

# Kann Datei nicht löschen!

Beitrag von „byebye123“ vom 30. Dezember 2023, 09:34

Diese Art Probleme sind immer mit `chown` und `chmod` lösbar.....

Es geht um Berechtigungen...

`chown` steht für `change owner` und erlaubt das Ändern des Eigentümer-Benutzers und/oder der Eigentümer-Gruppe von Dateien.

The following methods are valid (and probably the best options for shell scripting):

Code

1. `id -u #` returns UID
- 2.
3. `id -un #` returns username
- 4.
5. `id -g #` returns primary (a.k.a. effective) GID
- 6.
7. `id -gn #` returns primary Group Name

See `man id` for more options.

To answer this specific question if I was writing a script that would work in any environment, I wouldn't assume that the default group `staff` is in use and instead I would do the following if I was writing a script that could be run for any user (whether or not they modified their system):

Code

1. `chown "$(id -un):$(id -gn)" some_file`

It's always better to be platform agnostic whenever possible (if it's not too much of a time tradeoff), as you will incur less technical debt later.

Please note that `chown's -R` flag should be used with the utmost confidence. You could really mess up your system (Although it is worth noting that Apple made it hard to do since they implemented [System Integrity Protection](#)).

Consider, for instance, if you were to `chown -R 755 /var` instead of `chown -R 755 /var/www` as root with [System Integrity Protection](#) disabled I believe that you would be in for it. I

can't tell you what would break first, but it would break first on a Darwin system, but please take my word that it wouldn't be pretty.