

I'm not a robot





























The accepted answer is right, but the policy modification is only available for the currently running instance of the Powershell, meaning once the instance of the Powershell is shut down. The policy will be reset. If a user reopens another instance of Powershell, the default policy will be applied which is Restricted For me, I need to use the VisualStudio Code console and g++ from cygwin to build things. The console is using Powershell, with the default policy, nothing can be done. One solution is changing the policy everytime the console is fired in VisualStudio Code console, maybe a script of changing the policy. I am lazy, so another solution is when I run the Powershell in admin mode, similar to what the accepted answer does, but with an extra parameter which changes values in the Registry table. Once it been done. Other instances of Powershell will use the RemoteSigned policy by default. set-executionpolicy remotesigned -Scope CurrentUser I'm afraid it's not quite that simple. Environment variables are not limited by scope, as you suggest, but you are right that the lifetime of the value in the variable is different when comparing the verbs. set modifies the current shell's (the window's) environment values, and the change is available immediately, but it is temporary. The change will not affect other shells that are running, and as soon as you close the shell, the new value is lost until such time as you run set again. setx modifies the value permanently, which affects all future shells, but does not modify the environment of the shells already running. You have to exit the shell and reopen it before the change will be available, but the value will remain modified until you change it again. See here for an example: I want to set an environment variable from the command prompt and then be able to access it globally (for instance, I should see it by going to System -> Environment Variables). When I use the set command, it isn't accessible in a new cmd session. set NEWVAR=SOMETHING echo %NEWVAR% Related questions: 2

- <https://musicandyouapp.com/webupload/editorimg/files/75802061166.pdf>
- <https://trainova.com/userfiles/file/a899ec0b-52fb-4a07-a9ee-d4f28cc21428.pdf>
- <http://ingfrimale.com/userfiles/files/xeroxitulos.pdf>
- <https://grewalkennels.com/adhar/uploads/file/28282222186.pdf>
- [https://siltri.com/fck\\_upload/file/bb2a9171-e7e6-428a-a174-bf41c7c74de7.pdf](https://siltri.com/fck_upload/file/bb2a9171-e7e6-428a-a174-bf41c7c74de7.pdf)
- civapafuye
- games affected by 9/11
- <http://dynamicyunnan.com/ckfinder/userfiles/files/3fa8853a-2f6a-4c51-a1c4-120e6624f8f8.pdf>
- <http://main-target.com/userfiles/files/55574946618.pdf>
- <https://e-uchebnici.com/img/file/01afa520-ef97-4510-a4cf-0d64e48fcfa7.pdf>