

1. Monitorización en Linux (/proc) - CentOS y Ubuntu

Cuestión 1. Inspeccione algunos de los archivos y directorios anteriores, indique qué representa cada uno y realice un resumen de la actividad actual de su sistema Ubuntu y CentOS.

cpuinfo → muestra la información de la cpu.
uptime → tiempo que lleva encendida la máquina, usuarios conectados y carga.
mounts → sistemas de ficheros montados.
vmstat → estadísticas de memoria virtual.
swaps → detalles de memoria swap.
version → información de la versión.
hostname → nombre del host.

2. Monitorización en Linux (comandos Linux) - CentOS y Ubuntu

Cuestión 2. Pruebe cada alguno de los comandos anteriores y amplíe la información mostrada usando distintos parámetros admitidos por cada comando. Se deja a criterio del alumno qué opciones deberá utilizar en cada orden.

df → muestra el espacio de disco usado por el sistema de ficheros.
du → muestra la suma del espacio usado por cada archivo y todo su contenido.
free → muestra el el espacio libre de la memoria física y swap.
ps → información de los procesos actuales.
who → información sobre los usuarios autenticados.
finger → información sobre los usuarios del sistema.

3. Utilice la orden vmsat para medir la actividad del sistema durante un total de cinco minutos. El periodo entre medidas consecutivas será de 5 segundos. La información se guardará en el fichero de texto vmstat.res.

`vmstat -t 5 60 >> /home/i32moviv/vmstat.res`

3. Monitorización en Linux (sar) - CentOS y Ubuntu

Cuestión 4. Indique las distintas opciones que dispone sar así como una descripción de cada una de ellas.

OPTIONS

-A This is equivalent to specifying -bBdFHqrRSuvvWy -I SUM -I XALL -m ALL -n ALL -u ALL -P ALL.

-B Report paging statistics. The following values are displayed:

-b Report I/O and transfer rate statistics. The following values are displayed:

-C When reading data from a file, tell **sar** to display comments that have been inserted by **sadc**.

-d Report activity for each block device. When data are displayed, the device specification dev m-n is generally used (**DEV** column). **m** is the major number of the device and **n** its minor number. Device names may also be pretty-printed if option **-p** is used or persistent device names can be printed if option **-j** is used (see below). Note that disk activity depends on **sadc** options **-S DISK** and **-S XDISK** to be collected. The following values are displayed:

-f [filename]
Extract records from filename (created by the **-o filename** flag). The default value of the **filename** parameter is the current daily data file, the /var/log/sysstat/sadd file. The **-f** option is exclusive of the **-o** option.

-H Report hugepages utilization statistics. The following values are displayed:

-h Display a short help message then exit.

-I { int [,...] | SUM | ALL | XALL }
Report statistics for a given interrupt. **int** is the interrupt number. Specifying multiple **-I int** parameters on the command line will look at multiple independent interrupts. The **SUM** keyword indicates that the total number of interrupts received per second is to be displayed. The **ALL** keyword indicates that statistics from the first 16 interrupts are to be reported, whereas the **XALL** keyword indicates that statistics from all interrupts, including potential APIC interrupt sources, are to be reported. Note that interrupt statistics depend on **sadc** option **-S INT** to be collected.

-i interval
Select data records at seconds as close as possible to the number specified by the interval parameter.

-j { ID | LABEL | PATH | UUID | ... }
Display persistent device names. Use this option in conjunction with option **-d**. Options **ID**, **LABEL**, etc. specify the type of the persistent name. These options are not limited, only prerequisite is that directory with required persistent names is present in /dev/disk. If persistent name is not found for the device, the device name is pretty-printed (see option **-p** below).

-m { keyword [,...] | ALL }
Report power management statistics. Note that these statistics depend on **sadc** option **-S POWER** to be collected.

-n { keyword [,...] | ALL }
Report network statistics.


```

-o [ filename ]
    Save the readings in the file in binary form. Each reading is in
    a separate record. The default value of the filename parameter
    is the current daily data file, the /var/log/sysstat/sadd file.
    The -o option is exclusive of the -f option. All the data
    available from the kernel are saved in the file (in fact, sar
    calls its data collector sadc with the option -S ALL. See
    sadc(8) manual page).

-P { cpu [... ] | ALL }
    Report per-processor statistics for the specified processor or
    processors. Specifying the ALL keyword reports statistics for
    each individual processor, and globally for all processors.
    Note that processor 0 is the first processor.

-p
    Pretty-print device names. Use this option in conjunction with
    option -d. By default names are printed as dev m-n where m and
    n are the major and minor numbers for the device. Use of this
    option displays the names of the devices as they (should) appear
    in /dev. Name mappings are controlled by /etc/sysstat/sys-
    stat.ioconf.

-q
    Report queue length and load averages. The following values are
    displayed:

-R
    Report memory statistics. The following values are displayed:

-r
    Report memory utilization statistics. The following values are
    displayed:

-S
    Report swap space utilization statistics. The following values
    are displayed:

-s [ hh:mm:ss ]
    Set the starting time of the data, causing the sar command to
    extract records time-tagged at, or following, the time speci-
    fied. The default starting time is 08:00:00. Hours must be
    given in 24-hour format. This option can be used only when data
    are read from a file (option -f).

-t
    When reading data from a daily data file, indicate that sar
    should display the timestamps in the original local time of the
    data file creator. Without this option, the sar command displays
    the timestamps in the user's locale time.

-u [ ALL ]
    Report CPU utilization. The ALL keyword indicates that all the
    CPU fields should be displayed. The report may show the follow-
    ing fields:

-V
    Print version number then exit.

-v
    Report status of inode, file and other kernel tables. The fol-
    lowing values are displayed:
  
```

```
-W      Report swapping statistics. The following values are displayed:
```

```
-w      Report task creation and system switching activity.
```

```
-y      Report TTY devices activity. The following values are displayed:
```

*He omitido los valores que muestra cada comando, pero muestro que tiene con "The following values are displayed:", pueden consultarse fácilmente con man.

Cuestión 5. Escoja uno de los ficheros históricos de sar (/var/log/sa/saDD) disponibles en el sistema y analice el comportamiento de un día entero de los siguientes aspectos:

Escojo el fichero histórico /var/log/sa/sa19

● **Utilización del procesador (modo usuario, sistema y desocupado)**

```
sar -u -f /var/log/sa/sa19
```

● **Carga media del sistema (1, 5, 15 últimos minutos)**

```
sar -q -f /var/log/sa/sa19
```

● **Paginación**

```
sar -B -f /var/log/sa/sa19
```

● **Cambios de contexto**

```
sar -w -f /var/log/sa/sa19
```

Cuestión 6. Muestre la secuencia de comandos para realizar la instalación del paquete sysstat en Ubuntu y la configuración realizada para habilitar sar para que se ejecute en cada 10 minutos.

```
apt-get install sysstat
```

y cambiamos en /etc/default/sysstat:

ENABLE="false" por ENABLE="true" para que lo haga automáticamente.

Creemos ahora un crontab para que que sar se ejecute cada 10 minutos con el siguiente formato:

1-59/10 * * * * para que se ejecute todas las horas de todos los días de todos los meses de todos los años cada 10 minutos.

y otro con:

0 * * * * para que a en punto también se haga.

Cuestión 7. Ejecute sar en modo interactivo durante 5 minutos con una frecuencia de 30 segundos. Muestre el comando utilizado y analice el resultado.

```
sar -t 30 10
```

```
Linux 3.10.0-229.el7.x86_64 (localhost.localdomain) 11/12/15 _x86_64_ (1 CPU)
06:09:38      CPU      %user      %nice      %system      %iowait      %steal      %idle
06:10:08    all      10,98       0,00       2,31       0,10       0,00      86,60
06:10:38    all       5,29       0,00       0,64       0,10       0,00      93,97
06:11:08    all       5,39       0,00       0,68       0,14       0,00      93,79
06:11:38    all      13,26       0,00       1,68       0,13       0,00      84,92
06:12:08    all       5,36       0,00       0,74       0,13       0,00      93,76
06:12:38    all       5,28       0,00       0,57       0,13       0,00      94,01
06:13:08    all       4,73       0,00       0,77       0,13       0,00      94,36
06:13:38    all       6,19       0,00       0,87       0,10       0,00      92,84
06:14:08    all       4,82       0,00       0,70       0,17       0,00      94,31
06:14:38    all       5,12       0,00       0,57       0,13       0,00      94,18
Media:      all       6,64       0,00       0,95       0,13       0,00      92,27
```

4. Monitorización en Linux (otros monitores)

Cuestión 8. Enumere y muestre las características más importantes de otras herramientas de monitorización para Linux (al menos 3).

top: información sobre los procesos actuales. Esta información se actualiza automáticamente.

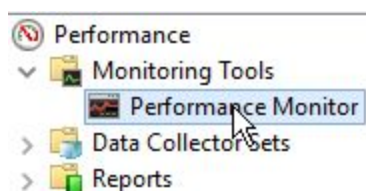
ps: información sobre los procesos actuales.

netstat: muestra el tráfico de red.

5. Monitorización en Windows (perfmon)

Cuestión 9. Configure el monitor de rendimiento para supervisar durante 5 minutos el estado del % de tiempo del procesador y de usuario, % de tiempo de lectura, escritura e inactividad del disco duro, errores de caché y MB disponibles en memoria. Muestre una gráfica y analice el resultado de los datos recogidos tras una ejecución.

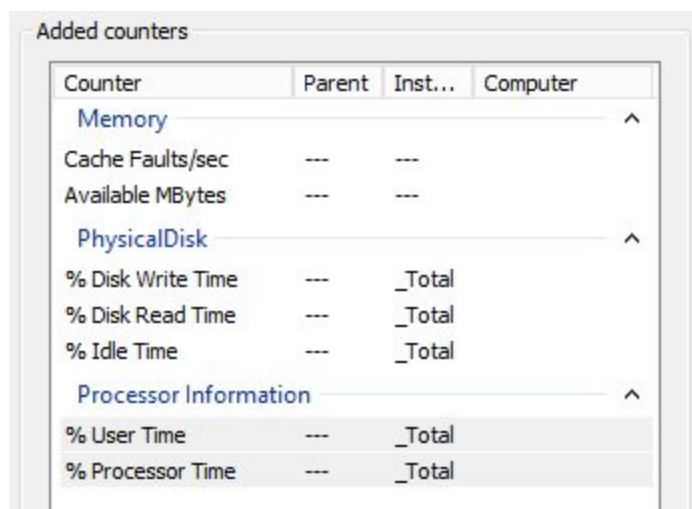
Entramos en la aplicación Performance Monitor.



Seleccionamos Performance Monitor.



Pulsamos el símbolo “+” verde para añadir uno nuevo.



Añadimos los siguientes counters.

☒ Sample automatically

Graph elements

Sample every

1

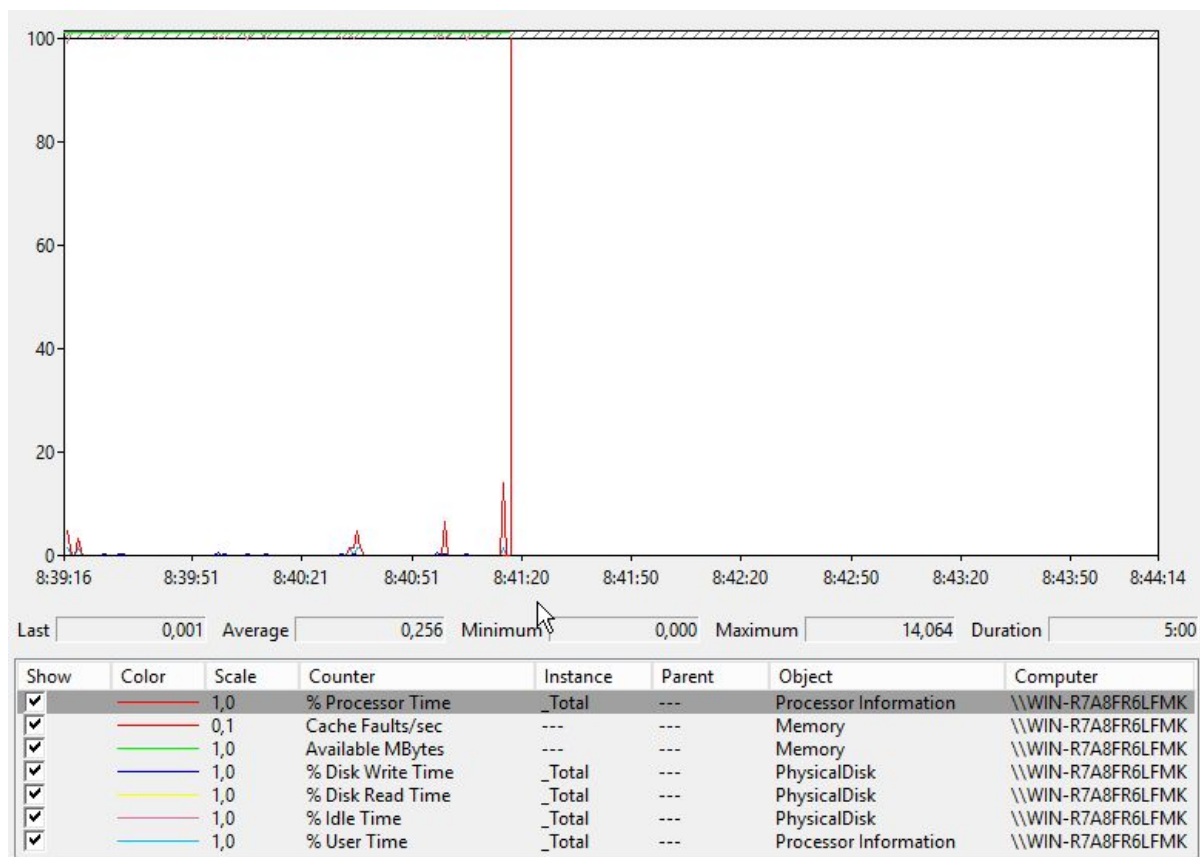
seconds

Duration:

300

seconds

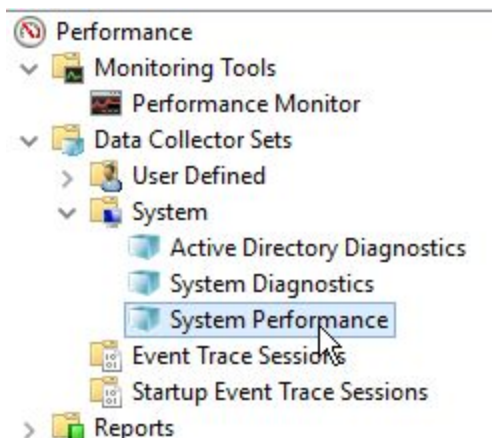
Seleccionamos con botón secundario Performance Monitor/Properties y establecemos la duración a 300 segundos, es decir, 5 minutos. Aplicamos y aceptamos.



Y podremos observar que se monitorizará durante 5 minutos lo que hemos configurado.

Cuestión 10. Ejecute el recopilador de datos del sistema configurado para el Rendimiento del Sistema y muestre el resultado del informe tras la ejecución.

Entramos en la aplicación Performance Monitor.



Seleccionamos Data Collector Sets/System/System Performance.



Pulsamos el botón “start”.

Name	Type	Output
NT Kernel	Trace	C:\perlogs\System\Performance\WIN-R7A8FR6LFMK_20151211-000004...
Performance Counter	Performance Counter	C:\perlogs\System\Performance\WIN-R7A8FR6LFMK_20151211-000004...

Vamos a la dirección de salida que se nos muestra “output”.

Name	Date modified	Type	Size
NtKernel.etl	11/12/2015 4:15	ETL File	320 KB
Performance Counter	11/12/2015 4:15	Performance Mon...	2.048 KB
report	11/12/2015 4:15	HTML File	684 KB
report	11/12/2015 4:15	XML Document	633 KB
report	11/12/2015 4:15	XSL Stylesheet	62 KB
rules	11/12/2015 4:15	Text Document	604 KB

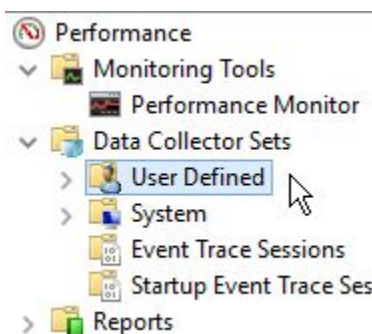
Aquí se encuentra el informe que nos ha generado el recopilador de datos, en varios formatos (HTML, XML, XSL).

Cuestión 11. Cree un recopilador de datos de un periodo de 5 minutos definido por el usuario (modo avanzado) que incluya tanto el contador de rendimiento como los datos de seguimiento:

- Todos los referentes al procesador, al proceso y al servicio web.
- Intervalo de muestra 15 segundos.
- Almacene el resultado en el directorio home del usuario practica4.

Incluya las capturas de pantalla de cada paso y el resultado de una ejecución.

Entramos en la aplicación Performance Monitor.



Seleccionamos Data Collector Sets/User Defined.



Seleccionamos el icono anterior para crear un nuevo recopilador de datos.

Name:

☐ Create from a template (Recommended)

☒ Create manually (Advanced)

Le damos nombre y seleccionamos las opciones avanzadas.

What type of data do you want to include?

☒ Create data logs

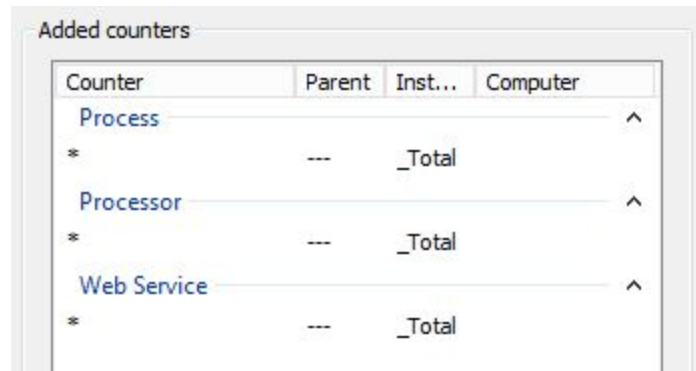
☒ Performance counter

☒ Event trace data

☐ System configuration information

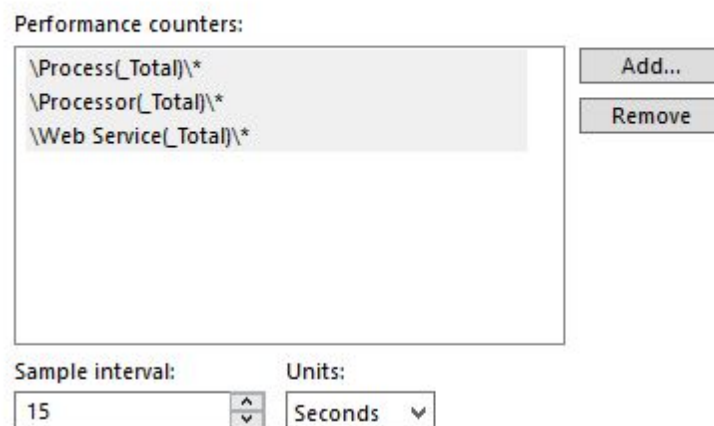
☐ Performance Counter Alert

Seleccionamos las opciones anteriores.



Seleccionamos los servicios Process, Processor y Web Service con todas sus instancias.

Which performance counters would you like to log?



Elegimos 15 segundos de intervalo.

Which event trace providers would you like to enable?

Providers:

Properties:

Property	Value

Saltamos al siguiente paso.

Where would you like the data to be saved?

Root directory:

C:\Users\j32moviv\practica4

Elegimos la ruta donde queremos guardar y finalizamos.

Name	Status
practica4	Stopped
Server Manager Performance M...	Stopped

Y ya quedará creado.

6. Monitorización en Windows (otros monitores)

Cuestión 12. Enumere y muestre las características más importantes de otras herramientas de monitorización para Windows (al menos 3).

OverClock Checking Tool: para temperaturas, voltajes, uso de CPU, GPU y fuente de alimentación. Además mide la velocidad del ventilador.

VoIP: comprobar factores como la pérdida de paquetes, el tiempo que tarda en la ida y vuelta y la variación en el retraso de paquetes.

Open Hardware Monitor: para medir la temperatura de los componentes.



7. Cuestiones propias.

Cuestión 13. ¿Qué significan los dígitos de los ficheros históricos de sar, por ejemplo /var/log/sa/sa19?

Los dígitos de los ficheros históricos de sar significan el día del mes que se realizó el fichero histórico en concreto.

Cuestión 14. ¿Qué significa que una herramienta de monitorización sea dinámica?

Que una herramienta de monitorización sea dinámica significa que los datos que nos monitoriza se actualizan al momento y no se monitoriza el momento de ejecución.