Shell policy is not the same as security

devblogs.microsoft.com/oldnewthing/20080131-00

January 31, 2008

Raymond Chen

Mark Russinovich pointed out that if you let users run arbitrary programs, they can circumvent policies. This is actually not surprising, because policy is not the same as security. Shell policies control how Explorer and other shell components behave, but that's just blocking the front door. For example, there is a shell policy to prevent the user from changing the wallpaper from the Desktop control panel. This disables the controls on the Desktop control panel for changing the wallpaper, but there are ways to change the wallpaper other than that. If users can run an arbitrary program, then they can run a program that calls `SystemParametersInfo(SPI_SETDESKWALLPAPER)` to change the wallpaper directly, bypassing the shell. The purpose of the shell policies is merely to make it more difficult for users to perform various categories of operations by removing them from the shell interface. But, of course, if the users are allowed to write their own program with its own user interface, then they can still access the underlying functionality. Setting a policy to remove the user interface for a feature is like removing the staircase that leads to the second floor to keep people out. If you let them bring a ladder, then they can still get up there. Mark Russinovich points out that policies are enforced at the application level, and since applications run as the user, the user can run a program that commandeers the application and patches out the code that checks the policy setting. Shell policies are just for modifying the user interface. If you want to block an operation even from users who bypass the normal user interface, you have to block it at a level below the user interface. For example, you might revoke write permission to the relevant registry key; that way, even if the user manages to run their own code on the machine, they still can't change the underlying setting.

Every so often, somebody who doesn't understand the difference between shell policy and security submits a security vulnerability report to MSRC saying, "Check this out, I can set the policy to prevent the user from changing the desktop wallpaper via the shell, but through this clever technique of injecting code into Explorer and patching the binary in memory, I can change the desktop anyway!" Well yeah, but why go to all that effort? Just write a program that changes the desktop wallpaper already.

Raymond Chen

Follow