

Taking ownership of a file doesn't give you access, yet

 devblogs.microsoft.com/oldnewthing/20150610-00

June 10, 2015



Raymond Chen

A colleague of mine accidentally ran a command from an elevated command prompt rather than a normal unelevated command prompt. By default, files created from an elevated command prompt are owned by the Administrators group, on the theory that you are doing work in the elevated command prompt in your rôle as the system administrator, so the things you are doing are done on behalf of all the administrators of the system. (If you don't like this you can use the [Default owner for objects created by members of the Administrators group](#) policy to change the default.)

My colleague attempted to repair the damage by taking ownership of all the files back: From the root of the directory tree of files that got created with the wrong owner, he ran `takeown /f . /r` to take them back. (You naturally have to do this from an elevated command prompt. Since you left off the `/A` flag, this assigns ownership to you personally, rather than to the administrators group.)

“I can confirm that I am the owner by using the `dir /q` command, but I still don't have access. What gives?”

Having ownership of a file does not automatically grant you full access. [Users always have WRITE_DAC permission on objects they own](#), but that's all. If you want more, you need to leverage `WRITE_DAC` into full access.

Fortunately, leveraging `WRITE_DAC` into full access is easy, because `WRITE_DAC` gives you permission to change permissions, so you can just change the permission to grant yourself full access:

```
cacls /e /g domain\user:F
```

[Raymond Chen](#)

Follow

