Usuario **root**
Comandos **su** y **sudo**

Laboratorio de Sistemas Operativos II

1.

root

Es el nombre del superusuario o administrador del sistema operativo

*“La cuenta de usuario **root** puede realizar cualquier acción en el sistema.*

*Posee los permisos y privilegios necesarios para la configuración del sistema y gestión de **todos** los recursos, usuarios, procesos y archivos del sistema en todos los modos de ejecución, monousuario y multiusuario.*

Sobre **root**

- **root:x:0:0:root:/root:/bin/bash**
- **UID** = **0**
- El directorio de inicio (home) del superusuario es **/root**
- Contar con una cuenta de usuario normal para disminuir la probabilidad de cometer errores.
- Acceder como root sólo en tareas de mantenimiento y administración para evitar dañar archivos importantes del sistema.
- Existen varias formas de ingresar al sistema como root.

Métodos para acceder al intérprete de órdenes de root utilizando la contraseña

- En consola de texto, loguearse como root.
- En entorno de escritorio: «Aplicaciones» → «Accesorios» → «Terminal de Root».
- En consola utilizando el comando **`su`**

Fuente: https://www.debian.org/doc/manuals/debian-reference/debian-reference.es.pdf

2.

Comando `su`

Permite usar el intérprete de comandos de otro usuario con sus privilegios, sin necesidad de cerrar la sesión actual. Por defecto el usuario root

Comando **`su`**

- Permite ejecutar una shell como otro usuario en la sesión activa.
- Es decir, permite asumir la identidad de otro usuario (siempre que conozcamos su password).
- Situaciones:
  - Se encuentra como un usuario común y necesita realizar una tarea como root
  - Está logueado como root y necesita ser otro usuario.

Sintaxis del comando **su**

- `$su [opciones] [usuario]`
- `su -l [usuario]`
  - Fuerza el inicio de un nuevo shell, con las preferencias por defecto del usuario elegido. No conserva el entorno del usuario actual;
- `su [usuario]`
  - Conserva parte del entorno del usuario actual.
- Volver al shell anterior: `exit`

Uso del comando **su**

- Para ser root se debe conocer la contraseña de root y se asume su identidad.

```
alumno@mipc~:$su
Contraseña:
root@mipc~:#
```

- Para ser otro usuario se debe conocer su contraseña, PERO si eres root, el prompt no pide contraseña.

```
alumno@mipc~:$su glopez
Contraseña:
glopez@mipc~:$
```

- Para volver a cuenta de usuario anterior: exit

Usar o compartir la cuenta root **no es aconsejable**

- Privilegio: Derecho de un usuario a realizar una o más tareas específicas
- Un usuario no necesita obtener todos los privilegios de la cuenta root, sino sólo los necesarios para realizar sus acciones.
- El principio del menor privilegio es una estrategia de seguridad, aplicable a distintos ámbitos. Consiste en la idea de otorgar únicamente los privilegios necesarios para el desempeño de cierta actividad.
- La forma segura de otorgar privilegios es utilizar el comando **`sudo`**

3.

sudo

Comando para para proporcionar algunos privilegios de superusuario a los usuarios de acuerdo a la configuración establecida

Comando **sudo**

- Es un programa diseñado para permitir a los administradores de sistemas proporcionar privilegios limitados de root a ciertos usuarios y registrar las actividades de root.
- La filosofía básica es dar tan pocos privilegios como sea posible pero permitiendo que la gente pueda desarrollar su trabajo.
- Para ver los comandos que puede ejecutar como usuario de sudo:
  - `alumno@mipc:~$sudo -l`

Edición de /etc/sudoers

- El comando **sudo** se configura a través del archivo **/etc/sudoers**.
- Este archivo se edita sólo con el comando visudo.
- El comando visudo abre un editor de texto igual al normal, pero valida la sintaxis del archivo al guardarlo. Esto evita que los errores de configuración bloqueen las operaciones sudo, que pueden ser su única forma de obtener privilegios root.
- El comando visudo abre el archivo /etc/sudoers con el editor de texto vi. Sin embargo, puede estar configurado para utilizar el editor de texto nano en su lugar.
- Para cambiar el editor predeterminado: **`update-alternatives --config editor`**

Configuración de **sudo**

- En el archivo /etc/sudoers se edita Lista de control de acceso.
- Establece **quién** (**usuarios**) puede ejecutar **qué** (**comandos**) y de **qué modo** (**opciones**)
- Su posible configuración puede dividirse en 3 partes:
  - **Alias**
  - **Opciones (Defaults)**
  - **Reglas de acceso**
- Ninguna de las secciones es obligatoria, o tienen orden específico, pero la que al menos debe de existir es la tercera, que es la definición de los controles o reglas de acceso.

Alias – Concepto y características

- Se refiere a un usuario, un comando o un equipo. Engloba bajo un solo nombre (nombre del alias) una serie de elementos que en la parte de definición de reglas serán referidos.
- Los alias tienen **4 tipos** de alias, dependiendo del tipo de elementos que agrupen:
  - **Cmnd_Alias** - alias de comandos.
  - **User_Alias** - alias de usuarios normales.
  - **Runas_Alias** - alias de usuarios administradores o con privilegios.
  - **Host_Alias** - define alias de hosts o equipos.
- El NOMBRE_ALIAS puede llevar letras, números o guión bajo (_) y DEBE de comenzar con una letra mayúscula, se acostumbra a usarlos siempre en mayúsculas.

Sintaxis para la Creación de **Alias**

- En el archivo /etc/sudoers se agrega la línea para crear ALIAS:

```
tipo_alias NOMBRE_ALIAS = elemento1, elemento2, …, elementoN
```

```
tipo_alias NOMBRE1 = elemento1, elemento2 : NOMBRE2 = elemento3,
elemento4
```

Ejemplos de creación de **Cmnd_Alias**

Definen uno o más comandos y otros alias de comandos que podrán ser utilizados después en alias de usuarios

```
Cmnd_Alias WEB = /usr/sbin/apachectl, /usr/sbin/httpd, sudoedit /etc/httpd/
```

Indica que a quien se le aplique el alias WEB podrá ejecutar los comandos apachectl, httpd y editar todo lo que este debajo del directorio /etc/httpd/, nótese que debe de terminar con '/' cuando se indican directorios. También, la ruta completa a los comandos debe ser indicada.

```
Cmnd_Alias APAGAR = /usr/bin/shutdown -h 23\:00
```

Al usuario que se le asigne el alias APAGAR podrá hacer uso del comando 'shutdown' exactamente con los parámetros como están indicados, es decir apagar -h (halt) el equipo a las 23:00 horas. Nótese que es necesario escapar el signo ':', asi como los símbolos ' : , = \

```
Cmnd_Alias NET_ADMIN = /sbin/ifconfig, /sbin/iptables, WEB
```

NET_ADMIN es un alias con los comandos de configuración de interfaces de red ifconfig y de firewall iptables, pero además le agregamos un alias previamente definido que es WEB, asi que a quien se le asigne este alias podrá hacer uso de los comandos del alias WEB.

Ejemplos de creación de **User_Alias**

Definen a uno o más usuarios, grupos del sistema (indicados con %), grupos de red (netgroups indicados con +) u otros alias de usuarios.

```
User_Alias MYSQL_USERS = andy, marce, juan, %mysql
```

Indica que al alias MYSQL_USERS pertenecen los usuarios indicados individualmente más los usuarios que formen parte del grupo 'mysql'.

```
User_Alias ADMIN = sergio, ana
```

'sergio' y 'ana' pertenecen al alias ADMIN.

```
User_Alias TODOS = ALL, !samuel, !david
```

Alias de usuario TODOS que al poner como elemento la palabra reservada 'ALL' abarcaría a todos los usuarios del sistema, pero no deseamos a dos de ellos, asi que negamos con '!', que serían los usuarios 'samuel' y 'david'. Es decir, todos los usuarios menos esos dos. NOTA IMPORTANTE: los usuarios nuevos que se añadan después al sistema también serán considerados como ALL, es mejor siempre definir específicamente a los usuarios que se requieran. ALL es válido en todos los tipos de alias.

```
User_Alias OPERADORES = ADMIN, alejandra
```

Los del alias ADMIN más el usuario 'alejandra'.

Ejemplos de creación de **Runas_Alias**

Funciona exactamente igual que User_Alias, la única diferencia es que es posible usar el ID del usario UID con el caracter '#'..

```
Runas_Alias OPERADORES = #501, fabian
```

Al alias OPERADORES pertenecen el usuario con UID 501 y el usuario 'fabian'

Ejemplos de creación de **Host_Alias**

Definen uno o más equipos u otros alias de host. Los equipos pueden indicarse por su nombre (si se encuentra en /etc/hosts) por nombre de dominio, si existe un resolvedor de dominios, por dirección IP, por dirección IP con máscara de red.

```
Host_Alias LANS = 192.168.0.0/24, 192.168.0.1/255.255.255.0
```

El alias LANS define a todos los equipos de las redes locales.

```
Host_Alias WEBSERVERS = 172.16.0.21, web1 : DBSERVERS = 192.168.100.10,
dataserver
```

Se define dos alias en el mismo renglón: WEBSERVERS y DBSERVERS con sus respectivas listas de elementos, el separador ':' es válido en cualquier definición de tipo de alias.

Reglas de Acceso

- Aunque no es obligatorio declarar alias, ni opciones (defaults), y de hecho tampoco reglas de acceso, pues el archivo /etc/sudoers no tendría ninguna razón de ser si no se crean reglas de acceso. De hecho podríamos concentrarnos en crear solamente reglas de acceso, sin opciones ni alias y podría funcionar todo muy bien.

- Las reglas de acceso definen qué usuarios ejecutan cuáles comandos bajo cuál usuario y en qué equipos.

Crear **Reglas de Acceso**

```
usuario host = comando1, comando2, ... comandoN
```

- Sintaxis básica, 'usuario' puede ser un usuario, un alias de usuario o un grupo (indicado por %), 'host' puede ser ALL cualquier equipo, un solo equipo, un alias de equipo, una dirección IP o una definición de red IP/máscara, 'comando1, comando2,...' es cualquier comando indicado con su ruta completa. Si se termina en '/' como en /etc/http/ entonces indica todos los archivos dentro de ese directorio.

Ejemplos de **Reglas de Acceso**

```
daniela ALL = /sbin/iptables
```

- Usuario 'daniela' en cualquier host o equipo puede utilizar iptables.

```
ADMIN ALL = ALL
```

- Los usuarios definidos en el alias 'ADMIN' desde cualquier host pueden ejecutar cualquier comando.

Más ejemplos de **Reglas de Acceso**

```
%gerentes dbserver = (director) /usr/facturacion, (root)
/var/log/*
```

- Un ejemplo más detallado. Los usuarios que pertenezcan al grupo del sistema llamado 'gerentes' pueden en el equipo llamado 'dbserver' ejecutar como si fueran el usuario 'director' la aplicación llamada 'facturacion', además como usuarios 'root' pueden ver el contenido de los archivos que contenga el directorio /var/log.
- Lo anterior introduce algo nuevo, que en la lista de comandos es posible indicar bajo que usuario se debe ejecutar el permiso. Por defecto es el usuario 'root', pero no siempre tener que asi. Además la lista 'hereda' la primera definición de usuario que se indica entre paréntesis ( ), por eso si se tiene más de alguno hay que cambiar de usuario en el comando conveniente, el ejemplo anterior también sería válido de la siguiente manera:

```
%gerentes dbserver = /var/log/*, (director) /usr/facturacion
```

Y más ejemplos de **Reglas de Acceso**

No es necesario indicar (root) ya que es el usuario bajo el cual se ejecutan los comandos por defecto. También es válido usar (ALL) para indicar bajo cualquier usuario. El ejemplo siguiente da permisos absolutos.

```
sergio ALL = (ALL) ALL
```

Se establece permiso para el usuario 'sergio' en cualquier host, ejecutar cualquier comando de cualquier usuario, por supuesto incluyendo los de root.

```
SUPERVISORES PRODUCCION = OPERACION
```

Una regla formada solo por alias. En el alias de usuario 'SUPERVISORES' los usuarios que esten indicados en ese alias, tendrán permiso en los equipos definidos en el alias de host 'PRODUCCION', de ejecutar los comandos definidos o listados en el alias de comandos 'OPERACION'.

Fuente:https://www.linuxtotal.com.mx/index.php?cont=info_admon_014

Para recordar:

Podemos recordar lo siguiente para escribir las reglas::

QUIÉN DÓNDE=(COMO QUIÉN) QUÉ

El **QUIÉN** indica a qué usuarios o alias se refiere la directiva.

En **DÓNDE** siempre veremos la palabra clave ALL que -para ser breves- podemos decir que representa el equipo en el que estemos trabajando.

COMO QUIÉN indica la cuenta de usuario cuyos privilegios serán «compartidos» para realizar **QUÉ** (el comando o conjunto de comandos).

https://blog.carreralinux.com.ar/2016/09/ejemplos-en-el-archivo-sudoers/

Y aún más ejemplos

`gcanepa ALL=(root) USUARIOS`

Indica que el usuario gcanepa podrá ejecutar los comandos listados en el alias USUARIOS con permisos de root en el equipo actual.

`jperez ALL=(root) PAQUETES`

Indica que jperez podrá ejecutar los comandos listados en PAQUETES con permisos de superusuario en el equipo actual.

`ADMINISTRADORES ALL=NOPASSWD:/sbin/shutdown`

Indica que los usuarios que integran el alias ADMINISTRADORES podrán utilizar /sbin/shutdown en el equipo actual sin necesidad de ingresar su contraseña (NOPASSWD).

`%www-data ALL=(root) /usr/sbin/apache2ctl`

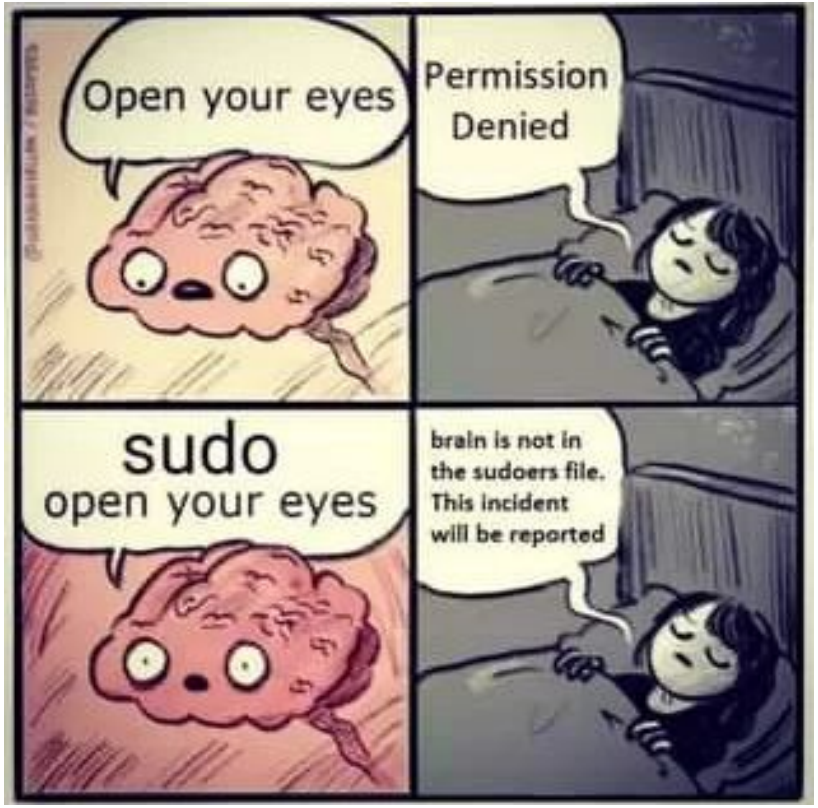
Permitirá que cualquier usuario que pertenezca al grupo www-data pueda detener, iniciar, reiniciar, etc. el servidor web Apache.

Entonces, la configuración de root es:

```
root ALL=(ALL:ALL) ALL
```

Donde cada ALL significa:

1) **ALL**=(ALL:ALL) ALL: en este se indica que la regla se aplica a cualquier anfitrión (o host).
2) ALL=(**ALL**:ALL) ALL: "user" podrá usar comandos de cualquier usuario.
3) ALL=(ALL:**ALL**) ALL: si el anterior "ALL" permitía usar comandos de usuarios, éste lo hará de grupos.
4) ALL=(ALL:ALL) **ALL**: las reglas se aplican a todos los comandos.


Open your eyes
Permission Denied
sudo open your eyes
brain is not in the sudoers file. This incident will be reported