TEMA 4

Seguridad

ADMINISTRACIÓN DE SISTEMAS OPERATIVOS

Curso 2020/2021

Cristian Martín Fernández

Contenido

- PRINCIPIOS GENERALES DE SEGURIDAD EN LA ADMINISTRACIÓN DE SISTEMAS OPERATIVOS
- SUDO
- GESTIÓN DE ACCESSO A PROGRAMAS
- SEGURIDAD SSH
- CORTAFUEGOS
- DETECCIÓN DE INSTRUSIÓN
- RESPUESTA A INCIDENTES
- DESPLIEGE DEL SERVIDOR DESPUÉS DE UN INCIDENTE
- ANÁLISIS FORENSE

Principios generales de la seguridad en SO

- No es posible garantizar la seguridad al 100%.
- Simplificar. La complejidad es un factor que perjudica la seguridad. Aumenta la vulnerabilidad. Falsa percepción de seguridad.
- Principio del privilegio mínimo. Tanto usuarios como software deben disponer siempre del mínimo nivel de seguridad.
- Diseño abierto. Evitar diseño basado en la oscuridad. Ralentización. Falsa sensación de seguridad.
- Actualizaciones de seguridad del software y componentes.
- Registro de eventos. Detección precoz.

Sudo

- Deshabilitar root y usar sudo
- Permite habilitar permisos de administración a cualquier usuario
- Inhabilita ataques a la cuenta root
- Limitar los permisos de administración mediante perfiles sudo. Principio del menor privilegio.
- Soporte para accesos basados en grupos y hosts. Archivo único de configuración para toda la red.
- Usabilidad: Caducidad automática de los accesos.
- Registro de todos los accesos sudo.
- Configuración de accesos sin contraseña a las reglas. Muy útil para ejecución de shell scripts cron.

/etc/sudoers

- El archivo /etc/sudoers controla quién puede ejecutar qué comandos y qué usuarios en qué máquinas y también puede controlar cosas especiales, como si necesita una contraseña para comandos particulares. El archivo se compone de alias y especificaciones de usuario.
- Edición del archivo /etc/sudoers:
  - Herramienta “visudo”. Comprueba errores antes de guardar cambios en el fichero.
  - No se recomienda editar el fichero manualmente. Un error en este fichero podría dejar inaccesible todo el acceso de administración si no tenemos habilitado el usuario “root”
  - El editor por defecto es nano. Para cambiar: $ sudo select-editor
- Podemos considerar tres secciones en /etc/sudoers:
  - Definiciones de alias
  - Ajuste de las opciones por defecto
  - Reglas de acceso
- Las tres secciones son opcionales. Sin la sección “Reglas de acceso” no tendría efecto el comando sudo, pero sería válido.

Ejemplo de regla en sudoers

Alumno webserver = (root) /usr/sbin/apache2ctl

Columnas:

1. “Alumno”: Usuario o grupo al que se aplica la regla (% para grupo)
2. “Webserver”: Hosts a los que se aplica la regla
3. “(root)”: Bajo qué usuario se ejecutarán los comandos sudo.
4. “/usr/sbin/apache2ctl”. Define qué comandos podrá ejecutar el usuario definido en la regla

Definiciones de Alias

- Con el objetivo de tener mejor organizado el fichero, aumentar la legibilidad, y evitar las repeticiones podemos hacer uso de los Alias.
- Desde el momento en que hemos definido un alias para un elemento, puede utilizarse en cualquier lugar, incluida la definición de un alias , en el que se espere un usuario, un comando o un host.
- La forma general de una definición de alias es:

Tipo_Alias NOMBRE_ALIAS = elemento1, elemento2, elemento3, ...

Tipo_Alias Puede ser uno de los siguientes:

- Cmnd_Alias: Para definir alias de comandos.
- User_Alias: Para definir alias de usuarios "normales".
- Runas_Alias: Para definir alias de usuarios "privilegiados".
- Host_Alias: Para definir alias de hosts.

Definiciones de Alias

- NOMBRE_ALIAS Es el nombre que daremos al alias. Para su composición deben seguirse unas simples normas:
  - Debe ser una cadena compuesta exclusivamente por letras, números y guiones bajos “_”.
  - La cadena debe comenzar necesariamente con una letra mayúscula.
  - Se recomienda que en los nombres de alias se utilicen solo letras mayúsculas para distinguirlos de un nombre de usuario.
- elemento
  - Son los elementos o listas de elementos para los cuales será expandido el NOMBRE_ALIAS.
  - Cuando se trate de una lista, deben separarse los elementos constitutivos de la misma mediante comas (,)
  - Un alias puede ser un elemento en otra definición de alias.

Definiciones de Alias

- Es posible definir varios alias de un mismo tipo en una sola línea. Las diversas definiciones deben separarse con dos puntos (:)

  Tipo_Alias NOMBRE_1 = elemento1, elemento3 : NOMBRE_2 = elemento2,\ elemento4 : NOMBRE_3 = elemento5, elemento6

- Esto equivaldría a :

  Tipo_Alias NOMBRE_1 = elemento1, elemento3

  Tipo_Alias NOMBRE_2 = elemento2, elemento4

  Tipo_Alias NOMBRE_3 = elemento5, elemento6

- Un alias puede ser un elemento en la definición de otro alias (del mismo tipo).

  Tipo_Alias NOMBRE_5 = elemento7, NOMBRE_3 : NOMBRE_6 = elemento8,\ elemento9, NOMBRE_1

- La única restricción es que los alias utilizados como elementos en las definiciones de otros alias (en nuestro caso, NOMBRE_3 y NOMBRE_1) deben haber sido definidos previamente, y deben ser del mismo tipo del alias que se define.

Alias de usuarios

- Los alias de usuario se utilizan para especificar grupos de usuarios.
- Alias de Usuarios: User_Alias

  User_Alias NOMBRE_ALIAS = Lista_Usuarios

  Lista_Usuarios: elemento1, elemento2, ...

  elemento :

  nombre_usuario

  %nombre_grupo (para system groups)

  +nombre_grupo (para net groups)

  !nombre_usuario (para quitar este usuario del grupo definido)

Alias de usuarios

- La lista de usuarios puede estar formada por uno o más nombres de usuarios, nombres de grupos (precedidos por % en el caso de "system groups" o por + en el caso de "net groups") y otros alias.
- Ejemplo:

  User_Alias INVITADOS = asd53, delfin, %impresora, +robinsones, REDACTORES, !paco

- Establece que el alias INVITADOS comprende a los usuarios asd53 y delfin, los usuarios del grupo (system-group) impresora, a los usuarios del grupo (net-group) robinsones y a los usuarios definidos en el alias REDACTORES exceptuando de todo paco.

Alias de usuarios privilegiados

- Las Runas_Alias son similares a las User_Alias, salvo en que las Runas_Alias pueden contener uid, con el prefijo #.
- Alias de Usuarios Privilegiados: Runas_Alias

  Runas_Alias NOMBRE_ALIAS = Lista_Runas

  Lista_Runas : elemento1, elemento2, ...

  elemento :

  nombre_runas

  %nombre_grupo (para system groups)

  +nombre_grupo (para net groups)

  #uid

Alias de usuarios privilegiados

- Ejemplo:
  - Runas_Alias ROOT = #0
- El UID 0 es normalmente utilizado para root
- Para todos los usuarios del grupo admin y el usuario root
  - Runas_Alias ADMINS = %admin, root

Alias de host

- Un alias de host es una lista de nombres de host, direcciones IP, redes y grupos de red (con el prefijo +).
- Si no especifica una máscara de red con una red, la máscara de red de las interfaces ethernet de los hosts se utilizará al hacer coincidir.
- Alias de Hosts : Host_Alias

  Host_Alias NOMBRE_ALIAS = Lista_hosts

  Lista_hosts : elemento1, elemento2, ...

  elemento :

  nombre_host

  dirección IP

  números de red (/máscara)

  +nombre_grupo (para net groups)

- El nombre_host puede incluir caracteres comodín (wildcards)

Alias de host

```
# Esto es para todos los servidores

Host_Alias SERVERS = 192.168.0.1, 192.168.0.2, server1

# Esto es para toda la red

NETWORK = 192.168.0.0/255.255.255.0

# Esto es para cada maquina en la red que no es un servidor

Host_Alias WORKSTATIONS = NETWORK, !SERVER

# Lo anterior también se puede hacer en una única línea

Host_Alias WORKSTATIONS = 192.168.0.0/255.255.255.0, !SERVERS
```

Alias de comandos

- Una Cmnd_Alias es una lista de uno o más nombres de comandos, directorios y/o otros alias. Si especifica un directorio, incluirá cualquier archivo dentro de ese directorio pero no en ningún subdirectorio.
- Cuando se especifique un comando, siempre debe ponerse la ruta completa al ejecutable.
- Alias de Comandos : Cmnd_Alias

  Cmnd_Alias NOMBRE_ALIAS = Lista_comandos

  Lista_comandos : elemento1, elemento2, ...

  elemento :

  nombre_comando

  nombre_comando argumentos

  nombre_comando wildcards

  Directorio

  sudoedit

Alias de comandos

- Ejemplos:

  Cmnd_Alias BASICOS = /bin/ls, /usr/bin/lpr

  Cmnd_Alias AVANZADOS = /bin/kill, /bin/passwd, BASICOS

- Cuando el nombre del comando se especifica con argumentos el usuario sólo podrá utilizar los argumentos especificados.
- Si el nombre del comando se especifica con el argumento guion bajo " _" , se indica que el comando sólo puede ejecutarse sin argumentos.
- Un directorio es un path, completo y válido, acabado en /.
- Ejemplo:

  /usr/local/bin/

Alias de comandos

- Si en lugar de un comando se especifica un directorio, el usuario puede ejecutar cualquier binario que se encuentre en el directorio especificado.
- En el siguiente ejemplo, el usuario podría ejecutar los comandos kill, passwd, hostname, todos los ejecutables contenidos en el directorio /usr/local/bin/ y los comandos correspondientes al alias BASICOS :

  Cmnd_Alias AVANZADOS = /bin/kill, /bin/passwd, /bin/hostname, BASICOS, /usr/local/bin/

- Muy importante. Los siguientes caracteres deben ser "escapados" con \ en el caso de ser utilizados en los argumentos del comando: ' : , = \
- El comando especial sudoedit se utiliza para permitir al usuario la ejecución de sudo –e para editar ficheros (permite argumentos).

Ajuste de las opciones

- Pueden modificarse ciertas opciones de configuración mediante una o más líneas Defaults en el archivo /etc/sudoers.
- Podemos definir las opciones:
    1. globalmente
    2. por usuario
    3. por usuario privilegiado
    4. por host
- Y sus respectivas sintaxis serian las siguientes:
    1. Defaults lista_opciones
    2. Defaults:usuario lista_opciones
    3. Defaults>usuario_privilegiado lista_opciones
    4. Defaults@host lista_opciones

Listas de opciones

- Las listas de opciones están constituidas por un conjunto de opciones separadas por comas:

  opcion1, opcion2, opcion3,.....

- Podemos considerar cuatro tipos de opciones:
  - Booleanas
  - Enteros
  - Cadenas
  - Listas

Opciones booleanas

- Se activan escribiendo el nombre de la opción.
- Se desactivan colocando el operador ! delante el nombre de la opción

Defaults>root !set_logname

#Desactiva la opción set_logname para el usuario root

Defaults:sebastian authenticate

#Activa la opción authenticate para el usuario sebastian

ignore_dot

Valor por defecto : off

Si se activa, sudo ignorará el punto . (Directorio actual) en la variable de entorno PATH. La variable PATH en sí no es modificada.

Opciones booleanas

mail_always

Valor por defecto : off

Advierte, mediante un mensaje de correo, al mailto user , cada vez que un usuario utiliza sudo.

mail_badpass

Valor por defecto : on

Advierte, mediante un mensaje de correo, al mailto user, cada vez que un usuario, al arrancar sudo, no entra el password correcto.

mail_no_user

Valor por defecto : on

Activada, se enviará un mensaje al mailto user , advirtiéndole cada vez que sudo es invocado por un usuario que no está en el archivo sudoers.

mail_no_host

Valor por defecto : off

Activada, se enviará un mensaje de correo al mailto user, cada vez que sudo es invocado por un usuario existente en sudoers, pero no habilitado para ejecutar comandos en el host actual.

Opciones enteras

- nombre_opcion = valor

Algunas opciones enteras:

passwd_tries

Valor por defecto : 3

- Especifica el número de veces que un usuario puede intentar entrar su contraseña antes que sudo considere el fallo definitivo y se cierre.

timestamp_timeout

Valor por defecto : 5

- Con esta opción se establecen los minutos que deben pasar antes que sudo vuelva a requerir la introducción de la contraseña.
- Si se establece un valor igual a 0 sudo requerirá la introducción de la contraseña para cada operación.
- Si el valor que se da es menor que 0, no habrá tiempo de caducidad de la contraseña introducida al inicio de la sesión.

Opciones enteras

- passwd_timeout

Valor por defecto : 5

Establece (en minutos) el tiempo de espera de introducción de contraseña. Si se establece en 0 se elimina el tiempo de espera de introducción de contraseña.

- umask

Valor por defecto : 022

Se utiliza para establecer la umask a usar durante la ejecución de un comando.

Negando esta opción (!) o estableciéndola en 0777 se conservará la umask del usuario

Opciones cadenas

nombre_opcion = “cadena”

- badpass_message

Valor por defecto : Sorry, try again

Establece el mensaje a mostrar cuando un usuario intente entrar con un password incorrecto.

- timestampdir

Valor por defecto : /var/run/sudo

Establece el directorio en el cual sudo almacenará sus archivos timestamp.

- timestampowner

Valor por defecto : root

Establece el propietario del directorio timestamp y de los archivos que contenga.

Opciones cadenas

- passprompt

Valor por defecto : Password

Establece el promtp a utilizar por defecto cuando se requiera la introducción del password. Puede ser sustituido via sudo -p o a través de la variable de entorno SUDO_PROMPT.

Soporta las siguientes opciones:

%u

Muestra el nombre_usuario del usuario que invoca sudo.

%U

Muestra el nombre_usuario del usuario en cuyo nombre se ejecutará el comando (por defecto: root).

%h

Muestra el hostname local, sin el nombre de dominio.

%H

Muestra el hostname local, incluyendo el nombre de dominio

%%

Muestra el signo %.

Opciones listas

- nombre_opcion = valor1,valor2, ...
- En las listas el operador = puede ser sustituido por += o -=, para añadir o quitar elementos.
- Los valores o cadenas, cuando contengan varias palabras, pueden encerrarse entre dobles comillas ("...").
- Los caracteres especiales pueden "escaparse" con \

- Algunas opciones de listas:

env_check

Las variables de entorno a eliminar del entorno del usuario si el valor de la variable contiene los caracteres % o / .

El argumento puede ser una lista de valores separados por espacios o encerrados entre dobles-comillas (" ").

La lista puede ser reemplazada, incrementada o disminuida usando los operadores = , += -= , o ! respectivamente.

La lista, por defecto, de variables de entorno a comprobar es mostrada cuando se ejecuta sudo -V como root.

Reglas de acceso

- "*If this was a film this part is where all the key threads of the story come together in the glorious unveiling before the final climatic ending.*"
- Las reglas de acceso nos permiten definir:
  - Los usuarios a los que permitimos utilizar sudo.
  - Los comandos que dichos usuarios podrán ejecutar.
  - En calidad de qué usuarios privilegiados podrán ejecutarlos.
  - En que hosts podrán hacerlo.
- La sintaxis básica es:
  - usuario host = (usuario_privilegiado) comando
- Cada uno de los elementos anteriores puede ser un alias o una lista de elementos.
- ALL aplica para todo
- Si no se proporciona usuario_privilegiado, el comando se ejecutará como root o el usuario que aparece en el parámetro -u de la invocación de sudo.

Etiquetas de comando (tags)

- Nos permiten modificar las condiciones en que el usuario puede ejecutar los comandos definidos.
- Un comando puede definirse con cero o más etiquetas asociadas a el.
- En el caso de utilizar ETIQUETAS DE COMANDO, la sintaxis de la regla de acceso pasaría a ser:

  usuario host = (usuario_privilegiado) ETIQUETA:comando, ...

- Disponemos de cuatro valores posibles para ETIQUETA:
  - NOPASSWD
  - PASSWD
  - EXEC
  - NOEXEC

Etiquetas de comando (tags)

- Una vez se ha asociado una etiqueta a un comando, los siguientes comandos de la lista de comandos heredan el valor de la etiqueta asociada, salvo que sea "anulada" por una etiqueta con valor opuesto.
- Las parejas de valores "opuestos" son:
  - NOPASSWD -- PASSWD
  - EXEC – NOEXEC

Etiquetas NOPASSWD - PASSWD

- Por defecto, sudo requiere que el usuario se autentifique mediante su contraseña antes de ejecutar un comando. Esto puede modificarse mediante la etiqueta NOPASSWD.
- Al aplicar la etiqueta NOPASSWD a un comando de una lista todos los comandos subsiguientes "heredarán" esta propiedad, sin embargo podemos desactivar la "herencia" con PASSWD .

Etiquetas de comando (tags)

jose1 hosto = (operador) NOPASSWD: /bin/kill, /bin/ls, /usr/bin/lpr

- La anterior definición nos dice que el usuario jose1, en el host hosto, en calidad de usuario_privilegiado operador, puede ejecutar sin necesidad de autentificarse los comandos /bin/kill, /bin/ls, y /usr/bin/lpr, .
- Si deseáramos que solo pudiera ejecutar /bin/kill sin necesidad de autentificarse, desactivaríamos NOPASSWD para /bin/ls y /usr/bin/lpr, mediante PASSWD:

  jose1 hosto = (operador) NOPASSWD: /bin/kill, PASSWD: /bin/ls, /usr/bin/lpr

- Si quisiéramos que /bin/ls fuera el único comando que pudiera ejecutarse sin autentificación, la sintaxis debería ser:

  jose1 hosto = (operador) /bin/kill, NOPASSWD: /bin/ls, PASSWD: /usr/bin/lpr

Ejercicio de clase

- Dado el siguiente alias de comando para controlar los comandos de reinicio, apagado y termino de sesión:
  - Cmnd_Alias SHUTDOWN_CMDS = /sbin/poweroff, /sbin/halt, /sbin/reboot
- Crea una nueva regla para sudo para permitir a los usuarios del grupo admin y en todos los hosts y usuarios privilegiados poder apagar/reiniciar/suspender el ordenador sin tener que introducir la contraseña.

Etiquetas NOEXEC - EXEC

- NOEXEC se utiliza para evitar que cualquier programa inicie shells por sí mismo (ya que una vez que un programa se ejecuta con sudo, tiene privilegios de root completos, por lo que podría iniciar un shell de root para eludir cualquier restricción en el archivo sudoers).
- Ejemplo:

myuser ALL = (root) NOPASSWD:NOEXEC: /usr/bin/vim

- Este ejemplo permite al usuario "myuser" ejecutar como root el binario "vim" sin una contraseña y sin dejar que vim ejecute una shell.

Caracteres especiales y comodines

- Sudo permite la utilización de caracteres comodín de shell (shell-style wildcards) en los pathnames y en los argumentos de comandos, utilizados en el archivo /etc/sudoers.
- Los comodines utilizados son:
  - *
  - ?
  - [….]
  - [!…]
- Se utiliza también \ para escapar caracteres especiales.
- El carácter \ , como parte de un path, no puede ser sustituido por un comodín:
- /usr/bin/* hará referencia a todos los archivos existentes en /usr/bin/ , pero no "expandirá" el path a , por ejemplo, /usr/bin/X11/xterm .

Caracteres especiales y comodines

- El signo # es utilizado para indicar comentarios. Cada línea de comentarios debe comenzar por un signo #.
- El signo ! puede utilizarse como un operador lógico de negación (not), tanto en alias como en comandos.
- Las líneas largas pueden continuarse en otra línea inferior, colocando \ como último carácter de la línea que se desea interrumpir.
- Los espacios en blanco entre elementos de una línea, incluyendo los caracteres sintácticos especiales tales como = , : ( ) , son opcionales.
- Los siguientes caracteres especiales deben ser escapados con \ cuando se utilicen como parte de una palabra: @ ! = : ( ) \ ,

Ejemplo de configuración

```
# ************************

# DEFINCION DE ALIAS

# ************************


# administradores con todos los privilegios

User_Alias ADMINS = aso1, aso2


# administradores de red

User_Alias NETOPS = aso3, aso4


# webmasters -

User_Alias WEBMAS = aso5, aso6


# supervisores de producción (todos los del grupo de sistema supervisores y cristian)

User_Alias SUPPRO = cristian, %supervisores


# usuarios que pueden conectarse desde Internet

User_Alias INETUS = NETOPS, ADMINS, cristian
```

Ejemplo de configuración

```
# servidores web

Host_Alias WEBSERVERS = 10.0.1.100, 10.0.1.101


# servidores de aplicaciones

Host_Alias APLICACIONES = WEBSERVERS, 10.0.1.102, 10.0.1.103, mailserver


# comandos de red permitidos

Cmnd_Alias REDCMDS = /sbin/ifconfig, /sbin/iptables


# comandos de apache

Cmnd_Alias APACHECMDS = /usr/sbin/apachectl, /sbin/service httpd *
```

Ejemplo de configuración

```
# ************************

# DEFINICION DE OPCIONES

# ************************


# Los usuarios administradores, requieren autentificarse con la contraseña de 'root'

Defaults>ADMINS rootpw


# Para todos los usuarios, tienen hasta dos intentos para ingresar su contraseña y 3 minuto para que esta expire

Defaults passwd_tries = 4, passwd_timeout = 1


# Los usuarios que se conectan desde Internet, solo tienen una oportunidad y cero timeout lo que implica

# que cada comando que usen a través de sudo requerirá siempre de autentificación.

Defaults:INETUS passwd_tries = 1, passwd_timeout = 0


# Máscara de directorios y archivos por defecto, para los que ejecuten sudo en los servidores web

Defaults@WEBSERVERS umask = 022
```

Ejemplo de configuración

```
# **********************

# DEFINCION DE REGLAS

# **********************


# administradores todo se les permite en cualquier equipo

ADMINS ALL = (ALL) ALL


# administradores de red, en todos los equipos, los comandos de red

NETOPS ALL = REDCMDS


# webmasters, en los servidores web con los comandos indicados en apachecmds y además sin necesidad

# de contraseña acceder a los logs de apache y reiniciar los servidores.

WEBMAS WEBSERVERS = APACHECMDS, NOPASSWD: /var/log/apache/, /sbin/reboot
```

Ejemplo de configuración

```
# supervisores, pueden ejecutar los comandos indicados en los equipos referenciados en el alias
#  de aplicaciones y además son ejecutados bajo el usuario apps.
SUPPRO APLICACIONES = NOEXEC: (apps) /usr/local/facturacion, /usr/local/ventas, /usr/local/nomina

# rrhh es de recursos humanos y puede cambiar contraseñas de cualquier usuario menos de root
rrhh ALL = /usr/bin/passwd *, !/usr/bin/passwd root

# “aso7”, puede apagar los equipos de aplicaciones
aso7 APLICACIONES = /sbin/shutdown, /sbin/halt

#  El equipo firewall de la red puede ser reiniciado (no apagado) por “aso8” que es asistente de redes
aso8 firewall = /sbin/shutdown -r now
```

Ejercicio de clase

- Cree una configuración que cumpla con las siguientes características:
- Usuarios:
  - FULLTIMERS: cristian, aso1, aso2
  - PARTTIMERS: aso3, aso4
  - WEBMASTER: alumno, aso5
- Privilegiados:
  - OP: root, operator
  - DB: oracle, sybase
- Host:
  - UMA1: 128.138.0.0/255.255.0.0
  - UMA2: 128.138.243.0, 128.138.204.0/24, 128.138.242.0
  - SERVERS: master, mail, www, ns
- Definir un comando para:
  - Tipos de Shell
  - Kill
  - Reboot

Ejercicio de clase

- Reglas:

1. Los administradores de sistemas a tiempo completo pueden ejecutar cualquier cosa en cualquier máquina sin una contraseña.
2. Los administradores a tiempo parcial pueden ejecutar todo pero necesitan contraseña.
3. alumno puede ejecutar cualquier cosa en las máquinas de UMA1.
4. alumno1 puede ejecutar comandos como Oracle o Sybase sin contraseña en UMA2.
5. operador puede correr los comandos definidos en alias en cualquier máquina con el usuario privilegio OP
6. alice puede ejecutar cualquier cosas excepto en los servidores
7. bob puede ejecutar cualquier comando en el directorio /usr/bin/ de los servidores, excepto las SHELLS
8. Usuarios WEBMASTERS pueden ejecutar cualquier comando como usuario www (que es propietario de las páginas web) o simplemente su a www.

Bibliografía

- Sudoers: https://help.ubuntu.com/community/Sudoers
- Sudoers: https://manpages.ubuntu.com/manpages/focal/en/man5/sudoers.5.html
- Man sudoers: https://man7.org/linux/man-pages/man5/sudoers.5.html
- Sudoer File Examples
  http://www.softpanorama.org/Access_control/Sudo/sudoer_file_examples.shtml
- AppArmor: https://ubuntu.com/server/docs/security-apparmor
- AppArmor: https://debian-handbook.info/browse/es-ES/stable/sect.apparmor.html