

TOP

NOMBRE

top – muestra los procesos que más CPU consumen

SINOPSIS

top [-] [**d** *intervalo*] [**q**] [**c**] [**S**] [**s**] [**i**]

DESCRIPCIÓN

top proporciona una visión continuada de la actividad del procesador en tiempo real. Muestra un listado de las tareas que hacen un uso más intensivo de la CPU en el sistema, y puede proporcionar una interfaz interactiva para manipular procesos. Puede clasificar las tareas por empleo de CPU, uso de memoria y tiempo de ejecución. puede configurarse mejor que el top normal del conjunto de programas procp. La mayoría de las características pueden seleccionarse mediante una orden interactiva o especificándola en el fichero de configuración personal o general. Hay más información sobre esto más adelante.

OPCIONES DE LA LÍNEA DE ÓRDENES

- d** Especifica el intervalo entre actualizaciones de la pantalla. Esto puede cambiarse con la orden interactiva **s**.
- q** Esto hace que **top** redibuje la pantalla sin intervalo ninguno. Si el que ejecuta el programa tiene privilegios de súper-usuario, top se ejecuta con la prioridad más alta posible.
- S** Especifica el modo acumulativo, donde cada proceso se lista con el tiempo de CPU que él, *así como sus hijos muertos*, ha gastado. Esto es como la opción **-S** de [ps\(1\)](#). Vea la discusión más abajo sobre la orden interactiva **S**.
- s** Le dice a **top** que se ejecute en modo seguro. Esto inhabilita el peligro potencial de las órdenes interactivas (vea más abajo). Un **top** seguro es una cosa curiosa para dejar ejecutándose en una terminal sin dueño.
- i** Arranca **top** descartando cualquier proceso inactivo o zombie. Vea más abajo la orden interactiva **i**.
- c** Muestra la línea de órdenes entera en lugar de solamente el nombre de la orden. El comportamiento predeterminado se ha cambiado puesto que esto parece ser de más utilidad.

DESCRIPCIONES DE LOS CAMPOS

top despliega una variada información sobre el estado del procesador. La pantalla se actualiza cada 5 segundos de forma predeterminada, pero esto se puede cambiar con la opción de la línea de órdenes **d** o la orden interactiva **s**.

uptime

Esta línea muestra el tiempo que el sistema ha estado activo, y las tres medias de carga para el sistema. Las medias de carga son el número medio de procesos listos para ejecutarse durante los últimos 1, 5 y 15 minutos. Esta línea es simplemente como la salida de [uptime\(1\)](#). La línea de uptime puede quitarse o ponerse con la orden interactiva **l** (ele minúscula).

processes

El número total de procesos ejecutándose cuando la última actualización. Este número también se divide en el número de tareas que están ejecutándose, durmiendo, paradas o no-muertas. Las líneas de procesos y estados pueden quitarse o ponerse con la orden interactiva **t**.

CPU states

Muestra el porcentaje de tiempo de CPU en modo de usuario, en modo de sistema, en tareas con la prioridad alterada por nice, y el tiempo de inactividad. (Las tareas con la prioridad alterada por nice son solamente aquellas cuyo valor nice es negativo.) El tiempo transcurrido en las tareas con la prioridad alterada por nice también se contará en el tiempo de sistema y de usuario, así que el total será superior al 100%. Las líneas de procesos y estados y tiempos de CPU pueden quitarse o ponerse con la orden interactiva **t**.

Mem	Datos sobre el empleo de memoria, incluyendo la memoria disponible en total, la memoria libre, la usada, la compartida, y la utilizada para búferes. La línea de la información de memoria puede ponerse o quitarse con la orden interactiva m .
Swap	Datos sobre el espacio de trasiego, incluyendo el total, el disponible y el empleado. Esto y Mem son sencillamente como la salida de free(1) .
PID	El identificador (ID) de proceso (PID) de cada tarea.
PPID	El ID del proceso padre de cada tarea.
UID	El ID de usuario del propietario de la tarea.
USER	El nombre de usuario del propietario de la tarea.
PRI	La prioridad de la tarea.
NI	El valor de nice de la tarea. Valores negativos indican menor prioridad.
SIZE	Se muestra aquí el tamaño del código de la tarea más datos más espacio de pila, en kB.

TSIZE

El tamaño del texto o código de la tarea. Esta da valores extraños para procesos del núcleo y no funciona para procesos ELF.

DSIZE

Tamaño de Datos + Pila. Esto no funciona para procesos ELF.

TRS	Tamaño del texto (código) residente.
SWAP	Tamaño de la parte de la tarea que está en el espacio de trasiego.
D	Tamaño de las páginas marcadas como sucias.
LIB	Tamaño de las páginas de biblioteca usadas. Esto no funciona para procesos ELF.
RSS	Se muestra aquí la cantidad total de memoria física utilizada por la tarea, en kilobytes. Para procesos ELF se cuentan aquí las páginas de biblioteca usadas, para procesos a.out no.

SHARE

Se muestra en esta columna la cantidad de memoria compartida empleada por la tarea.

STAT	Aquí se ve el estado de la tarea. El estado puede ser S para durmiente, D para sueño no interrumpible, R para ejecución, Z para zombies, o T para parados o trazados. Estos estados se modifican por un < al final para procesos con valor de nice negativo, N para un proceso con valor de nice positivo, o W para un proceso en espacio de trasiego (esto no funciona correctamente para procesos del núcleo).
-------------	---

WCHAN

Dependiendo de la disponibilidad de `/boot/psdatabase` o del mapa de enlace del núcleo `/boot/System.map`, esto muestra la dirección o el nombre de la función del núcleo en la cual la tarea está actualmente durmiendo.

- TIME** El tiempo total de CPU que la tarea ha usado desde que empezó. Si el modo acumulativo está activado, también incluye el tiempo de CPU empleado por los hijos del proceso que hayan muerto. Uno puede establecer el modo acumulativo con la opción de la línea de órdenes **S** o cambiarlo con la orden interactiva **S**. La línea de cabecera se cambiará entonces a **CTIME**.
- %CPU** La porción del tiempo de CPU consumido por la tarea desde la última actualización de la pantalla, expresada como un porcentaje del tiempo de CPU total.
- %MEM** La porción de la memoria física ocupada por la tarea.

COMMAND

El nombre de la orden de la tarea, que se truncará si es demasiado largo como para mostrarse en una línea. Las tareas en memoria tendrán una línea de órdenes completa, pero las que estén en espacio de trasiego solamente tendrán el nombre del programa entre paréntesis (por ejemplo, "(getty)").

A, WP

Estos campos del top de kmem no están soportados.

ÓRDENES INTERACTIVAS

Mientras **top** está ejecutándose se reconocen algunas órdenes de una sola tecla. Algunas están deshabilitadas si la opción **s** se ha dado en la línea de órdenes.

espacio

Actualiza la pantalla inmediatamente.

- ^L** Borra y redibuja la pantalla.

h o ?

Despliega una pantalla de ayuda dando un breve resumen de las órdenes, y el estado de los modos seguro y acumulativo.

- k** Mata un proceso. Se le preguntará el PID de la tarea, y la señal a enviarle. Para un asesinato normal, envíe la señal 15. Para asegurarse, pero de forma bastante abrupta, envíe la señal 9. La señal predeterminada, como en [kill\(1\)](#), es 15, **SIGTERM**. Esta orden no está disponible en el modo seguro.
- i** Descarta los procesos inactivos y zombies. Esta orden actúa como interruptor.

n o

Cambia el número de procesos a mostrar. Se le pedirá ese número. Esto sustituye a la determinación automática del número de procesos a mostrar, que se basa en una medida del tamaño de la ventana. Si se especifica 0, entonces top mostrará tantos procesos como quepan en la pantalla; esto es lo predeterminado.

- q** Abandonar.
- r** Vuelve a cambiar la prioridad de un proceso con nice. Se le preguntará el PID de la tarea, y el valor de nice a aplicarle. Un valor positivo hará que un proceso cambie a valores nice negativos, y pierda prioridad. Si es root quien está ejecutando **top**, puede introducir un valor negativo, haciendo que un proceso obtenga una prioridad más alta de la normal. El valor predeterminado de renice es 10. Esta orden no está disponible en el modo seguro.
- S** Esto cambia el modo acumulativo, el equivalente de **ps -S**, esto es, que los tiempos de CPU incluirán los hijos difuntos de un proceso. Para algunos programas, tales como compiladores, que trabajan bifurcándose en varias tareas separadas, el modo normal los hará aparecer como menos

demandantes de lo que en realidad son. Para otros, empero, tales como shells e **init**, este comportamiento es correcto. En cualquier caso, pruebe el modo acumulativo para una visión alternativa del uso de CPU.

- s** Cambia el intervalo entre actualizaciones. Se le pedirá que introduzca el tiempo de demora, en segundos, entre actualizaciones. Se reconocen valores fraccionarios hasta microsegundos. El 0 significa actualización continua. El valor predeterminado es 5 segundos. Observe que valores bajos produce que casi no se pueda ver nada, y aumentan grandemente la carga. Esta orden no está disponible en el modo seguro.

f o F

Añade campos a mostrar o los quita. Vea más adelante para más información.

o o O

Cambia el orden de los campos mostrados. Más información, más adelante.

- l** Cambia entre mostrar o no la información de carga media y de tiempo de actividad del sistema.
- m** Cambia entre mostrar o no la información sobre la memoria.
- t** Cambia entre mostrar o no la información de procesos y estados de la CPU.
- c** Cambia entre mostrar o no el nombre de la orden o la línea de órdenes completa.
- M** Clasifica las tareas por empleo de memoria residente.
- P** Clasifica las tareas por el empleo de CPU (esto es lo predeterminado).
- T** Clasifica las tareas por tiempo / tiempo acumulativo.
- W** Escribe la configuración actual en `~/toprc`. Ésta es la forma recomendada de escribir un fichero de configuración de top.

Las Pantallas de Campo y Orden

Tras pulsar **f**, **F**, **o** u **O** se le mostrará una pantalla especificando el orden de campos en la línea superior y pequeñas descripciones de los contenidos de los campos. La cadena de caracteres del orden de los campos sigue la sintaxis siguiente: si la letra en la cadena correspondiente a un campo está en mayúscula, el campo será mostrado. Esto se indica luego por un asterisco enfrente de la descripción del campo.

El orden de los campos corresponde al de las letras en la cadena.

De la pantalla de selección de campos uno puede cambiar el despliegue de un campo pulsando la letra correspondiente.

De la pantalla de orden uno puede mover un campo a la izquierda pulsando la correspondiente letra mayúscula, y similarmente a la derecha con la minúscula.

Ficheros de Configuración

top lee su configuración predeterminada de dos ficheros, `/etc/toprc` y `~/toprc`. El fichero de configuración global se puede emplear para restringir el uso de top al modo seguro para usuarios no privilegiados. Si esto es lo que se desea, el fichero deberá contener en una misma línea una 's' para especificar el modo seguro y un dígito d ($2 \leq d \leq 9$) para el intervalo predeterminado (en segundos). El fichero de configuración personal contiene dos líneas. La primera contiene letras mayúsculas y minúsculas que especifican qué campos en qué orden se van a mostrar. Las letras corresponden a las mismas de las pantallas de Campos o de Orden en top. Como esto no es muy instructivo, se recomienda seleccionar campos y orden en un proceso top en ejecución y salvar la configuración mediante la orden interactiva **W**. La segunda línea es más interesante (e importante). Contiene información sobre las otras opciones. Más importante, si uno ha salvado una configuración en modo seguro, no obtendrá un top inseguro sin quitar la ese minúscula de la segunda línea

de su `~/toprc`. Un dígito especifica el intervalo de tiempo entre actualizaciones, una *eme* mayúscula, el modo acumulativo, una *i* latina minúscula el modo de no-inactivos. Como en el modo interactivo, una *eme*, *ele* y *te* minúsculas suprimen el despliegue de la información sobre memoria, tiempo de actividad del sistema, procesos y estado de la CPU, respectivamente. Actualmente no se admite cambiar el orden clasificatorio predeterminado (por uso de CPU).

OBSERVACIONES

Este **top** basado en **proc**, funciona leyendo ficheros en el sistema de ficheros **proc** montado en `/proc`. Si `/proc` no está montado, **top** no funcionará.

%CPU muestra el porcentaje de tiempo de CPU/tiempo real en el período de tiempo entre actualizaciones. Para la primera actualización, se usa un corto intervalo, y el propio **top** domina el uso de CPU. Después de esto, **top** se irá más abajo, y queda disponible una estimación más fiable del uso de CPU.

Los campos **SIZE** y **RSS** no cuentan las tablas de página ni la **task_struct** de un proceso; esto es al menos 12 kB de memoria que está siempre residente. **SIZE** es el tamaño virtual del proceso (código+datos+pila).

Tenga presente que un proceso debe morir para que su tiempo se grave en su padre por el modo acumulativo. Quizás un comportamiento más útil sería seguir cada proceso hacia adelante, añadiendo tiempo, pero eso sería demasiado costoso, posiblemente prohibitivo. En cualquier caso, haría a **top** incompatible en comportamiento con **ps**.

FICHEROS

`/etc/toprc` El fichero de configuración global.

`~/toprc` El fichero de configuración personal.

VÉASE TAMBIÉN

[`ps\(1\)`](#), [`free\(1\)`](#), [`uptime\(1\)`](#), [`kill\(1\)`](#), [`renice\(1\)`](#).

FALLOS

Si la ventana es menor de aproximadamente 70×7, **top** no formateará la información correctamente.

Muchos campos aún tienen problemas con procesos ELF.

Las pantallas de ayuda no están aún optimizadas para ventanas de menos de 25 líneas.

AUTORES

top fue escrito originalmente por Roger Binns, basándose en el programa `ps` de Branko Lankester <lankeste@fwi.nl>. Robert Nation <nation@rocket.com> lo re-escribió de forma significativa para hacer uso del sistema de ficheros `proc`, según el `ps` basado en `proc` de Michael K. Johnson <johnsonm@redhat.com>. Michael Shields <mjshield@nyx.edu> hizo muchos cambios, incluyendo los modos seguro y acumulativo, y una limpieza general.

Helmut Geyer <Helmut.Geyer@iwr.de> lo cambió grandemente para incluir soporte para campos configurables y otras nuevas opciones, e hizo otra limpieza, además de usar la nueva interfaz `readproc`.

Michael K. Johnson <johnsonm@redhat.com> es ahora el que mantiene el programa.

Por favor, envíe sus informes sobre fallos a <procps-bugs@redhat.com>.

Esta traducción al español ha corrido a cargo de Gerardo Aburrizaga García <gerardo.aburrizaga@uca.es>, para el proyecto PAMELI liderado por Juan Piernas Cánovas <piernas@ditec.es>.

COMMENTS