Administración de Sistemas en la Cloud

Administración básica de sistema operativo en Windows

Índice

Esquema

ADMINISTRACIÓN BÁSICA DE SISTEMA OPERATIVO EN WINDOWS

Directivas de grupo

- Personalizan configuraciones de Windows
- Se agrupan en GPO
- Se aplican a nivel local, de sitio, de dominio y de OU

PowerShell remota

```
Get-Service schedule -ComputerName server1
Enter-PSSession -ComputerName server1
```

Herramientas del sistema

- Visor de eventos
- Administrador de tareas
- Programador de tareas
- *Suite* de herramientas Sysinternals

Ideas clave

9.1. Introducción y objetivos

Este tema profundiza en algunas de las herramientas necesarias para la administración de equipos Windows. Las directivas de grupo son la opción más extendida para aplicar configuraciones en grandes flotas de servidores y equipos de escritorio. En este se muestra en detalle cómo usar PowerShell remotamente. Para finalizar, se citan unas cuantas herramientas, que son la navaja suiza de cualquier administrador de equipos Windows.

Los comandos y sintaxis de PowerShell tienen poco valor, hasta que se ponen en un contexto más realista. Es necesario familiarizarse con ellos, pero no se aprecia su potencia hasta que no se integran en una tarea compleja.

Los **objetivos** que se pretenden conseguir son:

- Entender el concepto de directiva de grupo y su uso en Directorio Activo.
- Aprender el uso de PowerShell como herramienta de administración remota.
- Tomar un primer contacto con varias de las herramientas habituales de administración en Windows.

A continuación, en el vídeo *Administración de usuarios y procesos en Windows*, se explicará brevemente la consola de administración de directivas de grupo y algunas de las opciones disponibles para administrar servicios y procesos.

Accede al vídeo

9.2. Directivas de grupos

Una de las fortalezas de los sistemas operativos basados en Windows es su **flexibilidad.** Sin embargo, esta flexibilidad tiene un precio: en general, los usuarios sin privilegios administrativos de una red no deberían ser capaces de cambiar muchas de las configuraciones del sistema, como la configuración de TCP/IP y las políticas de seguridad de contraseñas.

Las **directivas de grupo,** o *group policies*, también denominadas GPO (*group policy object*), están diseñadas para proporcionar a los administradores del sistema la capacidad de personalizar la configuración del usuario final y establecer restricciones sobre los tipos de acciones que los usuarios pueden realizar. Los administradores pueden crear directivas de grupo y luego aplicarlas a uno o más usuarios, servidores o equipos de escritorio dentro del entorno. En general, estas configuraciones modifican opciones del registro de Windows, pero es más fácil configurar las opciones mediante directivas de grupo que realizar cambios en el registro manualmente.

Funcionamiento de las directivas de grupo

La configuración de las directivas se basa en las **plantillas administrativas de la directiva de grupo.** Estas plantillas proporcionan una lista de opciones de configuración. Por ejemplo, una opción para un usuario o equipo es Do not keep history of recently opened documents (no mantener un histórico de los documentos recientes, ver Figura 1). Cuando se establece la opción, se realiza el cambio apropiado en el registro de Windows de las sesiones de los usuarios y los equipos a los que se aplica la directiva.

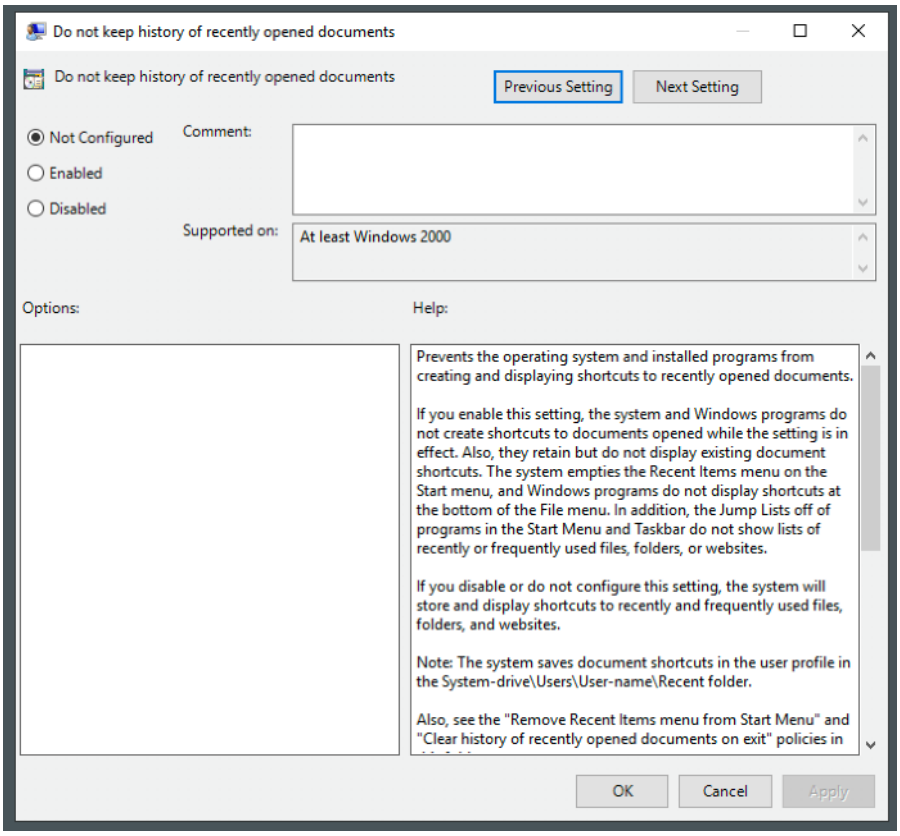


Figura 1. Directiva que modifica cómo se comporta el histórico de documentos recientes. Fuente: elaboración propia.

De manera predeterminada, Windows viene con varias plantillas administrativas. Además, los administradores y los desarrolladores de aplicaciones pueden crear sus propios archivos de plantilla administrativa para una funcionalidad específica.

La mayoría de los elementos de la directiva de grupo tienen tres opciones de configuración diferentes (ver Figura 1):

- Enabled: especifica que se ha establecido una configuración para esta GPO. Algunas GPO requieren que se establezcan valores u opciones concretos. Por ejemplo, la directiva de *account lockout* (bloqueo de cuenta) debe especificar cuántos intentos de inicio de sesión incorrectos pueden realizarse antes de que la cuenta se bloquee.

- Disabled. Especifica que esta opción está desactivada, es decir, que el administrador del sistema desea no permitir ciertas funciones.

- Not Configured. Especifica que esta configuración no se ha habilitado ni deshabilitado. Simplemente establece que esta directiva de grupo no establece esta configuración, aunque otras GPO pueden haberla configurado.

La configuración de la directiva de grupo puede aplicarse a dos tipos de objetos: usuarios y equipos, tanto locales como de Directorio Activo. Tanto los usuarios como los equipos pueden organizarse en unidades organizativas y las GPO pueden aplicarse precisamente a nivel de OU, por lo que este tipo de configuración simplifica la administración de un número arbitrario de objetos con una única GPO.

Las principales opciones que puede configurar dentro de las directivas de grupo son las siguientes:

- **Configuración de *software*** (*software settings*). Se aplican a aplicaciones y *software* específicos que pueden instalarse en los equipos. Los administradores pueden usar esta configuración para hacer que las nuevas aplicaciones estén disponibles para los usuarios finales y para controlar la configuración predeterminada de estas.

- **Configuración de Windows** (*Windows settings*). Permiten a los administradores personalizar el comportamiento del sistema operativo. Incluye opciones para configurar Internet Explorer (incluida la página de inicio predeterminada y otras configuraciones), seguridad (como la política de cuenta) o el registro de eventos.

- **Plantillas administrativas** (*administrative templates*). Se utilizan para configurar aún más las configuraciones de usuario y equipo. Además de las opciones predeterminadas disponibles, los administradores del sistema pueden crear sus propias plantillas administrativas con opciones personalizadas.

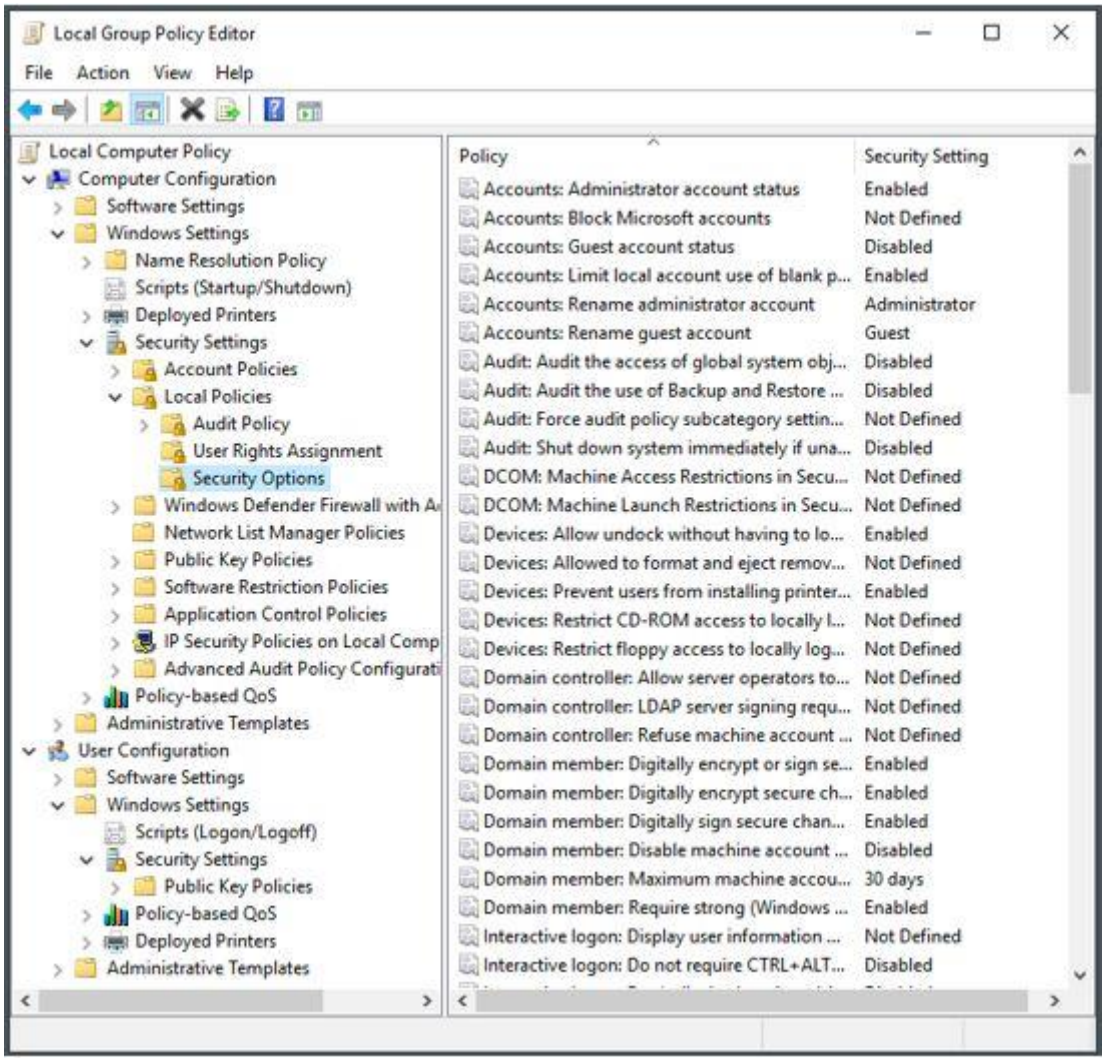


Figura 2. Opciones de seguridad en una directiva de equipo. Fuente: elaboración propia.

Configurar las directivas

Para que sean más fáciles de administrar, las directivas de grupo se agrupan en los ya citados objetos de directiva de grupo o GPO. Los GPO actúan como contenedores de directivas, lo que simplifica la administración de estas. Por ejemplo, un administrador puede tener diferentes políticas para usuarios y equipos en diferentes departamentos. Según estos requisitos, puede crear un GPO para los miembros del departamento de ventas y otro para los miembros del departamento de ingeniería. Luego, podría aplicar los GPO a la unidad organizativa para cada departamento.

Otro concepto importante es que la configuración de la directiva de grupo es jerárquica. Hay cuatro niveles que determinan la prioridad de procesamiento de GPO:

- Local. Cada equipo tiene un objeto de directiva de grupo que se almacena localmente.
- Sitios. La configuración de GPO de un sitio se aplica a todos los dominios y servidores que forman parte de este.
- Dominios. Los dominios son el tercer nivel al que los administradores pueden asignar GPO. La configuración de GPO ubicada en el nivel de dominio se aplicará a todos los objetos de usuario y equipo dentro del dominio.
- Unidades organizativas (OU). Es el nivel de configuración más granular. Si la estructura de OU está bien planificada, resultará fácil hacer asignaciones lógicas de GPO a departamentos o unidades de negocio.

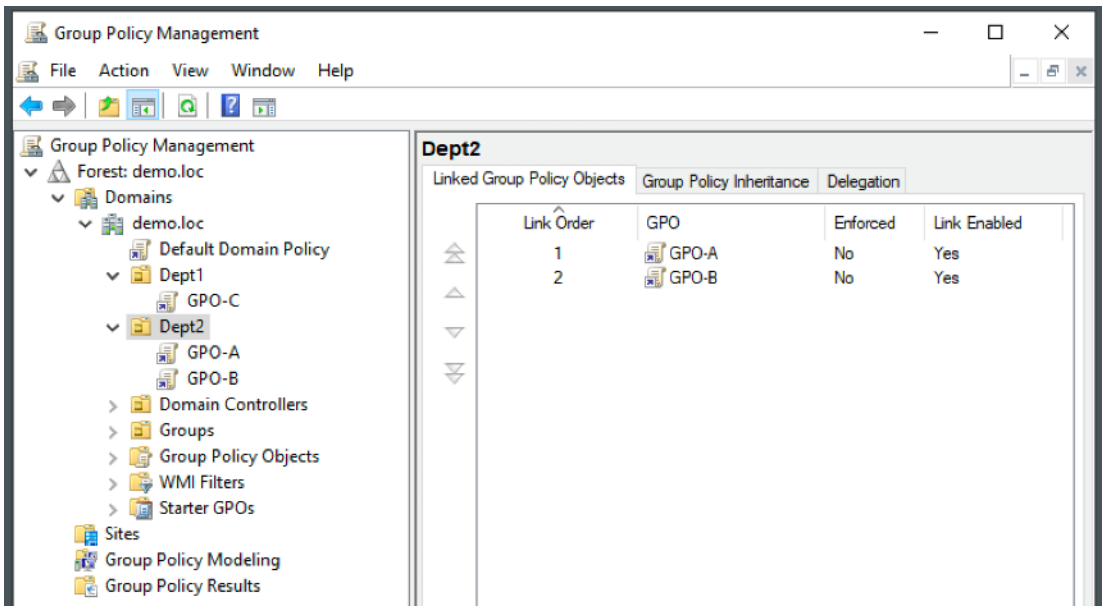


Figura 3. GPO a nivel de dominio y de unidad organizativa. Fuente: elaboración propia.

La Figura 3 muestra la consola de administración de directivas de grupo. La Default Domain Policy está enlazada a nivel de dominio, la GPO-C está enlazada en la OU Dept1 y las GPO-A y GPO-B están enlazadas en la OU Dept2.

Herencia de GPO

En la mayoría de los casos, la configuración de las directivas de grupo es **acumulativa.** Por ejemplo, un GPO a nivel de dominio podría especificar que todos los usuarios dentro del dominio deben cambiar su contraseña cada sesenta días, y un GPO en el nivel de OU podría especificar el fondo de escritorio predeterminado para todos los usuarios y equipos dentro de esa OU. En este caso, se aplican ambas configuraciones, por lo que los usuarios dentro de la unidad organizativa se ven obligados a cambiar su contraseña cada sesenta días y tienen la configuración predeterminada de escritorio.

¿Qué sucede si hay un **conflicto en la configuración**? Por ejemplo, en un escenario donde un GPO en el nivel del sitio especifica que los usuarios deben usar un fondo de pantalla rojo y otro GPO en el nivel de UO especifica que deben usar un fondo de pantalla verde. De manera predeterminada, la configuración en el nivel más específico (en este caso, la unidad organizativa que contiene el usuario) anula las de niveles más generales.

Las directivas de los GPO se aplican en el orden del listado anterior: primero las locales, luego las de sitio, las de dominio y, finalmente, las de OU, desde las OU superiores en el árbol a las más profundas. Prevalecerá la última configuración aplicada de una directiva. En el ejemplo anterior, los usuarios tendrán el fondo de pantalla verde.

Aunque el comportamiento predeterminado es que la configuración sea acumulativa y heredada, este comportamiento se puede modificar con **dos opciones:**

- Bloqueo de herencia. Especifica que las directivas de grupo para un objeto contenedor no se heredan de los contenedores superiores. Esto puede ser útil, por ejemplo, cuando una unidad organizativa secundaria requiere una configuración completamente diferente de una unidad organizativa principal.

- Forzar herencia de directivas. Cuando se activa esta opción en un GPO, se garantiza que todos los objetos de nivel inferior hereden estas configuraciones. En algunos casos, los administradores desean asegurarse de que la herencia de la directiva de grupo no esté bloqueada en otros niveles. Por ejemplo, se puede aplicar en una política corporativa de contraseña para que los administradores de OU inferiores no la bloqueen (hay que recordar que las OU permiten delegar la administración de objetos, por lo que los administradores de una OU no tienen por qué ser los mimos que se encargan del domino).

El último concepto que hay que considerar en la aplicación de GPO es que, si existe un conflicto entre la directiva aplicada a un equipo y a un usuario, prevalece la **configuración del usuario.** Esto es relevante, porque los objetos de usuario y de equipo no tienen por qué pertenecer a la misma OU y, por tanto, pueden recibir configuraciones diferentes. La configuración del usuario es más específica y permite a los administradores realizar cambios para usuarios individuales, independientemente del equipo en el que inician sesión.

9.3. PowerShell remota

PowerShell se puede usar como una herramienta de **administración remota.** Este capítulo muestra cómo hacerlo: en primer lugar, se explica cómo configurar un servidor para que acepte conexiones remotas de PowerShell y, a continuación, cómo conectarse al mismo para obtener información y realizar cambios.

Preparar el servidor remoto

Solo hay un par de elementos que deben ejecutarse y habilitarse en los servidores remotos para que acepten sesiones de PowerShell desde una máquina diferente. La comunicación remota de PowerShell está habilitada de manera predeterminada en equipos Windows Server 2012 y superiores y es posible que no sea necesario seguir

estos pasos. Sin embargo, si la funcionalidad ha sido deshabilitada manualmente o con una política de dominio, o si se intenta acceder a un equipo más antiguo, estos son los elementos para tener en cuenta:

- El servicio `WinRM`: el servicio `WinRM` es parte de la administración remota de Windows Server. Simplemente hay que asegurarse de que este servicio se esté ejecutando. Esto se puede verificar desde la consola de `services.msc` (es decir, la consola Servicios de MMC) o con el comando `Get-Service WinRM`.

- `Enable-PSRemoting -Force`: este comando debe ser ejecutado en cada servidor que vaya a aceptar conexiones remotas. Será necesaria una conexión por RDP para ejecutarlo, pero también se puede hacer preparando una imagen maestra con este comando. La ejecución de `Enable-PSRemoting` intenta iniciar el servicio `WinRM` (lo conseguirá si está parado, pero fallará si el servicio está deshabilitado), configura el sistema para aceptar conexiones remotas y crear una regla de *firewall* en el sistema para permitir este tráfico.

- Habilitar conexiones desde otros dominios o grupos de trabajo: si tanto el servidor administrado como el equipo que inicia la conexión están en el mismo dominio, como suele ser el caso en un entorno corporativo, entonces la autenticación entre máquinas es fácil de lograr, porque confían automáticamente entre sí. Sin embargo, si ambos equipos están en dominios diferentes, que no tienen una relación de confianza entre ellos, o si al menos uno de ellos pertenece a un grupo de trabajo (es decir, no es miembro de un dominio), entonces es necesario configurar el equipo manualmente, para que confíe en el equipo remoto que se va a conectar. Por ejemplo, si el equipo desde el que se inicia la conexión tiene `win-admin` como nombre de *host,* el comando necesario sería el siguiente:

```
Set-Item wsman:\localhost\client\Trustedhosts win-admin
```

Conexión al servidor remoto

Hay dos opciones a la hora de usar PowerShell de forma remota: por un lado, se pueden ejecutar cada comando en el sistema remoto de manera individual y, por otro, se puede abrir una sesión de PowerShell interactiva, de la misma manera que se puede iniciar una sesión interactiva de SSH en Linux.

La primera opción pasa por usar el parámetro **`-ComputerName`**. Muchos de los *cmdlets* disponibles en PowerShell, en particular los que comienzan con `Get-`, se pueden usar con el parámetro `-ComputerName`. Esto especifica que el comando en cuestión debe ejecutarse en el sistema remoto que especifique este parámetro. Por ejemplo, para comprobar si el servicio del programador de tareas está arrancado en el servidor `server1`, habría que ejecutar el siguiente comando:

```
Get-Service schedule -ComputerName server1
```

El parámetro acepta más de un nombre de *host,* por lo que, en caso de especificar varios, la salida mostrará la ejecución del comando en cada uno de los equipos.

Por otro lado, a veces es más cómodo usar **`Enter-PSSession`** e iniciar una sesión interactiva en la que ejecutar los *cmdlets.* Al ejecutar `Enter-PSSession`, la línea de comandos pasa a ser la *shell* del equipo remoto. Los *cmdlets* se ejecutarán en esa *shell,* que tiene su propio entorno y sus propias variables. Para arrancar una consola en el servidor `server1`, habría que ejecutar:

```
Enter-PSSession -ComputerName server1
```

En ambos casos, puede ocurrir que la cuenta de usuario del equipo sea diferente de la cuenta del equipo remoto. En ese caso, el parámetro `-Credential` sirve para indicar la cuenta de usuario. PowerShell abrirá un cuadro de diálogo para solicitar la contraseña.

```
Enter-PSSession -ComputerName server1 -Credential USERNAME
```

9.4. Herramientas de sistema

Este capítulo repasa algunas de las herramientas de mantenimiento y diagnóstico que un administrador usará en su día a día. Aunque no sean tan automatizables como PowerShell, porque son herramientas gráficas, son esenciales para facilitar la vida de cualquier administrador.

Visor de eventos

El Visor de eventos, o Event Viewer, es el visor de *logs* de Windows (Krause, 2019). Las aplicaciones pueden escribir sus *logs* en un fichero de texto, pero Windows ofrece un motor nativo para ello. Los eventos están categorizados en función de su origen (ver Figura 4), por lo que encontrar un evento concreto requiere revisar varias carpetas. El Visor de eventos se basa en la consola MMC, por lo que se puede acceder al Visor desde otro equipo sin necesidad de iniciar una sesión por RDP.

Los eventos precedidos por un icono azul son informativos, los amarillos indican una advertencia y los rojos son errores. De alguna manera, equivalen a los niveles de los tradicionales de `INFO`, `WARNING` y `ERROR`.

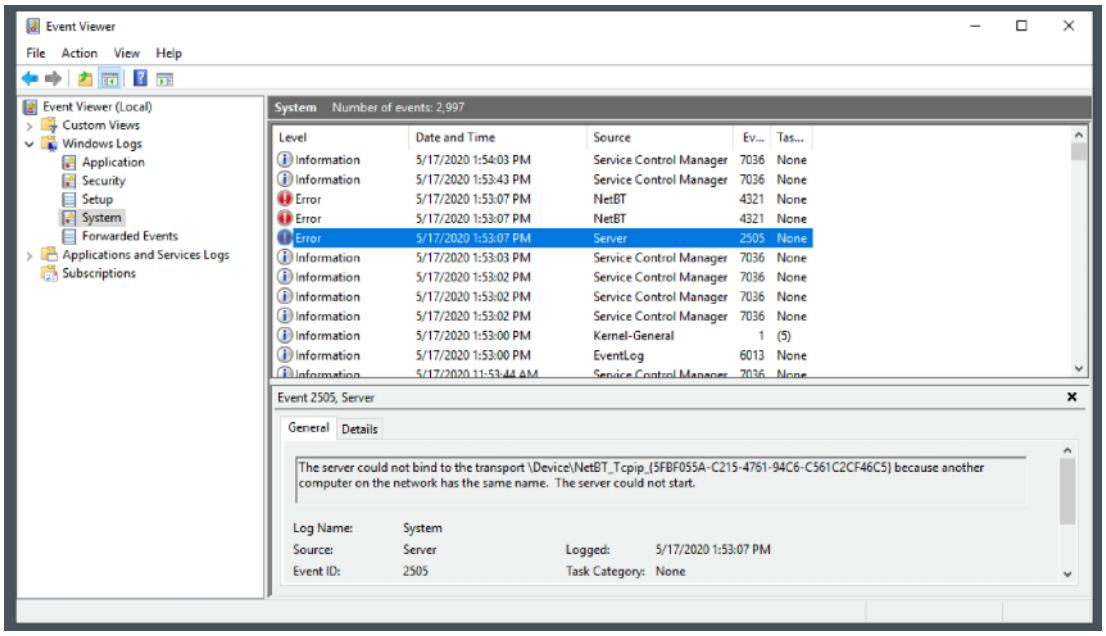


Figura 4. Visor de eventos. Fuente: elaboración propia.

Administrador de tareas

El Administrador de tarea, o Task Manager (Krause, 2019), es una herramienta que ha existido en todos los sistemas operativos Windows desde los primeros días de la interfaz gráfica, pero ha evolucionado bastante a lo largo de los años. Se invoca típicamente presionando `Ctrl + Alt + Supr` en el teclado y luego haciendo clic en Administrador de tareas o haciendo clic derecho en la barra de tareas y luego seleccionando Administrador de tareas. También se puede iniciar con la combinación de teclas `Ctrl + Shift + Esc` o escribiendo `taskmgr` en el cuadro de diálogo `Ejecutar`.

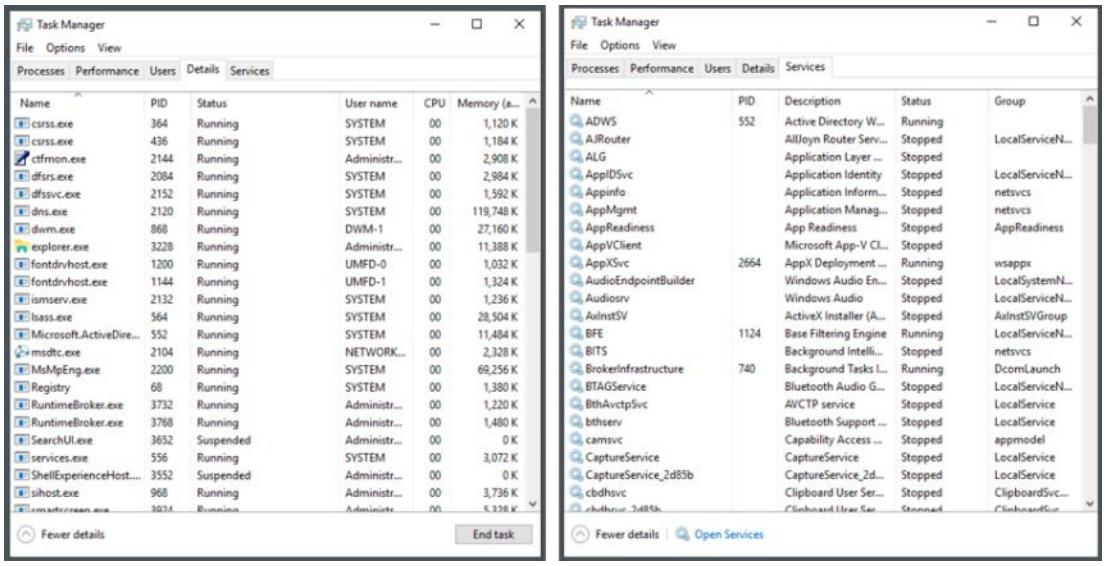


Figura 5. Administrador de tareas. Fuente: elaboración propia.

Las pestañas más relevantes para administrar procesos son **Detalles y Servicios.** La primera muestra qué aplicaciones se están ejecutando actualmente en el sistema. Se puede obtener información como el PID, el usuario que la ha iniciado, la llamada de línea de comandos, el uso de CPU y memoria, etc. Es, a grandes rasgos, un sustituto del comando `top` de Linux. Además, es posible detener la ejecución de un proceso con el menú contextual.

Algunos de los procesos pertenecerán a servicios de Windows. La pestaña Servicios muestra el estado de estos y permite arrancarlos, pararlos, etc. No es un sustituto completo de la consola Servicios de MMC, porque no se pueden ver y editar los detalles de cada servicio.

Programador de tareas

El Programador de tareas (Perrott, 2019), o Task Scheduler, permite programar tareas administrativas para que se **ejecuten automáticamente.** Muchos de los componentes de Windows tienen sus propias tareas listas para usar. Es, en cierta medida, equivalente a `cron`. Al igual que el Visor de eventos, el Programador de tareas es un *snapin* de la consola MMC.

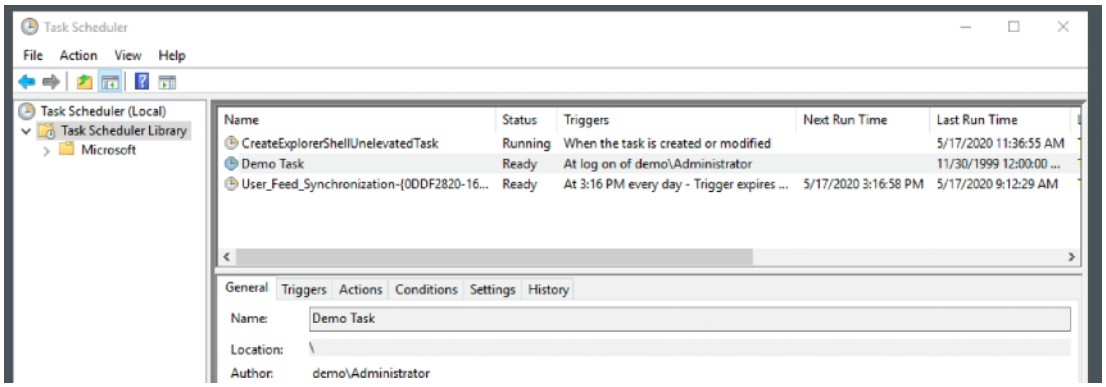


Figura 6. Programador de tareas. Fuente: elaboración propia.

Las tareas pueden ser llamadas a ejecutables o *scripts,* por lo que se puede programar cualquier tarea automatizable. Los desencadenantes, o *triggers*, pueden ser desde una hora concreta hasta un inicio de sesión, el arranque del equipo, un evento, un bloqueo de pantalla, etc.

Sysinternals

La web de Sysinternals, nacida en 1996 como **Ntinternals**, ofrece multitud de herramientas de diagnóstico para Windows. Vienen a cubrir la necesidad de muchos administradores, de tener herramientas potentes para tareas de mantenimiento que no son posibles, o al menos no son fáciles, en un sistema Windows recién instalado. Por ejemplo, **PsExec** es una herramienta de línea de comandos que permite ejecutar acciones y *scripts* en máquinas remotas. Actualmente, esto es relativamente fácil gracias a PowerShell, pero `PsExec` ha estado disponible mucho antes de la primera versión de esta consola.

Otra de las herramientas destacadas es **Process Explorer**. Viene a ser un sustituto del administrador de tareas: ofrece mucha más información de cada proceso y puede mostrar los procesos en orden jerárquico (ver Figura 7).

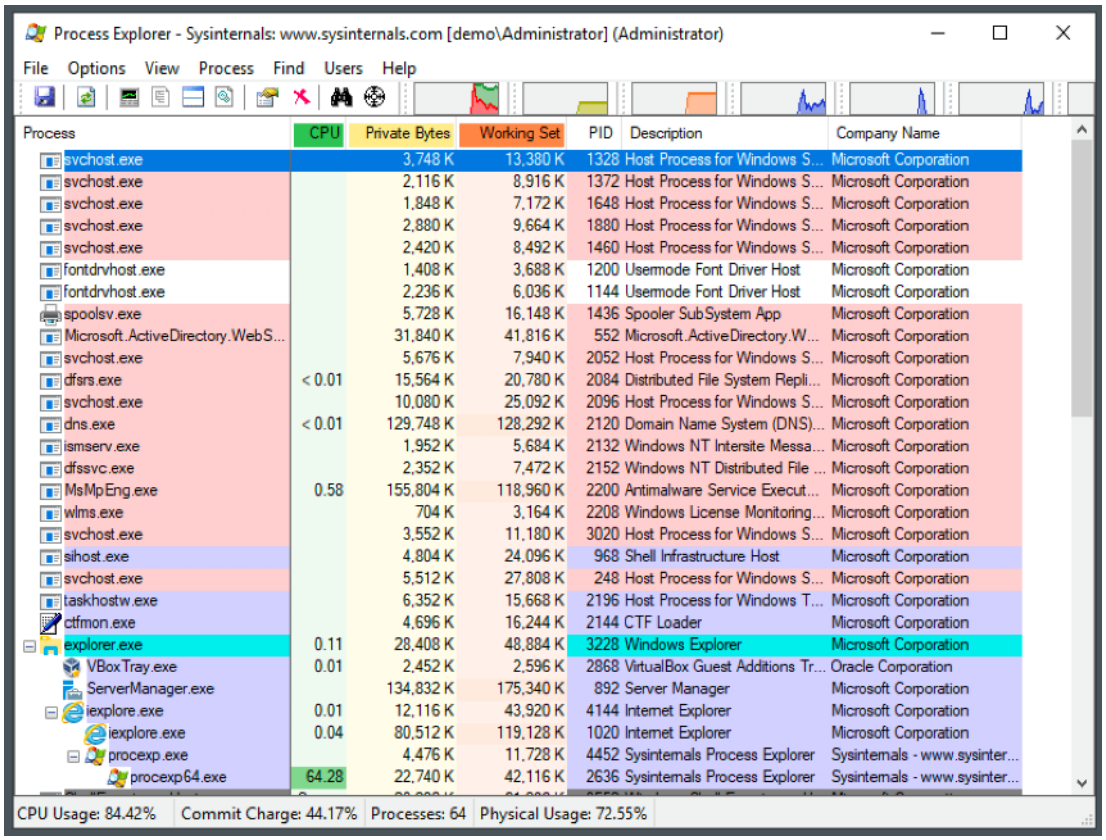


Figura 7. Process Explorer. Fuente: elaboración propia.

Aunque su objetivo puede parecer superfluo, la herramienta **BgInfo** está instalada en numerosos servidores. BgInfo se ejecuta en cada inicio de sesión y personaliza el fondo de pantalla con un bloque de texto. Se usa típicamente para mostrar información básica del equipo, como nombre de *host*, espacio libre en disco, dirección IP, etc. La Figura 8 muestra la pantalla de configuración de BgInfo y el fondo de escritorio que se obtiene tras aplicar la configuración.

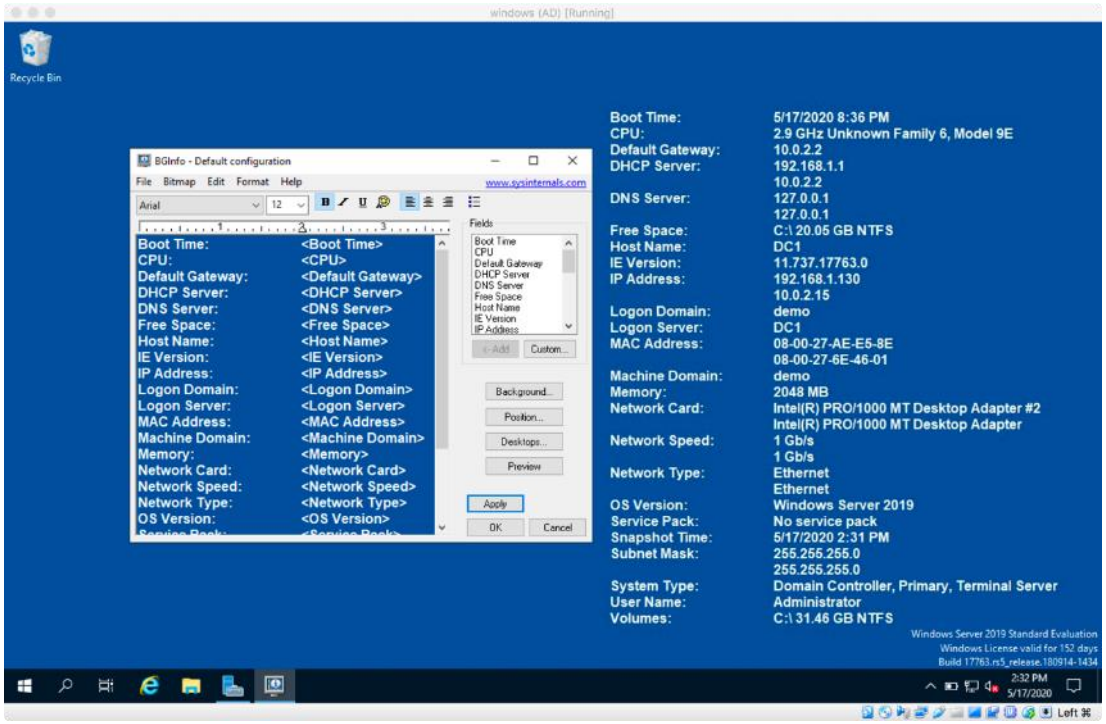
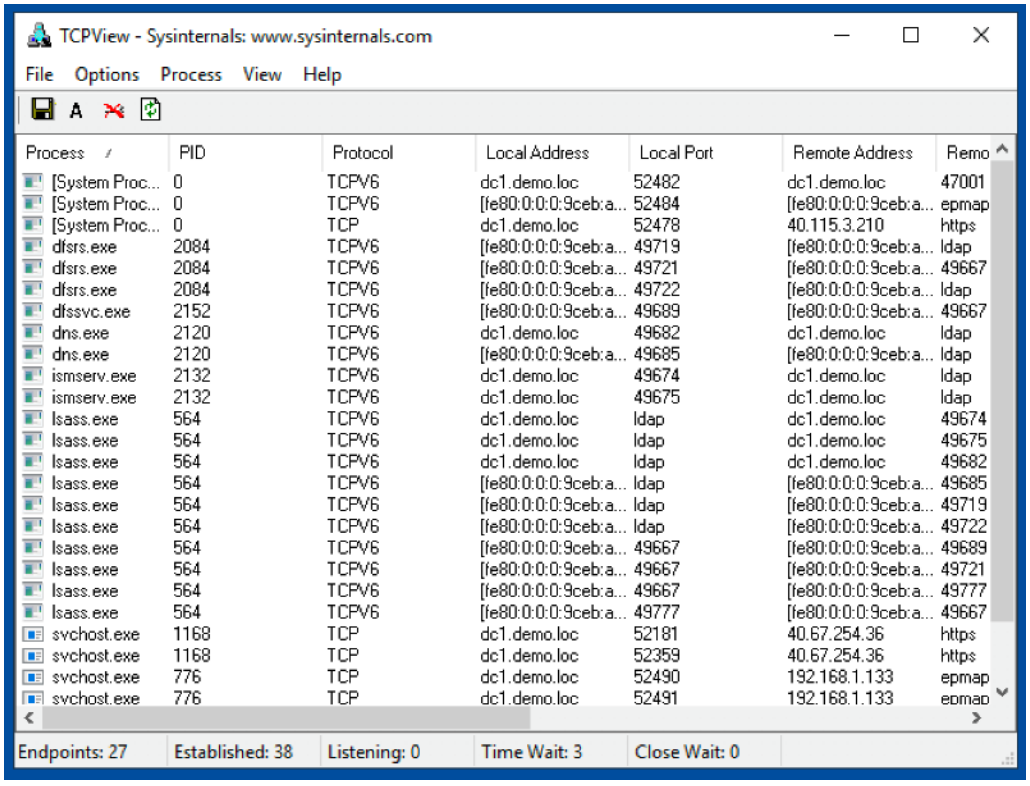


Figura 8. BgInfo. Fuente: elaboración propia.

TCPview es una alternativa gráfica a `netstat`. Muestra información detallada de los puertos abiertos y las conexiones establecidas.



Process	PID	Protocol	Local Address	Local Port	Remote Address	Remo
[System Proc...	0	TCPV6	dc1.demo.loc	52482	dc1.demo.loc	47001
[System Proc...	0	TCPV6	[fe80:0:0:0:9ceb:a...	52484	[fe80:0:0:0:9ceb:a...	epmap
[System Proc...	0	TCP	dc1.demo.loc	52478	40.115.3.210	https
dfsrs.exe	2084	TCPV6	[fe80:0:0:0:9ceb:a...	49719	[fe80:0:0:0:9ceb:a...	ldap
dfsrs.exe	2084	TCPV6	[fe80:0:0:0:9ceb:a...	49721	[fe80:0:0:0:9ceb:a...	49667
dfsrs.exe	2084	TCPV6	[fe80:0:0:0:9ceb:a...	49722	[fe80:0:0:0:9ceb:a...	ldap
dfssvc.exe	2152	TCPV6	[fe80:0:0:0:9ceb:a...	49689	[fe80:0:0:0:9ceb:a...	49667
dns.exe	2120	TCPV6	dc1.demo.loc	49682	dc1.demo.loc	ldap
dns.exe	2120	TCPV6	[fe80:0:0:0:9ceb:a...	49685	[fe80:0:0:0:9ceb:a...	ldap
ismserv.exe	2132	TCPV6	dc1.demo.loc	49674	dc1.demo.loc	ldap
ismserv.exe	2132	TCPV6	dc1.demo.loc	49675	dc1.demo.loc	ldap
lsass.exe	564	TCPV6	dc1.demo.loc	ldap	dc1.demo.loc	49674
lsass.exe	564	TCPV6	dc1.demo.loc	ldap	dc1.demo.loc	49675
lsass.exe	564	TCPV6	dc1.demo.loc	ldap	dc1.demo.loc	49682
lsass.exe	564	TCPV6	[fe80:0:0:0:9ceb:a...	ldap	[fe80:0:0:0:9ceb:a...	49685
lsass.exe	564	TCPV6	[fe80:0:0:0:9ceb:a...	ldap	[fe80:0:0:0:9ceb:a...	49719
lsass.exe	564	TCPV6	[fe80:0:0:0:9ceb:a...	ldap	[fe80:0:0:0:9ceb:a...	49722
lsass.exe	564	TCPV6	[fe80:0:0:0:9ceb:a...	49667	[fe80:0:0:0:9ceb:a...	49689
lsass.exe	564	TCPV6	[fe80:0:0:0:9ceb:a...	49667	[fe80:0:0:0:9ceb:a...	49721
lsass.exe	564	TCPV6	[fe80:0:0:0:9ceb:a...	49667	[fe80:0:0:0:9ceb:a...	49777
lsass.exe	564	TCPV6	[fe80:0:0:0:9ceb:a...	49777	[fe80:0:0:0:9ceb:a...	49667
svchost.exe	1168	TCP	dc1.demo.loc	52181	40.67.254.36	https
svchost.exe	1168	TCP	dc1.demo.loc	52359	40.67.254.36	https
svchost.exe	776	TCP	dc1.demo.loc	52490	192.168.1.133	epmap
svchost.exe	776	TCP	dc1.demo.loc	52491	192.168.1.133	epmap

Figura 9. TCPview. Fuente: elaboración propia.

9.5. Referencias bibliográficas

Krause, J. (2019). *Mastering Windows Server 2019: The Complete Guide for IT Professionals to Install and Manage Windows Server 2019 and Deploy New Capabilities* (2. ª ed.). Packt Publishing.

Microsoft. (2021, junio 22). *Windows Sysinternals.* https://docs.microsoft.com/en-us/sysinternals/

Panek, W. (2018). *MCSA Windows Server 2016 Complete Study Guide*. Sybex.

Perrott, S. (2019). *Windows Server 2019 & PowerShell All-in-One For Dummies*. Wiley.

A fondo

GPO a fondo

Panek, W. (2019). *MCSA Windows Server 2016 Complete Study Guide* (cap. XXI, pp. 961-1023). Sybex.

Aunque es algo denso, el capítulo dedicado a las directivas de grupo es muy útil para entender en profundidad el funcionamiento y la administración de estos objetos.

Sysinternals

Russinovich, M. (2020, abril 28). *Sysinternals Update April 2020* [Vídeo]. YouTube. https://www.youtube.com/watch?v=_MUP4tgdM7s

Mark Russinovich fue el creador de las herramientas Sysinternals antes de que estas fueran ofrecidas oficialmente por Microsoft. Aunque ahora es CTO (*chief information officer*) de Azure, Russinovich sigue participando en el desarrollo de Sysinternals. Una prueba de ello es este vídeo, en el que habla de las últimas novedades y aspira a convertir en una serie de noticias mensual.

Test

1. ¿Cuál de las siguientes herramientas es de Sysintenals?

A. `PsExec`.

B. BgInfo.

C. Process Explorer.

D. Todas las anteriores.

2. ¿Cuál de las siguientes herramientas se arranca como un *snapin* de MMC?

A. Visor de eventos.

B. Administrador de tareas.

C. Process Explorer.

D. TCPview.

3. ¿Cuál es el nivel que tiene preferencia a la hora de aplicar una GPO?

A. Local.

B. OU.

C. Dominio.

D. Sitio.

4. ¿A qué dos tipos de objetos se aplican las directivas de una GPO?

A. Usuarios y grupos de usuarios.

B. Usuarios y OU.

C. Equipos y usuarios.

D. Equipos y grupos de equipos.

5. ¿Qué hace la opción de bloquear herencia en las directivas de grupo?

A. Facilita la administración de GPO, copiando las de niveles superiores a una OU concreta.

B. Bloquea el uso de GPO en una OU concreta.

C. Evita que las directivas de GPO aplicadas en un nivel superior de la jerarquía se apliquen en una OU concreta.

D. Ninguna de las anteriores.

6. ¿Qué opción hay que seleccionar en una directiva de grupo para usar su valor por defecto?

A. Not Configured.

B. Enabled.

C. Disabled.

D. Unknown.

7. ¿Cómo se inicia una sesión remota de PowerShell en `server1`?

A. `Get-PowerShell -Host server1`.

B. `Invoke-PSSession -ComputerName server1`.

C. `Enter-PSSession -ComputerName server1`.

D. `ssh administrator@server1`.

8. ¿Qué herramienta permite ver *logs* nativos de Windows?

A. Notepad.

B. LogViewer.

C. Visor de eventos.

D. `tail`.

9. ¿Qué herramienta nativa es el equivalente a `cron` en Windows?

A. Programador de tareas.

B. `wincron`.

C. Servicios.

D. No hay un servicio equivalente.

10. ¿Qué herramienta es el equivalente a `netstat`?

A. TCPview.

B. `winTCP`.

C. NetServer.

D. TCPBrowser.