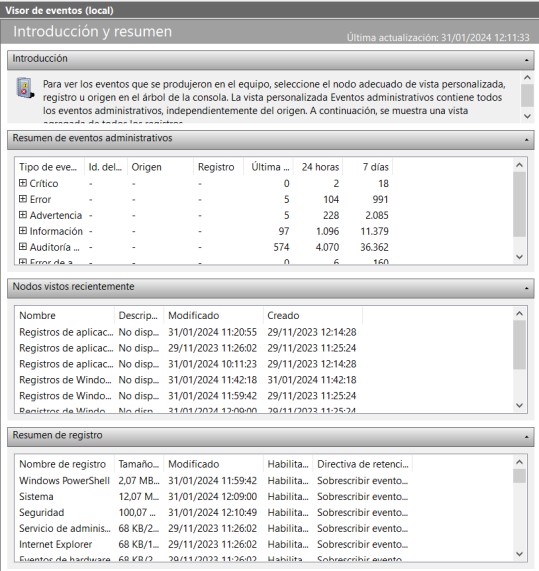
# 5. Visor de Eventos



## Acceso y Uso del Visor de Eventos

* Para lanzarlo hay dos posibilidades:
  + Desde **Administración de equipos** o **Administración del Servidor***.*
  + Windows+R  **eventvwr.exe** o **eventvwr.msc**
* El Visor de eventos proporciona un **resumen** rápido de los

mismos, indicando dónde y cuándo ocurrieron.

* Se puede obtener información más detallada haciendo doble clic sobre el evento.

**69**

# 5. Visor de Eventos



## Tipos de Eventos

* + **Información**: describe el funcionamiento correcto de una aplicación o controlador. (por ejemplo, cuando el controlador de red se carga correctamente generará un evento de este tipo).
  + **Auditoría correcta**: registra un intento de acceso de seguridad auditado correcto. (por ejemplo, un intento satisfactorio de un usuario de iniciar una sesión en el sistema).
  + **Auditoría de errores**: un intento de acceso de seguridad auditado erróneo. (por ejemplo, si un usuario intenta tener acceso a una unidad de red y se produce un error).
  + **Advertencia**: evento que a priori no es importante, pero que

es útil para la prevención de problemas futuros. (por ejemplo, **70**

queda poco espacio en el disco duro).

# 5. Visor de Eventos



## Tipos de Eventos

* **Error**: indica un problema significativo, como la pérdida de datos o de funcionalidad (por ejemplo, fallo en el inicio de un servicio).

## Información sobre los Eventos

* + **Nivel**: tipo de evento.

#### Fecha y hora.

* + **Origen**: La aplicación, servicio o

componente que lo generó.

**71**

# 5. Visor de Eventos



## Información sobre los Eventos

* + **Categoría de la tarea**: clasificación del evento por su origen, que se usa a veces para describir aún más la acción relacionada (principalmente en el registro de seguridad).
  + **Id. del Evento**: identificador del evento concreto que ha ocurrido (es un número).
  + **Usuario**: el que estaba conectado cuando ocurrió el evento.
  + **Equipo**: nombre del equipo donde ocurrió el evento.
  + **Descripción**: descripción textual del evento. (Para consultar este campo hay que pedir más información sobre el evento, haciendo doble clic sobre el mismo).
  + **Datos**: cualquier dato o código de error de salida del evento. (De nuevo se consulta al pedir más información del evento,

haciendo doble clic sobre el mismo). **72**

# 5. Visor de Eventos



## Configuración

* Cada uno de los **registros de eventos** (de aplicación, de seguridad, etc.) se almacenan en un fichero, con la extensión

**\*.evtx**, en el subdirectorio

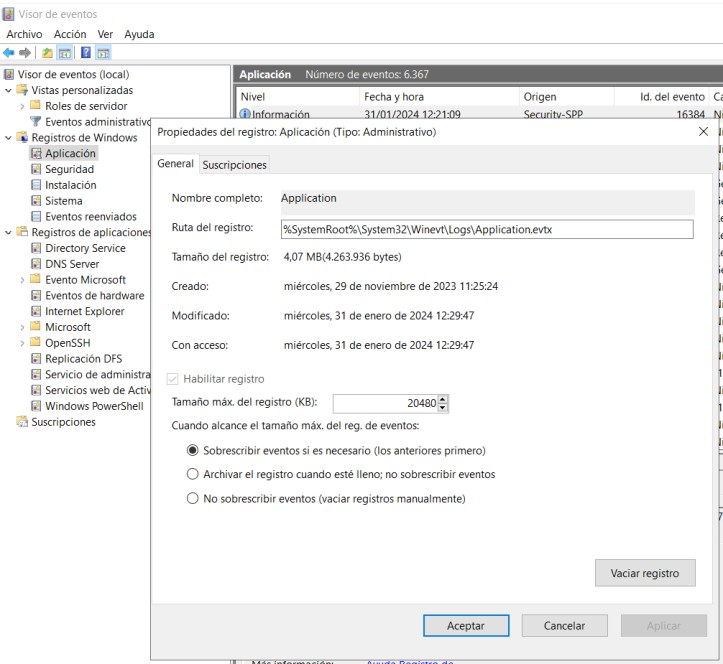
*%SystemRoot%\system32\Winevt\Logs\.*

* Por defecto, tienen un tamaño prefijado a 20480 KB. Este tamaño se puede modificar.
* Cuando el tamaño de un registro llega a ese límite: los eventos anteriores se sobrescriben para evitar que el fichero exceda el tamaño máximo indicado. (Esta opción también se puede configurar).
* Clic con el **botón derecho en el registro de eventos a configurar, seleccione la opción Propiedades***.* **73**

# 5. Visor de Eventos



## Configuración

* + En la ventana **Propiedades del registro** se puede indicar:
    - El **tamaño máximo del registro** (se puede cambiar).
    - **Qué hacer cuando se llena** el registro:
      * Sobrescribir eventos cuando sea necesario.
      * Archivar el registro cuando esté

lleno, no sobrescribir eventos.

* + - * No sobrescribir eventos (vaciar registros manualmente).

**74**

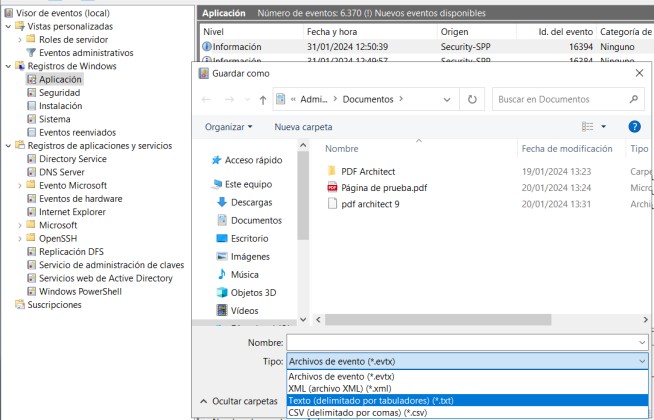
# 5. Visor de Eventos



## Otras Operaciones con los Registros de Eventos

* + Borrado de un registro de evento, eliminando todas las

entradas que lo componen.

* + Guardar o exportar los registros de eventos, realizando copias de seguridad para llevar un control más exhaustivo. Se puede hacer en los siguientes formatos:
    - **Formato Archivos de eventos**, para consultarlo será necesario acceder con el Visor de Eventos.
    - **Formato texto delimitado por tabuladores o comas**, que podrá ser editado por aplicaciones tipo

**Microsoft Word** o **Microsoft Excel**. **75**

* + - **Formato XML**.

# 

