**1. Gestión de recursos del sistema**

**Esta parte del laboratorio propone unas tareas para trabajar con los comandos relacionados con la monitorización**

**y gestión de los recursos del sistema.**

**1) Obtener el número de procesos en ejecución en el sistema.**

ps -e --no-headers | wc -l

**2) Obtener el número de procesos en ejecución que pertenezcan a usuario root.**

ps -u root --no-headers | wc -l

**3) Instalar el paquete “stress-ng”. Ejecutar el benchmark para 1 núcleo de CPU y 20 segundos.**

sudo apt update

sudo apt install -y stress-ng

stress-ng --cpu 1 --timeout 20s

**4) Ejecutar de nuevo “stress-ng”, esta vez sin límite de tiempo. Mientras esté en ejecución:**

$ stress-ng --cpu 1 &

* Con el &, hace en segundo plano.

stress-ng: info: [7037] defaulting to a 86400 second (1 day, 0.00 secs) run per stressor

stress-ng: info: [7037] dispatching hogs: 1 cpu

**a. Usar una señal para pausar su ejecución.**

kill -STOP 7037

* Para eliminar del todo cambiar -STOP por -KILL

**b. Usar una señal para reanudar su ejecución.**

kill -CONT 7037

**c. Reducir la prioridad del proceso al mínimo. ¿Cambia algo?**

sudo renice +19 -p 7037

# Ver prioridad y estado

ps -o pid,ni,pri,cmd -p 7037

**5) Detectar qué proceso tiene la mayor prioridad en el sistema. Buscar cuál es el propósito de ese proceso.**

ps -eo pid,ni,pri,cmd --sort=ni | head -n 5

**6) Limitar el máximo tiempo de uso de CPU a 5 minutos para todos los usuarios.**

$ ulimit -t 300

* 300 → Segundos

$ sudo vim /etc/security/limits.conf

Añadir al final:

\* hard cpu 5

**7) Crear un fichero crontab para el usuario root con las siguientes tareas:**

**a. Ejecutar el comando date cada minuto y escribir su salida estándar al fichero /tmp/date.log (se debe escribir al final del fichero cada vez).**

$ crontab -e

* Opción 2.

Añadir:

\* \* \* \* \* date >> /tmp/date.log

**b. Borrar el directorio /tmp los primeros 5 días de cada mes a las 17:00.**

Añadir:

0 17 1-5 \* \* rm -rf /tmp/\*

**8) Comprobar que las tareas de cron funcionan correctamente (ver el fichero /tmp/date.log).**

$ cat /tmp/date.log

**Las siguientes tareas del laboratorio son relativas a las conexiones de red y se propone trabajar en parejas. Seréis Ay B, y cada uno utilizará su propia máquina virtual:**

**9) Usando netcat, A abre una conexión a la escucha en el puerto 3000. Si fuese necesario abrir puertos de Google Cloud, se recomienda este tutorial1. B se conecta a ese puerto y escribe el mensaje “Hola A”. Tras recibirlo, en la misma conexión, B escribe “Hola B” y cierra la conexión.**

HAY QUE ABRIR EL PUERTO 3000 CON FIREWALL

$ sudo apt install netcat-openbsd

**A escucha puerto 3000:**

$ nc -l -p 3000

**B envía mensaje:**

echo "Hola A" | nc 34.51.172.172 3000

**10) El comando “dd if=/dev/urandom” sirve para generar números aleatorios. A abre una conexión a la escucha en el puerto 3000 y B envía números aleatorios a esa conexión de A. Sin cerrar la conexión, A y B utilizan nethogs para comprobar la tasa de bytes enviados y recibidos en cada parte. ¿Los valores coinciden? Después, A cierra la conexión.**

**A escucha y recibe datos:**

nc -l -p 3000 > /dev/null

**B envía datos aleatorios:**

dd if=/dev/urandom bs=1M count=100 | nc 34.51.172.172 3000

**Monitorizar tráfico con nethogs:**

sudo apt install nethogs

sudo nethogs

**11) A crea localmente un fichero con texto aleatorio. Utilizar netcat para que A envíe este fichero a B. ¿Qué diferencia hay entre utilizar netcat y scp para enviar ficheros entre diferentes máquinas?**

**A:**

nc -l -p 3000 > archivo\_recibido.txt

**B:**

nc 34.51.172.172 3000 < archivo\_local.txt

**SCP:**

scp archivo\_local.txt usuario@IP\_A:/ruta/destino/

* Es más seguro, cifrado, y con autenticación.

**2. Gestión de los registros del sistema**

**En esta parte del laboratorio se propone trabajar con los registros (logs) del sistema.**

**1) Leer la página de manual de logger. ¿Con qué parámetro se indica la prioridad de los mensajes?**

$ man logger

**2) Utilizando la línea de comandos, enviar el mensaje “Hola Mundo de Logs” al fichero /var/log/syslog. Comprobar que se ha hecho correctamente.**

sudo logger "Hola Mundo de Logs"

tail -n 5 /var/log/syslog

**3) Enviar todos los mensajes de nivel “debug” generados por el servicio sshd al fichero /var/log/ssh.log. Este fichero debe haber sido creado anteriormente y estar vacío. Configurar el servicio sshd para que funcione en modo “debug” y comprobar el efecto que tiene en el fichero ssh.log.**

**4) Configurar la rotación de logs (fichero /var/log/syslog) para que se guarden de manera mensual y comprimida, y para que todos los logs generados en un año se guarden en un directorio llamado /var/log/syslog.old.**