

Actividad 10. gestión y monitorización de procesos con comandos en sistemas operativos Linux y Windows

Control de Procesos en CMD de Windows

1. Visualización de Procesos

tasklist: Muestra una lista de todos los procesos en ejecución, incluyendo el nombre de la imagen, ID de proceso (PID), estado, uso de CPU y memoria.

Ejemplo básico: tasklist

Nombre de imagen	PID	Estado	Uso CPU	Uso Memoria
explorer.exe	1576	Consola	00	112 K
tasklist.exe	6256	Consola	00	40 K
cmd.exe	4444	Consola	00	36 K
dwm.exe	1068	Consola	01	100 K
services.exe	4024	Consola	00	84 K

Argumentos:

Parámetro	Descripción
/s <computer>	Especifica el nombre o la dirección IP de un equipo remoto (no se pueden usar barras diagonales inversas). La opción predeterminada es el equipo local.
/u <domain>\<username>	Ejecuta el comando con los permisos de cuenta del usuario especificado por <username> o por <domain>\<username>. El parámetro /u solo se puede especificar si también se especifica /s . El valor predeterminado son los permisos del usuario que ha iniciado sesión actualmente en el equipo que emite el comando.
/p <password>	Especifica la contraseña de la cuenta de usuario especificada en el parámetro /u .
/m <module>	Enumera todas las tareas con módulos DLL cargados que coinciden con el nombre de patrón especificado. Si no se especifica el nombre del módulo, esta opción muestra todos los módulos cargados por cada tarea.
svc	Enumera toda la información del servicio para cada proceso sin truncamiento. Válido cuando el parámetro /fo se establece en table .
/v	Muestra información detallada de la tarea en la salida. Para obtener una salida detallada completa sin truncamiento, use /v y /svc juntos.
/fo {table list csv}	Especifica el formato que se va a usar para la salida. Los valores válidos son table , lista y csv . El formato predeterminado para la salida es table .
/nh	Suprime los encabezados de columna en la salida. Válido cuando el parámetro /fo se establece en table o csv .
/fi <filter>	Especifica los tipos de procesos que se van a incluir o excluir de la consulta. Puede usar más de un filtro o usar el carácter comodín (\) para especificar todas las tareas o nombres de imagen. Los filtros válidos se enumeran en la sección Nombres, operadores y valores de filtro de este artículo.
/?	Muestra la ayuda en el símbolo del sistema.

Ejemplos:

Para enumerar todas las tareas con un identificador de proceso superior a 1000 y mostrarlas en formato csv, escriba:

```
tasklist /v /fi "PID gt 1000" /fo csv
```

Para enumerar los procesos del sistema que se están ejecutando actualmente, escriba:

```
tasklist /fi "USERNAME ne NT AUTHORITY\SYSTEM" /fi "STATUS eq running"
```

Para enumerar información detallada de todos los procesos que se están ejecutando actualmente, escriba:

```
tasklist /v /fi "STATUS eq running"
```

Consideraciones:

Los comandos CMD pueden ser más complejos para usuarios principiantes que la interfaz gráfica del Administrador de tareas.

Sin embargo, los comandos CMD ofrecen mayor flexibilidad y control para usuarios más experimentados.

2. Administración de Procesos

1. Finalización de Procesos

taskkill: Finaliza uno o más procesos o tareas. Los procesos se pueden finalizar por el identificador del proceso o el nombre de la imagen. Puede usar el comando tasklist para determinar el id. de proceso (PID) del proceso que se va a finalizar.

Ejemplo básico:

```
taskkill /pid [PID] /f
```

Termina el proceso con el ID especificado ([PID]) de forma forzosa (/f).

Argumentos:

Parámetro	Descripción
/s <computer>	Especifica el nombre o la dirección IP de un equipo remoto (no se pueden usar barras diagonales inversas). La opción predeterminada es el equipo local.
/u <domain>\<username>	Ejecuta el comando con los permisos de cuenta del usuario especificado por <username> o por <domain>\<username>. El parámetro /u solo se puede especificar si también se especifica /s . El valor predeterminado son los permisos del usuario que ha iniciado sesión actualmente en el equipo que emite el comando.
/p <password>	Especifica la contraseña de la cuenta de usuario especificada en el parámetro /u .
/fi <filter>	Aplica un filtro para seleccionar un conjunto de tareas. Puede usar más de un filtro o usar el carácter comodín (*) para especificar todas las tareas o nombres de imagen. Los filtros válidos se enumeran en la sección Nombres, operadores y valores de filtro de este artículo.
/pid <processID>	Especifica el id. de proceso del proceso que se va a finalizar.
/im <imagenname>	Especifica el nombre de imagen del proceso que se va a finalizar. Use el carácter comodín (*) para especificar todos los nombres de imagen.
/f	Especifica que los procesos se forzaron a finalizar. Este parámetro se omite para los procesos remotos; todos los procesos remotos se fuerzan a finalizar.
/t	Finaliza el proceso especificado y los procesos secundarios iniciados por él.

Ejemplos adicionales:

Para finalizar los procesos con identificadores de proceso 1230, 1241 y 1253, escriba:

```
taskkill /im notepad.exe /f
```

Terminar todos los procesos del usuario "luis":

```
taskkill /u luis /f
```

Mostrar una advertencia antes de finalizar un proceso con ID 1234:

```
taskkill /pid 1234 /t /f
```

2. Priorización de Procesos

start: Este comando permite iniciar un ejecutable con una prioridad específica.

Ejemplo básico:

```
start /HIGH [ruta_ejecutable]
```

Inicia el ejecutable especificado ([ruta_ejecutable]) con prioridad alta (/HIGH).

Argumentos:

/LOW: Inicia el ejecutable con prioridad baja.

/NORMAL: Inicia el ejecutable con prioridad normal (predeterminada).

/REALTIME: Inicia el ejecutable con prioridad en tiempo real.

Ejemplos adicionales:

Iniciar el programa "calc.exe" con prioridad alta:

```
start /HIGH C:\Windows\System32\calc.exe
```

Ejecutar el bloc de notas con prioridad baja:

```
start /LOW notepad.exe
```

Consideraciones:

La priorización de procesos debe usarse con cuidado, ya que puede afectar el rendimiento general del sistema.

Se recomienda no otorgar prioridad alta a muchos procesos, ya que puede generar inestabilidad.

Es importante comprender las implicaciones de la priorización antes de modificarla.

4. Monitoreo del Sistema

Systeminfo: Ver estadísticas del sistema:

```
systeminfo
```

typeperf: Uso de recursos del sistema:

```
typeperf "\Procesador(_Total)\% de tiempo de procesador"
```

performance monitor (perfmon): Inicia el Monitor de confiabilidad y rendimiento de Windows en un modo independiente específico.

Argumentos:

Parámetro	Descripción
/res	Inicia la Vista de recursos.
/report	Inicia el conjunto de recopiladores de datos de diagnóstico del sistema y muestra un informe de los resultados.
/rel	Inicia el Monitor de confiabilidad.
/sys	Inicia el Monitor de rendimiento.

Consideraciones:

Las herramientas gráficas ofrecen una interfaz más intuitiva para el monitoreo básico del sistema.

Los comandos CMD proporcionan mayor flexibilidad y control para usuarios más experimentados.

La elección de la herramienta depende de las necesidades y preferencias del usuario.

Control de Procesos en la Terminal de Linux

1. Introducción a la Terminal

Cómo abrir la terminal: Generalmente se abre buscando "Terminal" en el menú de aplicaciones o con el atajo Ctrl+Alt+T.

2. Visualización de Procesos

ps: La herramienta estándar para ver procesos en ejecución. Muestra información básica como ID de usuario (UID), ID de proceso (PID), ID del proceso padre (PPID), comando y estado.

Ejemplo básico:

```
ps
UID  PID  PPID  COMANDO      ESTADO
```

Argumentos:

- e: Muestra todos los procesos en ejecución, incluyendo aquellos que no están asociados a una terminal.
- f: Muestra información más detallada, incluyendo nombre completo del comando, argumentos, uso de CPU y memoria.
- u [nombre_usuario]: Muestra solo los procesos del usuario especificado.
- t: Muestra solo el ID de proceso (PID) de cada proceso.

top: Muestra una lista de los procesos en ejecución en tiempo real, ordenados por uso de CPU. Permite interactuar con la lista para ver información detallada de procesos específicos o matarlos.

Ejemplo básico:

```
top
Tareas en ejecución: 237 Totales: 1.033 En ejecución: 1 Dormidos: 1 Detenidos: 0
Uso CPU: 10.7% Us. usuario: 9.2% Us. sistema: 1.4% Inactivo: 88.3%
Memoria: 13356MiB / 15865MiB Libre: 2509MiB Uso: 10847MiB Buffers: 191MiB
PID USUARIO %CPU %MEM COMMAND
2564 root 22.8 1.2 R/S+ [ksmthread1]
2594 root 19.7 1.1 R/S+ [ksmthread2]
1609 root 10.5 1.1 R/S+ [kworker/u8:0]
1610 root 10.4 1.1 R/S+ [kworker/u8:1]
1611 root 10.4 1.1 R/S+ [kworker/u8:2]
```

Argumentos:

- h: Muestra una breve guía de las teclas de acceso rápido.
- p [PID]: Muestra información detallada del proceso con el ID especificado.
- s [SORT_COLUMN]: Ordena la lista por una columna específica.
- %CPU: Ordena por uso de CPU.
- %MEM: Ordena por uso de memoria.
- k [NUM]: Mata el proceso con el ID especificado.

htop: Una alternativa a top con una interfaz más intuitiva y colorida, que permite usar el mouse para interactuar con la lista de procesos.

Ejemplo básico:

```
htop
```

Nota: He omitido ejemplos de htop en esta respuesta por ser una herramienta visual.

Ejemplos adicionales:

Ver solo los procesos del usuario actual:

```
ps -u $(whoami)
```

Ver los 10 procesos que más CPU usan:

```
ps -eo pid,%cpu,cmd --sort=-%cpu | head -n 10
```

Buscar un proceso por nombre:

```
ps -ef | grep "[nombre_proceso]"
```

Matar un proceso por ID:

```
kill [PID]
```

Consideraciones:

ps es la herramienta más básica y versátil para ver procesos.

top y htop son útiles para monitorear procesos en tiempo real y observar su comportamiento.

La elección de la herramienta depende de las necesidades y preferencias del usuario.

3. Administración de Procesos

1. Finalización de Procesos

Comando: **kill**

Ejemplo básico:

```
kill [PID]
```

Envía la señal SIGTERM al proceso con el ID especificado ([PID]), indicándole que finalice.

Argumentos:

-9: Envía la señal SIGKILL al proceso, terminándolo forzosamente.

-u [nombre_usuario]: Envía la señal al usuario especificado ([nombre_usuario]) y a todos sus procesos.

-s [señal]: Envía una señal específica al proceso.

SIGTERM: Señal predeterminada para finalizar un proceso.

SIGKILL: Señal para finalizar un proceso forzosamente.

Ejemplos adicionales:

Terminar un proceso con ID 1234:

```
kill 1234
```

Finalizar forzosamente un proceso con ID 5678:

```
kill -9 5678
```

Terminar todos los procesos del usuario "maria":

```
kill -u maria
```

2. Priorización de Procesos

Comando: **renice**

Ejemplo básico:

```
renice +n [PID]
```

Aumenta la prioridad del proceso con el ID especificado ([PID]) en n niveles.

Argumentos:

-n: Nivel de prioridad a agregar. Valores válidos entre -20 y 20.

-u [nombre_usuario]: Ajusta la prioridad de los procesos del usuario especificado ([nombre_usuario]).

Ejemplos adicionales:

Aumentar la prioridad del proceso con ID 1234 en 5 niveles:

```
renice +5 1234
```

Disminuir la prioridad del proceso con ID 5678 en 2 niveles:

```
renice -2 5678
```

Ajustar la prioridad de todos los procesos del usuario "pedro" a 10:

```
renice -u pedro 10
```

Consideraciones:

La priorización de procesos debe usarse con cuidado, ya que puede afectar el rendimiento general del sistema. Se recomienda no otorgar prioridad alta a muchos procesos, ya que puede generar inestabilidad. Es importante comprender las implicaciones de la priorización antes de modificarla.

4. Monitoreo del Sistema

top: Muestra una lista de procesos en ejecución en tiempo real, ordenados por uso de CPU. Permite interactuar con la lista para ver información detallada de procesos específicos o matarlos.

Ejemplo básico:

```
top
Tareas en ejecución: 237  Totales: 1.033  En ejecución: 1 Dormidos: 1 Detenidos: 0
Uso CPU: 10.7%  Us. usuario: 9.2%  Us. sistema: 1.4%  Inactivo: 88.3%
Memoria: 13356MiB / 15865MiB  Libre: 2509MiB  Uso: 10847MiB  Buffers: 191MiB
PID  USUARIO      %CPU %MEM COMMAND
2564 root        22.8  1.2 R/S+ [ksmthread1]
2594 root        19.7  1.1 R/S+ [ksmthread2]
1609 root        10.5  1.1 R/S+ [kworker/u8:0]
1610 root        10.4  1.1 R/S+ [kworker/u8:1]
1611 root        10.4  1.1 R/S+ [kworker/u8:2]
```

Argumentos:

- h: Muestra una breve guía de las teclas de acceso rápido.
- p [PID]: Muestra información detallada del proceso con el ID especificado.
- s [SORT_COLUMN]: Ordena la lista por una columna específica.
- %CPU: Ordena por uso de CPU.
- %MEM: Ordena por uso de memoria.
- k [NUM]: Mata el proceso con el ID especificado.

htop: Una alternativa a top con una interfaz más intuitiva y colorida, que permite usar el mouse para interactuar con la lista de procesos.

Ejemplo básico:

```
htop
```

ps: Muestra información básica sobre procesos en ejecución, como ID de usuario, ID de proceso, ID de proceso padre, comando y estado.

Ejemplo básico:

```
ps
UID  PID  PPID  COMANDO          ESTADO
```

Argumentos:

- e: Muestra todos los procesos en ejecución, incluyendo aquellos que no están asociados a una terminal.
- f: Muestra información más detallada, incluyendo nombre completo del comando, argumentos, uso de CPU y memoria.
- u [nombre_usuario]: Muestra solo los procesos del usuario especificado.
- t: Muestra solo el ID de proceso (PID) de cada proceso.

Se pide:

1. Elaborar un documento mostrando el uso con comandos en sistemas operativos Linux y Windows para gestión de procesos