

2. Agile SysAdmin: Networking and Systems Administration

Instructor: Julián García-Sotoca Pascual





■ Troubleshooting



■ Índice

- Troubleshooting
 - Observar el estado del sistema
 - Herramientas de monitorización
 - Filesystem /proc
 - Logs
 - Resumen herramientas
 - PCP + Vector



■ Índice

- Troubleshooting
 - Observar el estado del sistema
 - **Herramientas de monitorización**
 - Filesystem /proc
 - Logs
 - Resumen herramientas
 - PCP + Vector



■ Observar el estado del sistema

La principal herramienta para observar el sistema es TOP. Proporciona una visión en tiempo real de lo que está pasando en el servidor

```
top - 21:39:34 up 17 days, 23:13, 1 user, load average: 0,85, 1,06, 1,07
Tasks: 312 total, 1 running, 309 sleeping, 0 stopped, 2 zombie
%Cpu0 : 10,0 us, 3,3 sy, 0,3 ni, 83,1 id, 1,3 wa, 0,0 hi, 2,0 si, 0,0 st
%Cpu1 : 13,7 us, 2,3 sy, 0,0 ni, 81,3 id, 0,3 wa, 0,0 hi, 2,3 si, 0,0 st
%Cpu2 : 19,0 us, 2,7 sy, 0,0 ni, 76,0 id, 1,0 wa, 0,0 hi, 1,3 si, 0,0 st
%Cpu3 : 14,4 us, 2,7 sy, 0,0 ni, 81,5 id, 0,3 wa, 0,0 hi, 1,0 si, 0,0 st
KiB Mem : 8044440 total, 311776 free, 6381580 used, 1351084 buff/cache
KiB Swap: 2097148 total, 501656 free, 1595492 used. 491760 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
29773	julian	20	0	991408	466664	66600	S	0,3	5,8	1:28.67	/opt/google/c+
28814	julian	20	0	930948	369244	104936	S	0,3	4,6	2:32.13	/opt/google/c+
2394	julian	20	0	3555092	336412	47728	S	29,7	4,2	90:22.82	/usr/bin/gnom+
27033	julian	20	0	1844768	331704	23948	S	1,0	4,1	0:45.68	/usr/lib/fire+
12579	julian	20	0	1893456	323996	295904	S	1,3	4,0	5:42.58	/usr/lib/virt+
26895	julian	20	0	2811168	314260	73436	S	0,0	3,9	4:24.38	/usr/lib/fire+
22826	julian	20	0	2493892	313932	18844	S	0,0	3,9	1:23.17	/usr/bin/pyth+
13724	julian	20	0	1079364	304176	61552	S	0,3	3,8	3:46.66	/opt/google/c+
24347	julian	20	0	2303140	292016	237592	S	1,3	3,6	3:13.35	/usr/lib/virt+
4404	julian	20	0	1855328	274656	49192	S	1,3	3,4	68:18.97	/opt/google/c+
29603	julian	20	0	891872	244948	57892	S	0,0	3,0	4:21.41	/opt/google/c+
22749	julian	20	0	758208	226468	50128	S	0,3	2,8	0:39.42	/opt/google/c+
24028	julian	20	0	2398136	223200	29988	S	0,0	2,8	1:18.35	/usr/bin/pyth+
17630	julian	20	0	1445616	214552	27280	S	11,2	2,7	2:24.67	/usr/bin/perl+
3551	julian	20	0	1541948	190152	4848	S	0,0	2,4	1:14.18	/usr/bin/gnom+
4668	julian	20	0	1947208	187868	30036	S	0,0	2,3	3:48.80	/opt/google/c+
27179	julian	20	0	1633248	166024	75640	S	2,6	2,1	2:13.24	/usr/lib/fire+
2653	julian	39	19	1014300	153752	4088	S	0,0	1,9	0:06.37	/usr/lib/trac+
2230	julian	20	0	528668	142100	102824	S	5,3	1,8	57:48.17	/usr/lib/xorg+
4452	julian	20	0	481088	134400	81904	S	0,3	1,7	59:23.56	/opt/google/c+



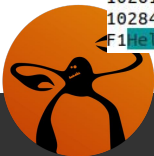
Observar el estado del sistema

HTOP ofrece la misma información pero con un aspecto algo más moderno:

```
 1 [|||||] 9.8% 4 [||] 4.7% 7 [||] 4.0% 10 [||||] 7.3%
 2 [||||] 6.0% 5 [||||] 7.3% 8 [||||] 7.8% 11 [||] 2.7%
 3 [||] 2.0% 6 [||] 4.6% 9 [||||] 7.2% 12 [||||] 6.6%
Mem[|||||||||||||||||||||||||||||||||||||||||] 92.6G/126G Tasks: 51, 517 thr; 3 running
Swp[|] 0K/1.86G Load average: 0.42 0.54 0.54
Uptime: 75 days, 11:09:54
```

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
22381	mongod	20	0	99G	91.8G	23948	S	66.3	72.9	108h	/usr/bin/mongod -f /etc/mongod.conf
5164	mongod	20	0	99G	91.8G	23948	S	3.3	72.9	1h17:37	/usr/bin/mongod -f /etc/mongod.conf
69868	mongod	20	0	99G	91.8G	23948	S	2.7	72.9	1h40:16	/usr/bin/mongod -f /etc/mongod.conf
102856	mongod	20	0	99G	91.8G	23948	S	2.7	72.9	0:00.32	/usr/bin/mongod -f /etc/mongod.conf
102878	mongod	20	0	99G	91.8G	23948	S	2.7	72.9	0:00.32	/usr/bin/mongod -f /etc/mongod.conf
26107	mongod	20	0	99G	91.8G	23948	S	2.7	72.9	4h01:01	/usr/bin/mongod -f /etc/mongod.conf
102937	mongod	20	0	99G	91.8G	23948	S	2.0	72.9	0:00.09	/usr/bin/mongod -f /etc/mongod.conf
102821	mongod	20	0	99G	91.8G	23948	S	2.0	72.9	0:00.34	/usr/bin/mongod -f /etc/mongod.conf
102818	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:00.59	/usr/bin/mongod -f /etc/mongod.conf
102595	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:01.33	/usr/bin/mongod -f /etc/mongod.conf
102778	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:00.53	/usr/bin/mongod -f /etc/mongod.conf
102902	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:00.26	/usr/bin/mongod -f /etc/mongod.conf
102899	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:00.10	/usr/bin/mongod -f /etc/mongod.conf
102855	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:00.42	/usr/bin/mongod -f /etc/mongod.conf
102801	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:00.48	/usr/bin/mongod -f /etc/mongod.conf
102875	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:00.30	/usr/bin/mongod -f /etc/mongod.conf
102866	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:00.09	/usr/bin/mongod -f /etc/mongod.conf
102614	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:01.12	/usr/bin/mongod -f /etc/mongod.conf
102844	mongod	20	0	99G	91.8G	23948	S	1.3	72.9	0:00.53	/usr/bin/mongod -f /etc/mongod.conf

F1 Help F2 Setup F3 Search F4 Filter F5 Tree F6 Sort By F7 Nice F8 Vnice F9 Kill F10 Quit



■ Observar el estado del sistema

Las principales métricas que podemos observar con top son:

- Load Average
- Uso de CPU
- Uso de Memoria
- Uso de Swap
- Uso de recursos por los procesos



■ Observar el estado del sistema

Load Average

- Calculada a partir del número medio de procesos esperando a ser ejecutados por en cierto momento
- Ya que una CPU únicamente puede ejecutar un proceso esta métrica va relacionada al número de CPUs del sistema
- Un load average superior al número de CPUs indica que en algún elemento se está produciendo saturación, no necesariamente de CPU



■ Observar el estado del sistema

Uso de CPU

- top también muestra el porcentaje de uso de CPU
- las métricas que muestra son:

Indicador	Explicación
us	Porcentaje de tiempo que la CPU gasta gestionando procesos de usuario.
sy	Porcentaje de tiempo que la CPU gasta en tareas del kernel.
ni	Porcentaje de tiempo que el servidor gasta gestionando procesos en los que se ha cambiado la prioridad.
id	Porcentaje de tiempo que la CPU está libre
wa	Tiempo que el procesador está esperando por operaciones de I/O no interrumpible
hi	Tiempo que el procesador gasta gestionando interrupciones de hardware
si	Tiempo que el procesador gasta gestionando interrupciones software
st	Porcentaje de tiempo robado al hipervisor



■ Observar el estado del sistema

Uso de Memoria

- Viendo estas estadísticas se puede analizar si el servidor tiene suficiente RAM para ejecutar las tareas que tiene que ejecutar

Indicador	Explicación
KiB Mem	Tamaño de la memoria física en KB
used	Uso de memoria RAM para cualquier propósito
free	Memoria que no está en uso por nada
buffers	Memoria usada para almacenar datos desestructurados
cached Mem	Memoria usada para cachear ficheros que se han leído recientemente del disco

- Por el uso que hace el Kernel de caches y buffers, que un sistema tenga poca memoria libre no quiere decir que tenga problemas de rendimiento.



■ Observar el estado del sistema

Uso de Swap

- Cuando el sistema se queda sin memoria, el kernel puede hacer dos cosas:
 - Eliminar cachés: liberando memoria pero puede provocar que se tengan que leer de nuevo ficheros a disco
 - Usar swap: mueve páginas usadas por aplicaciones pero no usadas recientemente
- Si un sistema continuamente está moviendo páginas a swap tendremos problemas de rendimiento

```
j[redacted]:~$ swapon -s
```

Filename	Type	Size	Used	Priority
/dev/dm-1	partition	1949692	106384	-2



■ Observar el estado del sistema

Uso de recursos por los procesos

- Los procesos se ordenan por el uso de recursos, normalmente por CPU
- Con las teclas > y < se puede cambiar el parámetro de ordenación
- Presionando f se pueden añadir campos a la vista



■ Observar el estado del sistema

Otras herramientas

- iostat: muestra estadísticas de rendimiento de I/O
- mpstat: muestra el uso de CPU en un sistema con varias CPU
- pidstat: muestra estadísticas para un determinado PID
- vmstat: muestra información detallada sobre el uso de memoria



■ Observar el estado del sistema

iostat

- permite obtener información sobre qué dispositivo puede tener un uso intensivo de recursos
- Para cada dispositivo muestra el número de transacciones por segundo, cuántos kilobytes se han leído y escrito por segundo y cuántos kilobytes se han leído y escrito en total.

```
Linux 4.15.0-24-generic (lgmadcengans01v)      23/06/19      _x86_64_      (4 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0,99    0,00    0,32    0,02    0,00   98,68

Device:            tps    kB_read/s    kB_wrtn/s    kB_read    kB_wrtn
fd0                 0,00         0,00         0,00       1452         0
sda                 2,11        12,36        11,09    103707020    93098392
dm-0                1,36         8,71         5,25    73096441    44070872
dm-1                1,73         3,16         3,78    26530060    31698236
dm-2                0,14         0,01         0,63     47185     5251436
dm-3                0,23         0,48         1,44    4029289    12077848
```



■ Observar el estado del sistema

vmstat

- muestra información sobre el uso de memoria virtual dividida en varios grupos:
 - procs: número de procesos activos o esperando por IO
 - memory: parámetros de memoria
 - swap: actividad de swap
 - io: actividad de I/O
 - system: número de interrupciones y cambios de contexto
 - cpu: información de uso de CPU



■ Índice

- Troubleshooting
 - Observar el estado del sistema
 - Herramientas de monitorización
 - **Filesystem /proc**
 - Logs
 - Resumen herramientas
 - PCP + Vector



■ Observar el estado del sistema

/proc

- Este directorio ofrece una interfaz con el kernel y se puede utilizar para analizar el estado actual del kernel y hacer modificaciones
- la mayoría de herramientas leen información de este directorio

Fichero	Uso
meminfo	Información detallada del uso de memoria
cmdline	La línea usada para arrancar el sistema
cpuinfo	Información detallada del uso de CPU
partitions	Lista los dispositivos de almacenamiento y las particiones de esos dispositivos
sysrq-trigger	Permite enviar instrucciones de bajo nivel al kernel
modules	Lista los módulos de kernel actualmente activos



■ Observar el estado del sistema

/proc

- Para cada PID existe un subdirectorio con información detallada
- En el fichero status se puede obtener información detallada sobre el uso de memoria
- En el fichero environ se puede ver las variables de entorno que se usan por el proceso



■ Observar el estado del sistema

/proc

- En el directorio /proc/sys se puede modificar en caliente el comportamiento del kernel
- Para cambiar un parámetro se puede hacer de dos formas:
 - con *echo* para escribir el nuevo parámetro en el archivo
 - con *sysctl -w*
- Estos cambios no son persistentes, una vez confirmado que funciona se debe hacer también en los ficheros de configuración de *sysctl*



■ Observar el estado del sistema

/proc/sys

- Durante el arranque el servicio systemd-sysctl arranca
- Este servicio lee la configuración del sysctl de diferentes ficheros:
 - /etc/sysctl.conf: fichero por defecto gestionado por el sistema
 - /usr/lib/sysctl.d: directorio para optimizaciones por defecto gestionado a través de RPM
 - /etc/sysctl.d: directorio para personalizaciones del usuario
- Para ver todos los parámetros personalizables se puede usar el comando `sysctl -a`



■ Índice

- Troubleshooting
 - Observar el estado del sistema
 - Herramientas de monitorización
 - Filesystem /proc
 - **Logs**
 - Resumen herramientas
 - PCP + Vector



■ Observar el estado del sistema

Logs

- Por defecto los logs de todas las aplicaciones se guardan en /var/log
- Logs importantes:
 - syslog
 - messages (RH)
 - auth.log
 - wtmp
 - dmesg



■ Observar el estado del sistema

Journalctl

- Analiza los registros de systemd
- Por defecto se guardan en `/run/log/journal` o `/var/log/journal/`
- La configuración se realiza en `/etc/systemd/journald.conf`
- Los comandos más usuales son:

Comando	Descripción
<code>systemctl status systemd-journald</code>	Verificar el estado del servicio
<code>journalctl --disk-usage</code>	Analiza el espacio usado en logs
<code>journalctl -b n</code>	Registros desde el n-ésimo inicio de sesión
<code>journalctl --list-boots</code>	Lista los reinicios que ha tenido el sistema
<code>journalctl -f</code>	Salida continúa del registro
<code>journalctl -p err warning crit</code>	Devuelve registro de cierta criticidad



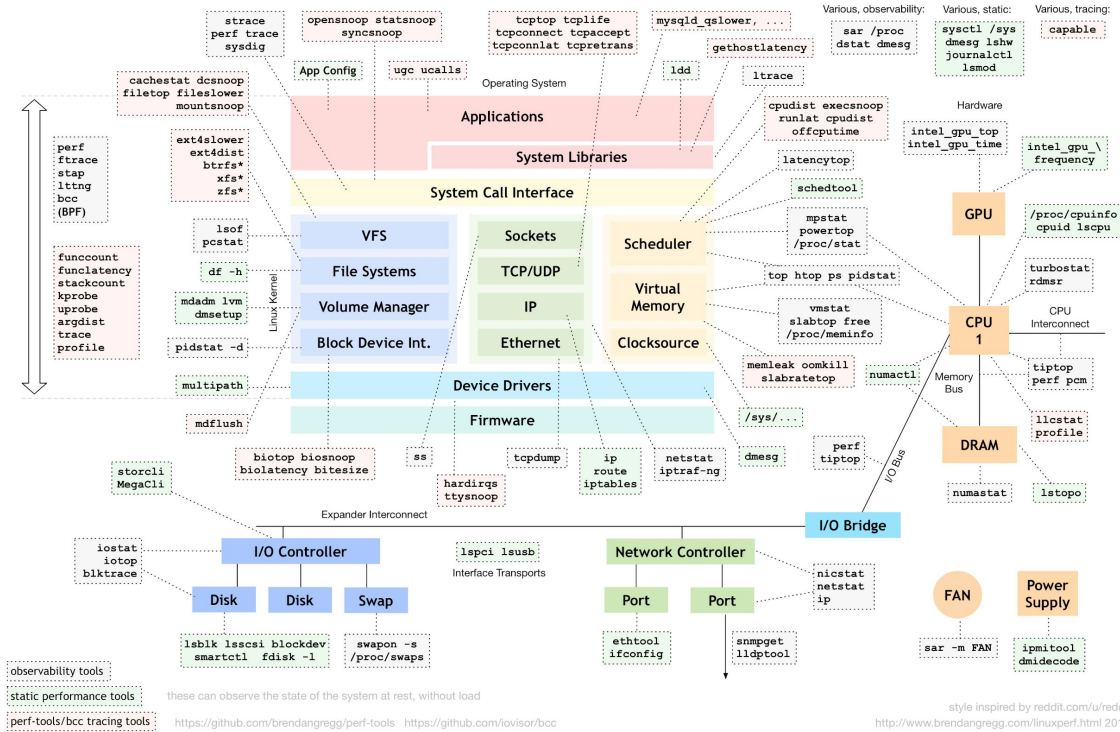
■ Índice

- Troubleshooting
 - Observar el estado del sistema
 - Herramientas de monitorización
 - Filesystem /proc
 - Logs
 - **Resumen herramientas**
 - PCP + Vector



Resumen de herramientas

Linux Performance Tools



■ Índice

- Troubleshooting
 - Observar el estado del sistema
 - Herramientas de monitorización
 - Filesystem /proc
 - Logs
 - Resumen herramientas
 - **PCP + Vector**



■ PCP y Vector

- PCP: Performance Co-Pilot → pcp.io
 - Herramienta para el análisis de rendimiento de sistemas
 - Recoge, monitoriza y análisis de métricas de sistema
 - Fácilmente extensible y flexible
 - Fácil de instalar
 - API accesible por otros componentes



■ PCP y Vector

- Vector → `getvector.io`
 - Herramienta de monitorización de rendimiento de hosts
 - Desarrollada inicialmente por Netflix y liberado como Open Source
 - Proporciona métricas con una resolución de hasta un segundo en tiempo real
 - Obtiene las métricas de la API de PCP



PCP y Vector





¿Preguntas?



GRACIAS

www.keepcoding.io

