
PRACTICA 2

Alumno: Adrián Bennasar Polzin | Grupo: Mañana

Características del sistema:

```
adrian@adrian-VirtualBox:/proc$ more cpuidinfo
processor       : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 142
model name     : Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz
stepping       : 9
cpu MHz        : 2711.998
cache size     : 3072 KB
physical id    : 0
siblings       : 4
core id        : 0
cpu cores      : 4
apicid         : 0
initial apicid : 0
fpu            : yes
fpu_exception  : yes
cpuid level    : 22
wp             : yes
flags          : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr
p_tsc cpuid tsc_known_freq pni pclmulqdq ssse3 cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx rd
hvpclid rdseed clflushopt flush_l1d
bugs           : cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass l1tf mds swapgs itlb_multihit
bogomips       : 5423.99
clflush size   : 64
cache_alignment : 64
address sizes  : 39 bits physical, 48 bits virtual
power management:
```

```
adrian@adrian-VirtualBox:/proc$ more meminfo
MemTotal:      4030252 kB
MemFree:       1008000 kB
MemAvailable:  1927992 kB
Buffers:       107120 kB
Cached:        1077556 kB
SwapCached:    0 kB
Active:        1989252 kB
Inactive:      759808 kB
Active(anon):  1566124 kB
Inactive(anon): 108172 kB
Active(file):  423128 kB
Inactive(file): 651636 kB
Unevictable:   16 kB
Mlocked:       16 kB
SwapTotal:     2097148 kB
SwapFree:      2097148 kB
Dirty:         960 kB
Writeback:     0 kB
AnonPages:     1564504 kB
Mapped:        439976 kB
Shmem:         109916 kB
KReclaimable:  96596 kB
Slab:          137100 kB
SReclaimable:  96596 kB
SUnreclaim:    40504 kB
KernelStack:   12256 kB
PageTables:    48644 kB
NFS_Unstable:  0 kB
Bounce:        0 kB
WritebackTmp:  0 kB
CommitLimit:   4112272 kB
Committed_AS:  6205832 kB
VmallocTotal:  34359738367 kB
VmallocUsed:    36568 kB
VmallocChunk:   0 kB
Percpu:        1296 kB
HardwareCorrupted: 0 kB
AnonHugePages: 0 kB
ShmemHugePages: 0 kB
ShmemPmdMapped: 0 kB
CmaTotal:      0 kB
CmaFree:       0 kB
HugePages_Total: 0
HugePages_Free: 0
```

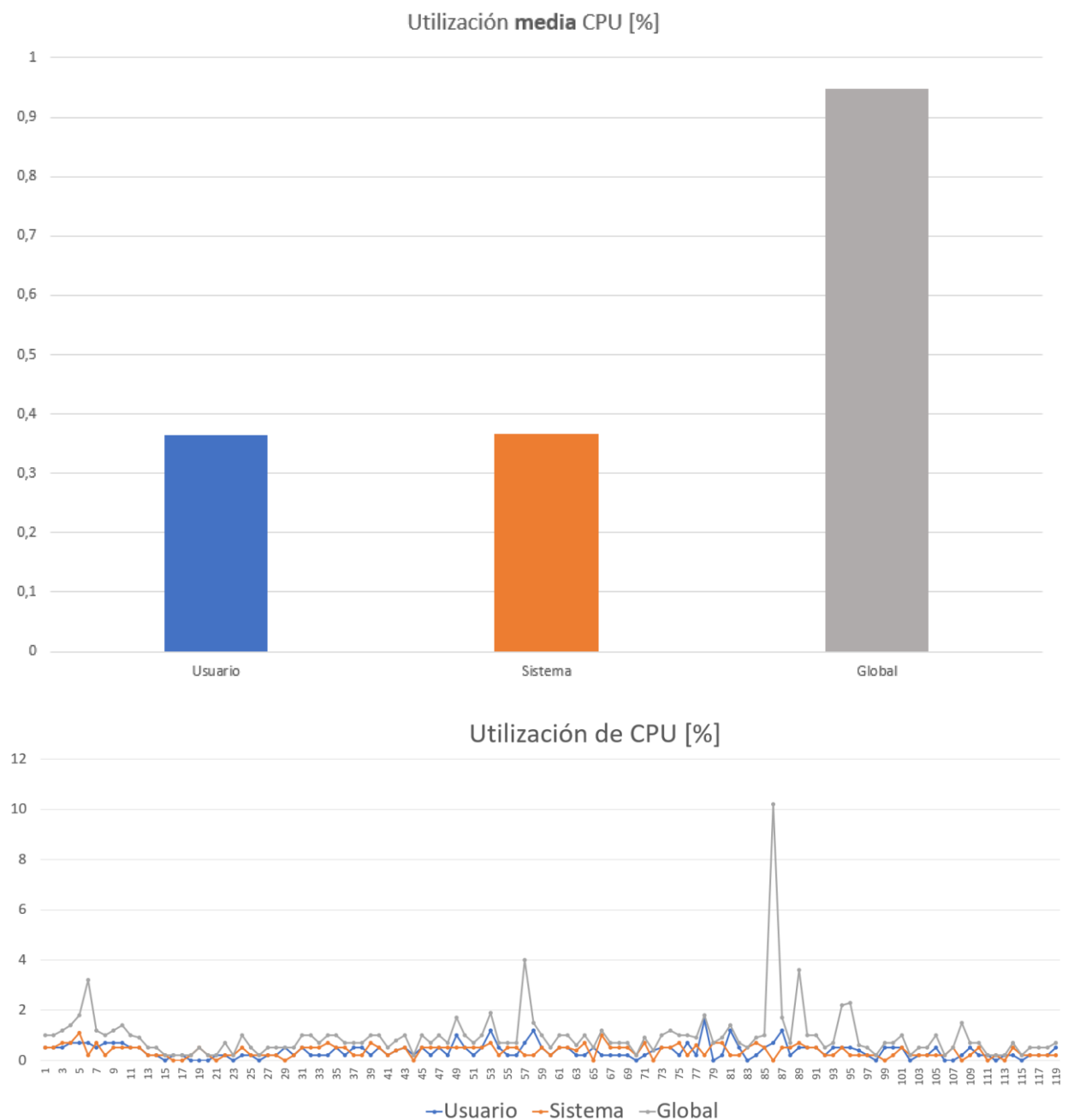
Actividad 1

Para la actividad 1 he ejecutado el comando **top** durante 2 minutos y he redireccionado el output a un fichero de texto de la siguiente manera:

top -b -d1 -n120|grep -i "%Cpu(s)" >> top.txt

Observando los resultados de la ejecución de este comando podemos calcular la utilización media en modo usuario, sistema y global de la CPU durante estos 2 minutos.

****La primera fila del output no la he tenido en cuenta como se indicó en clase, ya que sale muy desproporcionada al resto de filas. Con estos resultados he construido las siguientes gráficas:**



Para la utilización global he hecho el cálculo **100 – iddle**.

Para calcular la sobrecarga del monitor hay que dividir el tiempo de ejecución del monitor para una muestra entre el intervalo de muestreo. Para obtener el tiempo de ejecución del monitor para una muestra he usado el comando **time** de la siguiente manera:

time top -b -d1 -n1|grep -i “%Cpu(s)”

Para obtener un valor lo más real posible, he ejecutado esta instrucción 5 veces y he hecho la media, utilizando el valor que sale a la derecha de la palabra **real**

Tiempo de ejecución: (0,193s + 0,196s + 0,194s + 0,195s + 0,198s) / 5 = 0.1952s.

$S = 0.1952/1 = 0.1952, *100 = \mathbf{19.52\% \text{ sobrecarga.}}$

Esto es una sobrecarga muy elevada y la manera de reducirla sería aumentar el tiempo que se espera entre cada muestra es decir aumentar el intervalo de ejecución. Por ejemplo, aumentarlo de 1 segundo a 10 segundos.

```
adrian@adrian-VirtualBox:~/Desktop/ACSIC$ time top -b -d1 -n1|grep -i "%Cpu(s)"
%Cpu(s):  3,8 us,  1,4 sy,  0,1 ni, 88,0 id,  6,5 wa,  0,0 hi,  0,2 si,  0,0 st

real    0m0,193s
user    0m0,011s
sys     0m0,032s
adrian@adrian-VirtualBox:~/Desktop/ACSIC$ time top -b -d1 -n1|grep -i "%Cpu(s)"
%Cpu(s):  3,7 us,  1,4 sy,  0,1 ni, 88,1 id,  6,5 wa,  0,0 hi,  0,2 si,  0,0 st

real    0m0,196s
user    0m0,021s
sys     0m0,024s
adrian@adrian-VirtualBox:~/Desktop/ACSIC$ time top -b -d1 -n1|grep -i "%Cpu(s)"
%Cpu(s):  3,7 us,  1,3 sy,  0,1 ni, 88,2 id,  6,4 wa,  0,0 hi,  0,2 si,  0,0 st

real    0m0,194s
user    0m0,027s
sys     0m0,017s
adrian@adrian-VirtualBox:~/Desktop/ACSIC$ time top -b -d1 -n1|grep -i "%Cpu(s)"
%Cpu(s):  3,7 us,  1,3 sy,  0,1 ni, 88,3 id,  6,4 wa,  0,0 hi,  0,2 si,  0,0 st

real    0m0,195s
user    0m0,010s
sys     0m0,032s
adrian@adrian-VirtualBox:~/Desktop/ACSIC$ time top -b -d1 -n1|grep -i "%Cpu(s)"
%Cpu(s):  3,6 us,  1,3 sy,  0,1 ni, 88,4 id,  6,3 wa,  0,0 hi,  0,2 si,  0,0 st

real    0m0,198s
user    0m0,027s
sys     0m0,017s
adrian@adrian-VirtualBox:~/Desktop/ACSIC$ time top -b -d1 -n1|grep -i "%Cpu(s)"
%Cpu(s):  2,6 us,  0,9 sy,  0,1 ni, 91,8 id,  4,4 wa,  0,0 hi,  0,2 si,  0,0 st
```

Actividad 2

Para la actividad 2 de lo primero que hay que asegurarse es que el monitor vmstat y el directorio /proc meminfo trabajan con las mismas unidades. No he encontrado una manera de averiguar con qué unidades trabaja el vmstat **por defecto**, pero el /proc meminfo me da la información en kB(1000 bytes). Por lo tanto mi solución ha sido obligar al vmstat a trabajar con 1000 bytes también, y esto lo he hecho aplicando el parámetro **-S k**.

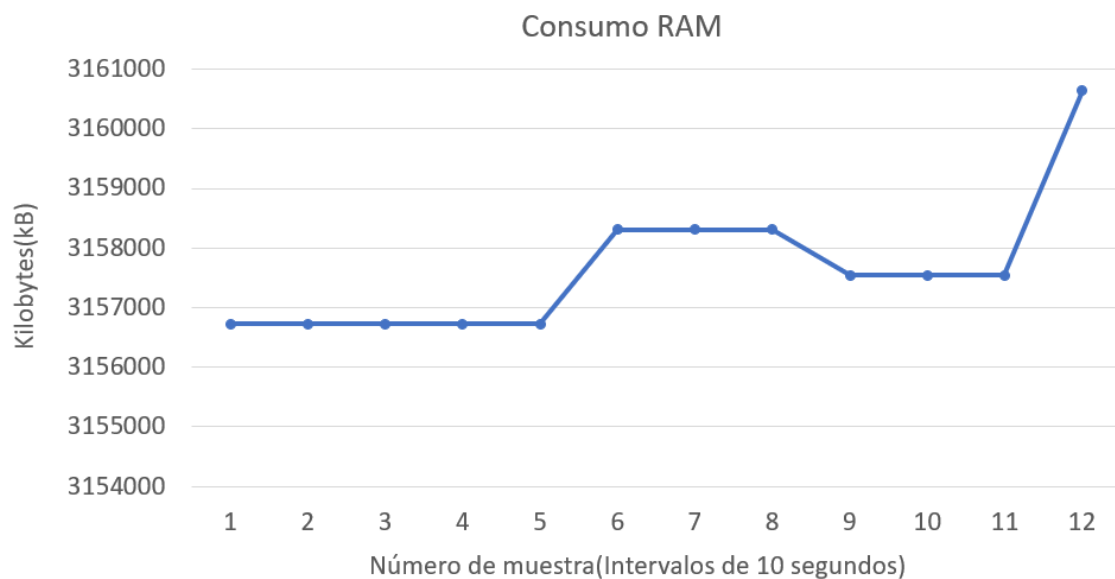
El comando completo que he usado para monitorizar la utilización de la memoria principal durante 2 minutos es:

vmstat -S k 10 12|awk '{print \$1, \$2, \$3, \$4, \$5, \$6, \$7, \$8}' El parámetro awk '{print \$1, \$2, \$3, \$4, \$5, \$6, \$7, \$8}' es para obtener solo la información útil de la memoria que pide la actividad 2, es decir, las 8 primeras columnas del monitor vmstat.

Para calcular la utilización media he realizado el siguiente cálculo:

$[(\text{MemTotal} - (\text{valor medio free}) / \text{MemTotal}) * 100]$, donde MemTotal es el valor obtenido de /proc meminfo que informa de la memoria principal total del sistema.

Por tanto: $[(4030252\text{kB} - 872599,3333\text{kB}) / 4030252\text{kB}] = 78.35\% \text{ RAM}$



Para la sobrecarga se hace igual que en la actividad 1:

He ejecutado la siguiente instrucción 5 veces y he hecho la media del campo **real** para obtener el tiempo de ejecución del monitor vmstat para una muestra:

time vmstat 1 1|awk '{print \$1 \$2 \$3 \$4 \$5 \$6 \$7 \$8}'

El valor obtenido de hacer la media lo divido entre el intervalo de muestreo que es cada 10 segundos, por tanto: $S = 0.0034 / 10 = 0.00034$, * 100 = **0.034 % sobrecarga**. Esto es una sobrecarga muy conveniente ya que prácticamente no tiene impacto en el sistema.

```
adrian@adrian-VirtualBox:~$ time vmstat 1 1|awk '{print $1 , $2, $3, $4, $5, $6, $7, $8}'
procs -----memory----- ---swap-- ----io---- -system-- -----cpu-----
r b swpd free buff cache si so
1 0 0 754660 113924 1213544 0 0

real    0m0,004s
user    0m0,004s
sys     0m0,001s
adrian@adrian-VirtualBox:~$ time vmstat 1 1|awk '{print $1 , $2, $3, $4, $5, $6, $7, $8}'
procs -----memory----- ---swap-- ----io---- -system-- -----cpu-----
r b swpd free buff cache si so
0 0 0 752164 113924 1213488 0 0

real    0m0,003s
user    0m0,003s
sys     0m0,000s
adrian@adrian-VirtualBox:~$ time vmstat 1 1|awk '{print $1 , $2, $3, $4, $5, $6, $7, $8}'
procs -----memory----- ---swap-- ----io---- -system-- -----cpu-----
r b swpd free buff cache si so
1 0 0 748508 113932 1213488 0 0

real    0m0,004s
user    0m0,001s
sys     0m0,004s
adrian@adrian-VirtualBox:~$ time vmstat 1 1|awk '{print $1 , $2, $3, $4, $5, $6, $7, $8}'
procs -----memory----- ---swap-- ----io---- -system-- -----cpu-----
r b swpd free buff cache si so
0 0 0 747872 113932 1213488 0 0

real    0m0,003s
user    0m0,004s
sys     0m0,000s
adrian@adrian-VirtualBox:~$ time vmstat 1 1|awk '{print $1 , $2, $3, $4, $5, $6, $7, $8}'
procs -----memory----- ---swap-- ----io---- -system-- -----cpu-----
r b swpd free buff cache si so
2 0 0 744388 113932 1213488 0 0

real    0m0,003s
user    0m0,004s
sys     0m0,000s
adrian@adrian-VirtualBox:~$
```

Actividad 3

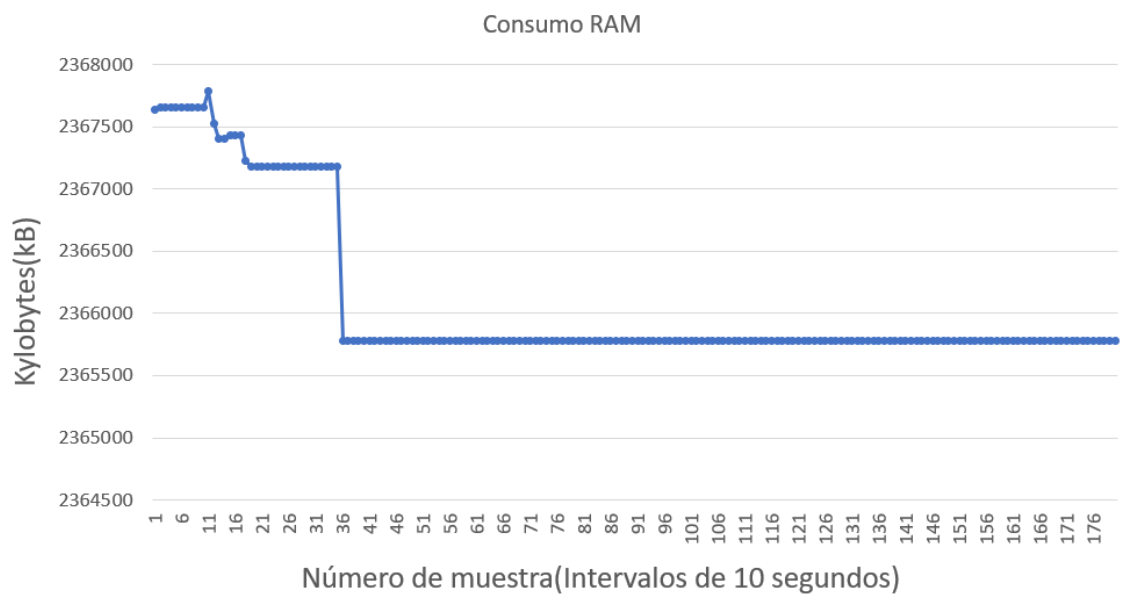
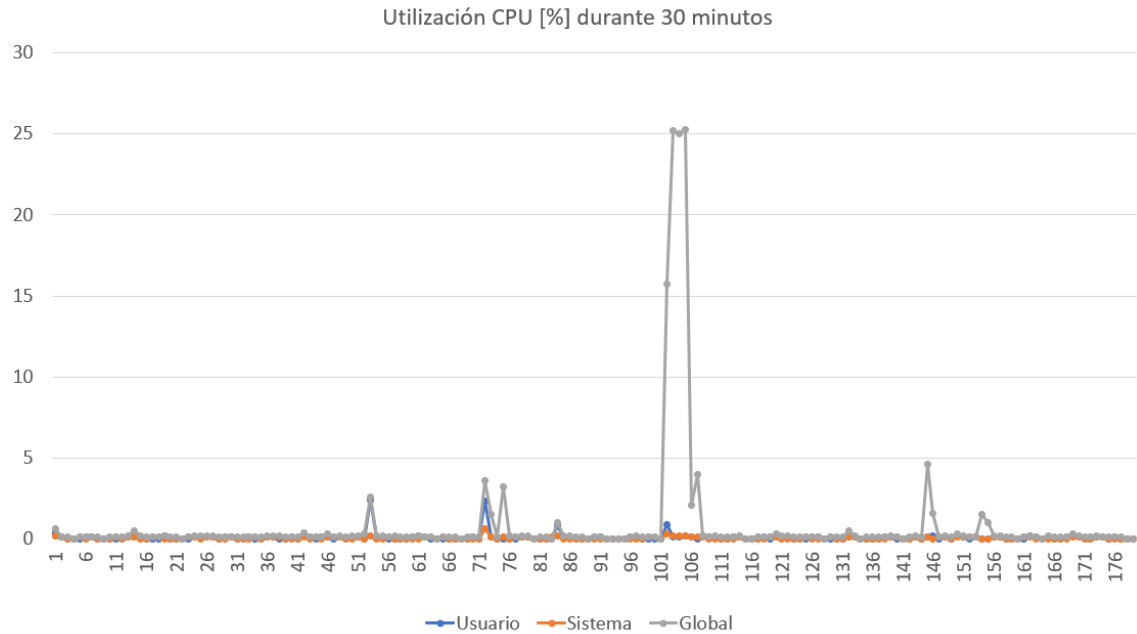
El monitor que he considerado más adecuado para determinar la utilización de la CPU en modo usuario, sistema y en global es el **top**.

El monitor que he considerado más adecuado para determinar la capacidad y utilización de la memoria principal es el monitor **vmstat**, combinado con la información del directorio **/proc**.

Para que los monitores top y vmstat se ejecuten durante media hora (30 minutos) con un intervalo de muestreo de 10 segundos he utilizado las siguientes instrucciones:

- **top -b -d10 -n180|grep -i “%Cpus(s)”**
- **vmstat 10 180|awk ‘{\$1, \$2, \$3, \$4, \$5, \$6, \$7, \$8}’**

Los resultados obtenidos, mostrados de manera gráfica, son los siguientes:



La utilización media de la CPU, calculada mediante la media aritmética, es la siguiente:

- modo usuario: 0.0863 %
- modo sistema: 0.0469 %
- global: 0.785 %

La utilización media de la memoria principal es:

$$[(4030252\text{kB} - 1664160.58) / 4030252\text{kB}] = 58.71 \% \text{ RAM}$$

Finalmente, la sobrecarga que ejercen estos 2 monitores durante la media hora de su ejecución es:

- Monitor top: $S = 0.1952/10 = 0.01952$, $*100 = \mathbf{1.95 \% \text{ sobrecarga.}}$
- Monitor vmstat: $S = 0.0034 / 10 = 0.00034$, $* 100 = \mathbf{0.034 \% \text{ sobrecarga.}}$

Como se puede observar en el gráfico de monitorización de CPU, durante los 30 minutos la utilización se mantiene más o menos igual excepto en las muestras 101-106 que experimenta un pico muy notable. Esto puede ser debido a varias causas, por ejemplo, que un proceso en segundo plano de elevada carga haya empezado a trabajar en ese momento y después haya terminado causando que el pico desaparezca.

En la gráfica de monitorización de memoria principal podemos observar otro cambio brusco, pero esta vez es una bajada en lugar de una subida de utilización. Esto puede ser por motivos parecidos a lo comentado en el párrafo anterior, pero al revés, es decir que un proceso que estaba ocupando memoria haya terminado y haya liberado los recursos.

En cuanto a las sobrecargas podemos observar que la del top es mucho mas baja que la que provocaba en la actividad 1 (**19.52%**) ya que en lugar de ejecutarlo con un intervalo de 1 segundo se ha ejecutado con uno intervalo de 10 segundos.