



# Sistemas Operativos y Distribuidos



## Anexo práctico 1 Casos de estudio



### GNU/Linux

La realización del anexo GNU/Linux es obligatorio. Es importante la comprensión de estos contenidos dado que serán necesarios para la resolución de los próximos anexos.

**A.L.1.1.** Explicar la diferencia entre una distribución GNU/Linux y el kernel Linux.

**A.L.1.2.** ¿En qué lenguajes de programación está escrito el kernel Linux? ¿Considerás que es apropiado el uso de estos lenguajes de programación para escribir el kernel de un sistema operativo?

**A.L.1.3.** En sistemas operativos de tipo Unix como GNU/Linux existe un usuario denominado "root". Explicar la razón de su existencia. ¿Cuál es el objetivo de los comandos "su" y "sudo"?

**A.L.1.4.** Descargar la herramienta "VMware Workstation Player" del siguiente enlace: <https://my.vmware.com/web/vmware/downloads> e instalarla en tu computadora (Nota: también se puede usar la herramienta VirtualBox). Abrir y ejecutar la máquina virtual provista por la materia (en clases prácticas se informará cómo obtenerla). Iniciar sesión con el usuario provisto en la misma (leer el archivo info.txt dentro de la carpeta de la máquina virtual), ejecutar una terminal y seguir los siguientes pasos:

- Utilizar el usuario "root" para crear un usuario personalizado. Por ejemplo, si tu nombre es Estanislao Eraclio Perez tu nombre de usuario puede crearse como "eperez".
- Asignar un *password* al mismo.
- Cerrar la sesión actual e iniciar una nueva sesión con el usuario creado anteriormente.

**A.L.1.5.** El sistema GNU/Linux mantiene una base de datos que asigna los nombres de usuarios a los *userids*. Esta base de datos se almacena en el archivo de configuración `/etc/passwd`.

- Utilizar el comando "cat" para identificar el usuario "root" y el *uid* del mismo.
- Identificar el usuario creado en el inciso anterior y su directorio HOME. Relacionar esta información con la emitida por el comando "pwd".
- ¿En qué archivo se almacenan las contraseñas de los usuarios?
- ¿Cuál es la utilidad de los comandos: "whoami", "logname", "id"? ¿De dónde obtienen la información?

**A.L.1.6.** El archivo `/etc/shells` contiene todas las *shells* disponibles en el sistema. Utilizar al menos dos formas alternativas para conocer el intérprete de comandos que está asociado a su usuario.

**A.L.1.7.** ¿Cuál es la utilidad del comando “`man`”? Investigar cómo es la estructura de las páginas de manual. ¿Cuál es la relación del comando anterior con los comandos “`apropos`” e “`info`”?

**A.L.1.8.** Determinar la versión del kernel que se está utilizando y el tipo de arquitectura subyacente. ¿Posee soporte de virtualización a nivel de hardware?

**A.L.1.9.** ¿Que es un script “`BASH`”?

a) Crear un archivo denominado `sys.sh` y copiar en el mismo la siguiente información:

```
#!/bin/bash
clear
echo "Información provista por sys.sh."
echo "Hola, $USER"
echo
echo "Hoy es el día `date`, esta es la semana `date +%V` ."
echo
echo "Los siguientes usuarios están conectados:"
w | cut -d " " -f 1 - | grep -v USER | sort -u
echo
echo "Este es `uname -s` ejecutando en un procesador `uname -m` ."
echo
echo "El uptime es:"
uptime
echo
echo "Que tenga un buen día!"
exit 0
```

b) Para que el archivo tenga permisos de ejecución se debe ejecutar el siguiente comando: “`chmod u+x sys.sh`” (el cual veremos en detalle en próximos anexos).

c) Ejecutar el script y explicar mediante comentarios en el mismo lo que hace cada una de las líneas.

**A.L.1.10.** Todo comando posee un estado de terminación (*exit status*). Un comando que se ejecutó satisfactoriamente arrojará un valor de 0 mientras que un comando que no se procesó de manera correcta finalizará con un número diferente de 0. La variable `?` lee el valor de terminación del último comando ejecutado.

a) ¿Qué es una variable en `BASH`?

b) Utilizar la variable mencionada anteriormente para obtener el valor de salida del script anterior.

c) Crear una variable en una terminal e imprimir el valor de la misma.

**A.L.1.11.** Ejecute un comando para listar los últimos comandos ejecutados en la sesión actual.

**A.L.1.12.** Para que se utilizan los comandos “`strace`” y “`ltrace`”. Implemente un programa en C que utilice `printf` para imprimir por pantalla “hello world”, aplique el comando “`strace`” e identifique la llamada al sistema “`write`”. Cual es la relación entre `printf` y `write`? Relacione esta pregunta con la pregunta de `syscall` del práctico.



## Microsoft Windows

El anexo Microsoft Windows no es obligatorio pero se recomienda la resolución del mismo para tener una mejor comprensión de dicho sistema operativo.

**A.W.1.1.** El caso “Estados Unidos contra Microsoft” fue un conjunto de acciones legales planteadas el 8 de mayo de 1998 por el Departamento de Justicia de Estados Unidos (DOJ) junto con 20 Estados norteamericanos contra Microsoft Corporation, en virtud de la ley para limitar monopolios. El tema central del caso consistía en si se le permitía a Microsoft juntar en un mismo paquete su producto de navegación (Internet Explorer) con su sistema operativo Microsoft Windows.

- a) Discutir si un sistema operativo debería o no incluir aplicaciones como un navegador WEB o programas de correo. Dar argumentos a favor y en contra.
- b) Dada la baja cuota en el mercado de internet explorer, ¿considerás necesario una restricción para no incorporarlo?

**A.W.1.2.** ¿Qué es la Windows API? ¿Por qué han existido tantas versiones desde su comienzo?

**A.W.1.3.** ¿Es posible ejecutar Microsoft Windows sin una cuenta de usuario?

**A.W.1.4.** Identificar si tiene un sistema de 32-bit o de 64-bit. ¿Es posible instalar un sistema de 32-bit en una computadora que posea un procesador físico de 64-bit?

**A.W.1.5.** Ejecutar la herramienta “*computer management*” con el siguiente comando: “`compmgmt.msc`” e identificar la siguiente información:

- a) Nombre del usuario actual con el cual inicio sesión.
- b) Nombre del usuario administrador de sistema. ¿Es este usuario análogo al usuario root de GNU/Linux?
- c) ¿Cuál es la utilidad de la herramienta “*Event Viewer*”?
- d) ¿Cuál es la utilidad de la herramienta “*Disk Management*”?

**A.W.1.6.** En el inciso 1.15 del práctico 1 se resolvió una pregunta sobre los distintos modos de operación. A continuación haremos un experimento para monitorear cuánto tiempo consume nuestro sistema Windows en “*kernel mode*” vs. “*user mode*”:

- a) Ejecutar “*Performance Monitor*” con el siguiente comando: “`perfmon.exe`”
- b) Una vez ejecutado, seleccionar la entrada “*Performance Monitor*”.
- c) Hacer click en el botón “+” de la barra de herramientas.
- d) Expandir la sección “*Processor*”, hacer click en “% *Privileged Time*” y mientras se mantiene apretada la tecla Ctrl, hacer click en “% *User Time*”.
- e) Hacer click en “Add” y luego en “OK”.
- f) Ejecutar en una consola “CMD” el siguiente comando (sin las comillas): “`dir \\%computername%\c$ /s`”, el cual escanea su directorio disco C.
- g) En el gráfico resultante podrás ver el uso en “*kernel mode*” vs. “*user mode*”.

- h) La misma información puede ser vista en el *Task Manager*: ejecutar "Taskmgr", luego en "Performance CPU" hacer click derecho sobre el gráfico y seleccionar "Show kernel times".

**A.W.1.7.** Investigar que es "PowerShell" y para que se utiliza.

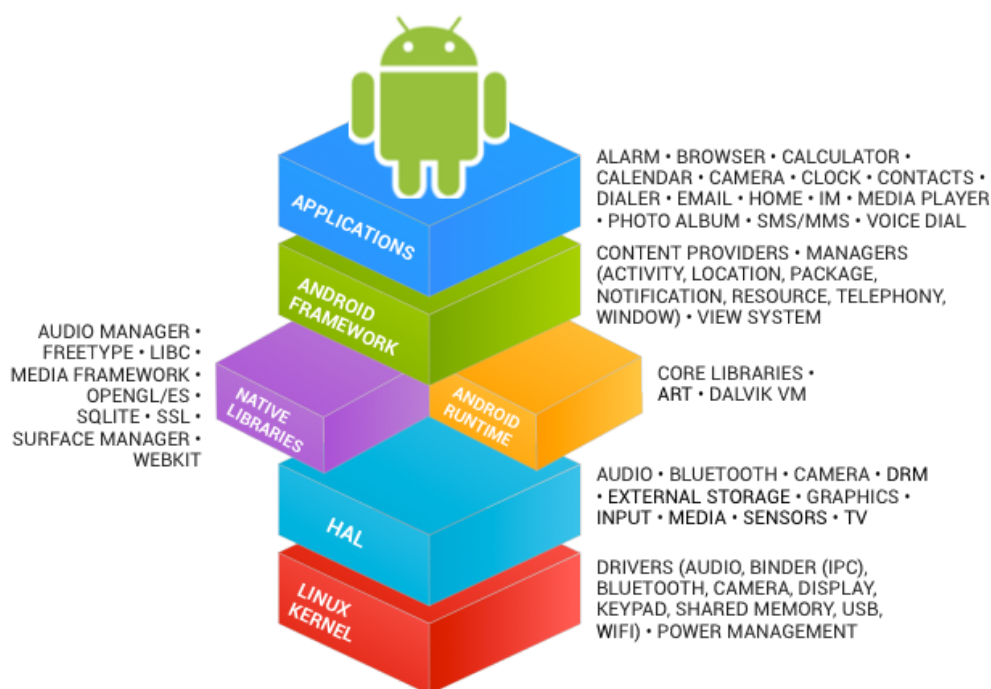
- a) En la misma escribir "Get-S" y presionar la tecla Tab varias veces. ¿Qué sucede?
- b) En la misma escriba "c:\\" y presionar la tecla Tab varias veces. ¿Qué sucede?
- c) Leer la descripción del comando "Get-Help" e investigar cómo se pueden actualizar los archivos de ayuda.
- d) Utilizar un comando para obtener la versión de "PowerShell".
- e) Utilizar un comando para obtener el nombre de su "Host".
- f) Utilizar un comando para obtener la fecha y hora del sistema.
- g) Ejecutar el siguiente comando: `"Get-CimInstance -ClassName win32_operatingsystem | select csname, lastbootuptime"`. ¿Qué información se obtiene? (Nota: la versión de PowerShell debe ser mayor a 3.0).
- h) Contrastar el comando anterior con el comando que se utiliza en GNU/Linux para obtener la misma información.



## Android

El anexo Android no es obligatorio pero se recomienda la resolución del mismo para tener una mejor comprensión de dicho sistema operativo.

**A.A.1.1.** ¿Por qué Android es considerado un “*software stack*” y no un sistema operativo? Ver la figura que se muestra a continuación:



**A.A.1.2.** ¿En qué son similares iOS y Android ? ¿En qué se diferencian?

**A.A.1.3.** Explicar por qué los programas de Java que se ejecutan en sistemas Android no utilizan la API estándar de Java y la máquina virtual.

**A.A.1.4.** ¿Qué es el “*Android runtime*” (ART)? ¿Cuál es su diferencia con “*Dalvik*”?

**A.A.1.5.** A partir de algún dispositivo que funcione con Android responder las siguientes preguntas:

- ¿Qué versión de Android tiene instalada?
- ¿Sobre qué versión de Kernel está ejecutando?
- ¿Cuenta con “*Security Path Level*”? ¿Cuál es su fecha?
- ¿Tiene acceso al “*uptime*” del mismo?
- ¿Está cifrado su teléfono? ¿Qué significa esto?
- Instalar la aplicación “*CPU-Z*” y obtener la siguiente información:
  - Cantidad de “*Cores*”.
  - “*Kernel Architecture*”.
  - Cantidad de Sensores.
  - “*Health*” de la batería.