**#2 Use gsudo to elevate individual tabs and commands in Windows Terminal**

gsudo is an open-source project on GitHub that provides a sudo equivalent for Windows. It lets you run individual commands with elevated privileges; and elevate the current shell (tab) in a Windows Terminal window or in a new window. If you use gsudo to elevate the current shell (tab) in a Windows Terminal window, then you should take into consideration the security concerns that I outlined above.

* Open Windows Terminal. By default, you should be in a PowerShell tab. If not, open one.
* Open the gsudo GitHub page [here](https://github.com/gerardog/gsudo).
* Scroll down to the section on *Installation*. Copy the PowerShell command line, paste the command into Windows Terminal, and press **ENTER**. You will see a warning about pasting a command that has multiple lines. You can safely ignore it and click **Paste anyway**.
* You’ll be prompted to create an alias ‘sudo’ for gsudo. This is optional but if want to create the alias, type **y** and press **ENTER**. UAC will require you provide consent to make this change.

Text

Description automatically generated

The command runs a script that downloads gsudo, unzips it, and places it on your PC so that you can access gsudo from the command line and Windows Terminal.

* Now close Windows Terminal and reopen it.
* Once Windows Terminal has restarted, you can use gsudo to elevate any PowerShell or command line tab. Just type **gsudo**followed by the command you want to run with elevated privileges, and press **ENTER**. In the example below, I use gsudo to open the hosts file.

gsudo notepad %windir%\system32\drivers\etc\hosts

* Or type **gsudo** and press **ENTER** to elevate all further commands in the tab. Note that the tab label is now prefixed with ‘Administrator’.

Text

Description automatically generated

I’ve found gsudo to work reliably in command line tabs in Windows Terminal. But in Windows PowerShell or PowerShell 7 tabs, calling apps like Notepad with gsudo throws an error. But as providing you stick to running PowerShell cmdlets, like the one shown below, gsudo seems to work just fine.

New-Item -ItemType directory -Path 'c:\program files\myapps'

Text

Description automatically generated

For a full description of what you can do with gsudo, check out the project’s GitHub page [here](https://github.com/gerardog/gsudo).