Bisoños Usuarios de GNU/Linux de Mallorca y Alrededores | Bergantells Usuaris de GNU/Linux de Mallorca i Afegitons

Iniciación de sudo para bisoños (32040 lectures)

Per **Domingo Fiesta Segura**, ***C2H5OH*** (http://www.krenel.net)

Creado el 03/06/2003 14:30 modificado el 03/06/2003 14:30

En ambientes corporativos con muchos usuarios hay situaciones en las que determinados usuarios requieran privilegios de los que no disponen. Con sudo(1) podemos administrarlos sin tener que compartir contraseñas (como la de `root`).

Índice

1. Intro

El principal fin de *sudo* es reemplazar a *su*. En algunas circumstancias puede utilizarse como reemplazo del SUID. La ventaja principal es que no es necesario dar a conocer la contraseña de `root` o de algún otro usuario privilegiado, pudiendo ejecutar comandos como tales usuarios.

En éste artículo nos centraremos en el fichero `/etc/sudoers`, que es el fichero donde se guardan todas las reglas de acceso, los alias y las opciones por defecto. Para consultar las opciones disponibles des de la línea de comandos os recomiendo echarle un vistazo a su página man(14).

Para utilizar un comando con *sudo* es tan sencillo como:

```
etanol@botijo:~$ sudo comando
```

2. El fichero `/etc/sudoers`

Desde este fichero lo controlamos **todo**. A continuación una lista de las posibilidades que nos ofrece:

- Podemos crear alias de comandos, usuarios, usuarios privilegiados[1] y *hosts*.
- Podemos establecer opciones de comportamiento globales, por usuario, por usuario privilegiado y por *host*.

- La sintaxis del `/etc/sudoers` está pensada para entornos distribuídos; de forma que podemos gestionar toda una red con un único fichero.
- Y, cómo no, **flexibilidad** absoluta a la hora de crear reglas de acceso.

> **NOTA**
>
> Cuando nos referimos a **usuarios** y a **usuarios privilegiados** también es posible referirnos a UIDs, grupos o GIDs, *netgroups* o alias de usuario. Para ello basta anteponer el símbolo `#` para los UIDs (#1003), el símbolo `%` para los grupos (%cdrom) y el símbolo `+` para los *netgroups* (+secretarias).

Para editar `/etc/sudoers` **debemos** utilizar el comando `visudo` puesto que aparte de lanzar nuestro editor favorito realiza otras tareas adicionales como **bloquear** el fichero para evitar edición concurrente y **comprobar la sintaxis** antes de guardar los cambios.

2.1. Estructura

El `/etc/sudoers` se divide en tres grandes secciones:

```
#
# Definiciones de alias
#

#
# Ajuste de opciones por defecto
#

#
# Reglas de acceso
#
```

Todas son opcionales. Obviamente, la más necesaria es la última ya que sin ésta el uso de *sudo* no tiene sentido. Como se puede observar, los comentarios se insertan igual que en los *scripts* del *shell*.

2.2. Definiciones de alias

Los alias son abreviaciones para cualquier tipo de elemento: comandos, usuarios, usuarios privilegiados y *hosts*. Éstos alias pueden ser utilizados en **cualquier** lugar donde se espere un comando, un usuario privilegiado o un *host* respectivamente. Insisto, **cualquier** lugar; incluída la definición de un alias.

Veamos la sintaxis general y sus variantes.

```
Tipo_Alias     NOMBRE_ALIAS1 = elemento1, elemento2, elemento3
Tipo_Alias     NOMBRE_ALIAS2 = elemento1, elemento5 : NOMBRE_ALIAS3 = elemento4
Tipo_Alias     NOMBRE_ALIAS4 = elemento7, elemento2 :\
               NOMBRE_ALIAS5 = elemento6, NOMBRE_ALIAS1
```

- **Tipo_Alias:** Puede ser uno de los siguientes[2]:
  1. `Cmnd_Alias` para comandos
  2. `User_Alias` para usuarios
  3. `Runas_Alias` para usuarios privilegiados
  4. `Host_Alias` para *hosts*
- **NOMBRE_ALIAS:** Es el nombre del alias. Debe empezar por letra mayúscula y sólo se permiten letras mayúsculas y números.
- El resto son los elementos o listas de elementos por los cuales `NOMBRE_ALIAS` será expandido.

> **NOTA**
>
> Existe un alias especial, `ALL`, que se utiliza para englobar a **todos** los comandos, usuarios, usuarios privilegiados o *hosts*.

2.3. Ajuste de opciones

Como ya hemos dicho podemos definir opciones globalmente, por usuario, por usuario privilegiado y por *host*. La sintaxis es la siguiente:

```
Defaults                                lista_opciones
Defaults:usuario                        lista_opciones
Defaults>usuario_privilegiado           lista_opciones
Defaults@host                           lista_opciones
```

La `lista_opciones` es una lista de opciones (cómo no) separadas por comas. Existen cuatro tipos de opciones:

1. Booleanos: Que se activan con sólo escribir el nombre de la opción y se desactivan con el símbolo `!` delante.
2. Enteros: De la forma `nombre_opcion = valor`
3. Strings: Igual que los enteros `nombre_opcion = "valor"`
4. Listas: Que pueden ser de la forma `nombre_opcion = "valor1 valor2"`. Éstas opciones también pueden utilizar += y −= en lugar de = para añadir elementos y quitar elementos respectivamente.

> **NOTA**
>
> Algunas opciones pueden ser utilizasas en un contexto booleano lo que implica, entre otras cosas, que pueden ser deshabilitadas con el operador `!`. Para más detalles ver la página man[15].

2.4. Reglas de acceso

Ahora toca definir los usuarios a los que permitimos utilizar *sudo*, los comandos que pueden ejecutar, bajo qué usuarios privilegiados ejecutarán los comandos y en qué *hosts* pueden hacerlo.

```
usuario      host = (usuario_privilegiado) comando
```

Bastante simple, ¿verdad? Hay que decir que cada elemento puede ser tanto un alias como una lista de elementos. La mención del `usuario_privilegiado` o la lista de ellos es opcional y por defecto se toma el `root`. En los ejemplos comentaremos algunos detalles aclaratorios de la sintaxis.

Existe una última posibilidad, y es poder eliminar la petición de contraseña para ejecutar uno o varios comandos. Se trata de las etiquetas `NOPASSWD` y `PASSWD`. Son opcionales y por defecto se asume `PASSWD`.

```
usuario      host = (usuario_privilegiado) NOPASSWD: comando
```

[1] Aquí me refiero a los usuarios en los que nos queremos convertir cuando utilizemos *sudo*.

[2] Respetad las mayúsculas y las minúsculas.

3. Ejemplos

A continuación un ejemplo de `/etc/sudoers` sencillo. Con sus apropiadas explicaciones. No he incluído ejemplos de ajustes de opciones porque son características avanzadas que un usuario experto sabrá entender consultando página man[15].

```
#
# Aliases
#

# Comandos para grabar CD
Cmnd_Alias GRABARCD = /usr/bin/cdrecord, /usr/bin/cdrdao

# Comandos para administrar el sistema
Cmnd_Alias ADMIN = /usr/bin/apt-get, /usr/bin/dpkg
```

```
# Usuarios que pueden grabar CD
User_Alias GRABADORES = etanol, marta

# Usuarios privilegiados
Runas_Alias OP = root, operator

# Los hosts de mi red local
Host_Alias MIRED = botijo, solomillo

#
# Reglas de acceso
#

# Sólo pueden grabar CDs los usuarios permitidos y sólo en la máquina
# con grabadora (por defecto, como root). No necesitan contraseña.
GRABADORES      botijo = NOPASSWD: GRABARCD

# Sólo etanol, aparte del root, puede administrar el sistema en cualquier
# ordenador.
etanol          ALL = (OP) ADMIN

# Sólo los usuarios del grupo "vip" pueden apagar los ordenadores de mi red
%vip            MIRED = /sbin/shutdown, /sbin/poweroff, /sbin/halt, /sbin/reboot

# Los usuarios del grupo "cdrom" pueden ripear CDs
%cdrom          botijo,butterfly = /usr/bin/cdparanoia, /usr/bin/cdda2wav

# También tenemos webmasters que pueden ejecutar el apache como usuario "www"
# ¿Servirá de algo?
etanol,houzy,mkd  solomillo = (www) /usr/sbin/apachectl

# El root puede hacer lo que quiera.
root            ALL = (ALL) ALL
```

3.1. Aclaraciones

Cuando especificamos un `usuario_privilegiado` toma efecto a todos los comandos listados a partir de éste. Por ejemplo:

```
USUARIO    HOST = (fulano) comando1, comando2, (mengano) comando3
```

Esta línea permite a los usuarios del alias `USUARIO` en las máquinas del alias `HOST` ejecutar `comando1` y `comando2` como usuario `fulano` y ejecutar `comando3` como usuario `mengano`.

Lo mismo pasa con las etiquetas `NOPASSWD` y `PASSWD` que afectan a todos los comandos que las siguen. Entonces, utilizando el mismo ejemplo:

```
USUARIO    HOST = NOPASSWD: comando1, comando2, PASSWD: comando3
```

Ahora los usuario de `USUARIO` pueden ejecutar en `HOST` `comando1` y `comando2` sin tener que introducir su contraseña, al contrario que con `comando3`. Y, mucho ojo, los tres comandos se ejecutan como **`root`**.

4. Comentarios finales

4.1. Consideraciones de seguridad

En `/etc/sudoers` cuando se espeicifique un comando **siempre** se debe poner la **ruta completa** al ejecutable. Cuando ejecutamos *sudo* **no** es necesario, es decir, no necesito escribir `sudo /usr/bin/cdrecord` sino me basta con `sudo cdrecord`.

El uso del alias `ALL` para especificar comandos es altamente peligroso, supongo que las razones son obvias.

4.2. Funcionalidades que no se han comentado

Hay un varios detalles que no he mencionado. El primero es que en lugar de ejecutables se pueden indicar rutas a directorios para indicar que cualquier binario de dicho directorio se incluye en la regla de acceso; basta con finalizar la ruta en `/` y se entenderá como un directorio.

El segundo es que para los comandos podemos utilizar *wildcards* o caracteres comodín. Además, es posible controlar qué parametros le podemos pasar a los comandos también mediante caracteres comodín. De igual forma podemos controlar qué parametros **NO** podemos pasar a los comandos utilizando *wildcards* y el símbolo `!` al principio de la ruta.

Todas éstas funcionalidades están bien comentadas en la documentación correspondiente. Aquellos de vosotros interesados en ellas ya sabéis lo que toca: `man sudoers` ;-)

5. Enlaces

Todo lo que os cuento aquí lo he aprendido leyendo éste[16] y éste otro[17] artículos del señor Michael Lucas en su columnas Big Scary Daemons[18] sobre BSD. Tampoco dejéis de echarle un vistazo a las páginas man `sudo(8)`[19] y `sudoers(5)`[20].

Generado: lunes, 23-agosto-2004 2:40

Lista de enlaces de este artículo:

1. http://www.courtesan.com/sudo/
2. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=1#sec_1
3. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=2#sec_2
4. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=2#sec_2_1
5. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=2#sec_2_2
6. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=2#sec_2_3
7. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=2#sec_2_4
8. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=3#sec_3
9. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=3#sec_3_1
10. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=4#sec_4
11. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=4#sec_4_1
12. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=4#sec_4_2
13. http://bulma.net/body.phtml?nIdNoticia=1779&nIdPage=4#sec_5
14. http://www.courtesan.com/sudo/man/sudo.html#options
15. http://www.courtesan.com/sudo/man/sudoers.html#defaults
16. http://www.onlamp.com/pub/a/bsd/2002/08/29/Big_Scary_Daemons.html
17. http://www.onlamp.com/pub/a/bsd/2002/09/12/Big_Scary_Daemons.html
18. http://www.onlamp.com/pub/ct/13
19. http://www.courtesan.com/sudo/man/sudo.html
20. http://www.courtesan.com/sudo/man/sudoers.html

E-mail del autor: etanol _ARROBA_ krenel.net