

Killing Administrator

Solving Three Problems by Eliminating Administrative Rights
By Greg Shields

Killing Administrator

It's high time we killed Administrator.

Now before you run for the authorities, we're not talking about killing *the* Administrator. That would be you, and that would be wrong. Rather, it's time we eliminated *the role of* Administrator from our Windows servers and desktops.

Why? Inside Microsoft Windows, the notion of Administrator and administrator rights was never well designed in the first place. With that design, we've long been suffering under what amounts to a binary view of privileges: *Do you have administrator rights or don't you* have been the only options at our disposal.

Yet simply killing off Administrator doesn't solve the problem. Something must fill the hole its death leaves behind. In its place, IT dreams of a more granular approach to privilege management that aligns the actions users want to accomplish with those that you've specifically permitted.

Eliminating "the Administrator" from administrator rights solves three big problems that have plagued Windows administration for years. For the first, moving rights management away from its historical person-based approach brings us closer to the goal we've been dreaming about: *Least Privilege*.

Problem number one is figuring out how to get there.

Getting to Least Privilege requires a new approach, along with a new set of tools. That's why solving the second problem starts by flushing the binary notion of Administrator versus non-Administrator from our minds. Doing so frees us to think granularly about how users can be mapped to actions based on policies. **Problem number two** is all about getting that granularity.

As you can imagine, lots of granularity means lots of potential actions to catalog. Least Privilege won't build itself. That's why the third problem's resolution requires some way to share those rules that work best. **Problem number three** is in finding those rules that work.

Let's take a closer look at these three big problems that get solved with the Administrator's demise. With a little effort and the right tools, you'll find that making the move to granular privilege management never before seemed so possible.

Problem #1: Getting to Least Privilege

Don is a software developer in a mid-sized company, but his company isn't in the software business. They sell tires. While most of his co-workers deal with the day-to-day activities associated with tires and their sale, Don's position is unique. He's tasked with creating software solutions that enable the others to do their job.

There's just one problem. Don's company places a high value on security. So high, in fact, that very few people are granted Administrator rights. Even Don can't get them. Getting anything done that requires elevated rights requires calling the Help Desk, and often a lengthy wait.

While Don admits most salespeople don't really need elevated rights, he thinks he most certainly does. Without them he can't install his developer tools, which require regular installation as he updates code, runs tests, and resets everything for the next round. While his company will never allow him Administrator, he dreams every day about just getting a piece of it...

While perhaps a bit draconian in its implementation, Don's company is well within its rights to heavily restrict administrator rights. Doing so eliminates the possibility that inappropriate software gets onto systems. It also reduces the probability of malware infection, since applications and their configurations are tightly controlled. Maintaining configuration control for regulatory compliance represents yet another valid reason.

Yet his company's rights management means he can't get his job done. His productivity is severely impacted, costing the company untold dollars in wasted time.

Don's desire for "just a piece" of Administrator sounds very similar to the granular approach most IT professionals dream about. That granular approach was first considered way back in 1974 when it was called the Principle of Least Privilege. While that principle's exact words are unimportant for this conversation, know that Least Privilege desires to give a person only those rights that are absolutely essential to accomplishing their assigned tasks.

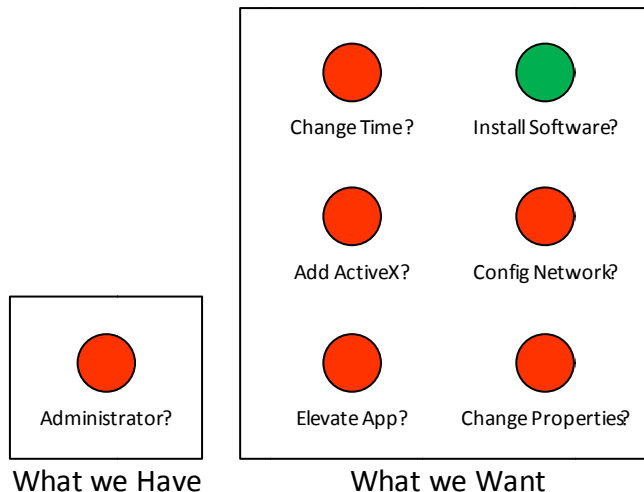


Figure 1: The rights we have versus the rights we want.

Implementing the Least Privilege approach means thinking outside the box of the rights we currently have with Windows today. Figure 1 shows that box in relation to Don’s plight. Lacking Administrator, he’s not a productive employee. But he at the same time doesn’t need to perform every action that Administrator brings. All he needs is the ability to install software, not to necessarily elevate applications or change system properties.

What he wants – and what Least Privilege requires – starts by fracturing all the possible actions a person might need to do. Once broken down, that person can be specifically assigned only those actions that meet the requirements of their job. You can’t get this out of Windows alone. You need additional tools on both servers and desktops that interact with a centralized management infrastructure to get there.

Problem #2: Getting the Necessary Granularity

Eloise was just promoted to security officer at her not-quite-small company. Charged with new responsibilities, she immediately aims to eliminate administrator rights all around the network. Doing so, she believes, will overcome many of the daily problems faced by IT. You know the use cases: Inappropriate and unlicensed software in places it’s not supposed to be, users who break computers by doing things they shouldn’t, not to mention IT’s inability to control its desktop configuration.

She’ll be the star of the company, she thinks, as she ponders that future state.

But beginning the project one day she quickly finds that pulling those rights isn’t as trivial as it seems. Some users actually need them, or at least some portion thereof. Others might not, but their applications do. Removing rights from the person also removes them from poorly-written but necessary applications. When those applications stop functioning, people get unhappy.

To get secure – and earn the promotion she was just handed – Eloise quickly realizes she needs a better approach...

Eliminating administrator rights isn't a project that will happen overnight. Developers need application installations. Users on the road require special consideration. Even the applications themselves are affected when they're not properly coded and require elevation. Your steps required to move from Administrator-everywhere to Administrator-nowhere are going to take time.

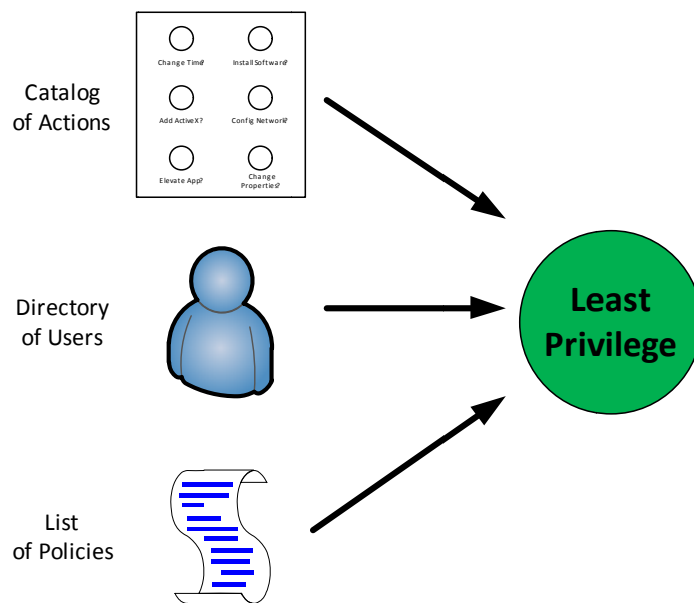


Figure 2: Actions, users, and policies are required.

One of the biggest consumers of that time will be figuring out the mapping between users, actions, and company policies. These three elements must come together to create the Least Privilege environment that replaces the death of Administrator. Figure 2 shows how those three elements interrelate.

What first you require is a catalog of the possible actions any user might need to accomplish. These are actions like changing the time, installing software, adding ActiveX controls, changing system properties, and elevating applications, among others. That catalog represents your master list of actions you'll eventually provision to users. *An effective privilege management solution will deliver this catalog via its administrative console.*

Your next requirement is a directory of the users who perform the actions you've cataloged. *That directory you already have.* For most of us, the directory our users already authenticate against is Microsoft's Active Directory. If your business has been operating for really any period of time, you also have the groups and organizational units that map users to their job roles. Finance users, for example, are in the Finance group, while the sales team occupies the Sales group. You've been using these groups already for assigning rights to files and folders. Now, with a privilege management solution, you're merely extending their use to determine the actions each group will be allowed to accomplish.

The final requirement isn't technical in nature, but procedural. Your business is unique compared to every other business out there, which means that its policies are unique as well. Yet while you probably have a general understanding of those policies, they may not be documented in exactly the same format as your catalog of actions. *Gathering your lists of policies and translating them into user actions is the final step in this process.*

The integration of these three elements is what defines privilege management that follows Least Privilege. The granularity gained by their separation is what enables you to balance, for example, Eloise's desire for lockdown with her company's requirement for productivity.

Problem #3: Getting Rules that Work

Chris manages the sales team at his medium-sized company. He's a great employee with a long track record of performance. Doing his job requires plenty of phone time, and not a little bit of travel. It also requires a set of applications to which few in the company have access. Those applications report on sales data, and include plenty of sensitive information that would hurt his company if it ever got out.

Problem is that Chris also has another application he likes to use on his company laptop. His file sharing application he keeps around because it brings him music and movies for those long work trips. Having administrative rights for a while, he used them to install that application to his laptop. He's never really considered how "file sharing" and "sensitive data" might not go well together on the same corporate laptop.

The company announced today that they're getting rid of administrator rights. Yet Chris knows both his sales and file sharing application require them to even start up. He needs those rights to do his job. He also hopes he can continue file sharing once they've made the change...

Locking down users and applications might seem the easy project at first blush: Just pull rights from users and watch the network grow secure. The reality, however, is far from trivial. Once you begin digging into the instances of actions you'll assign, you'll quickly find there's a lot of work required before you can see the project finished.

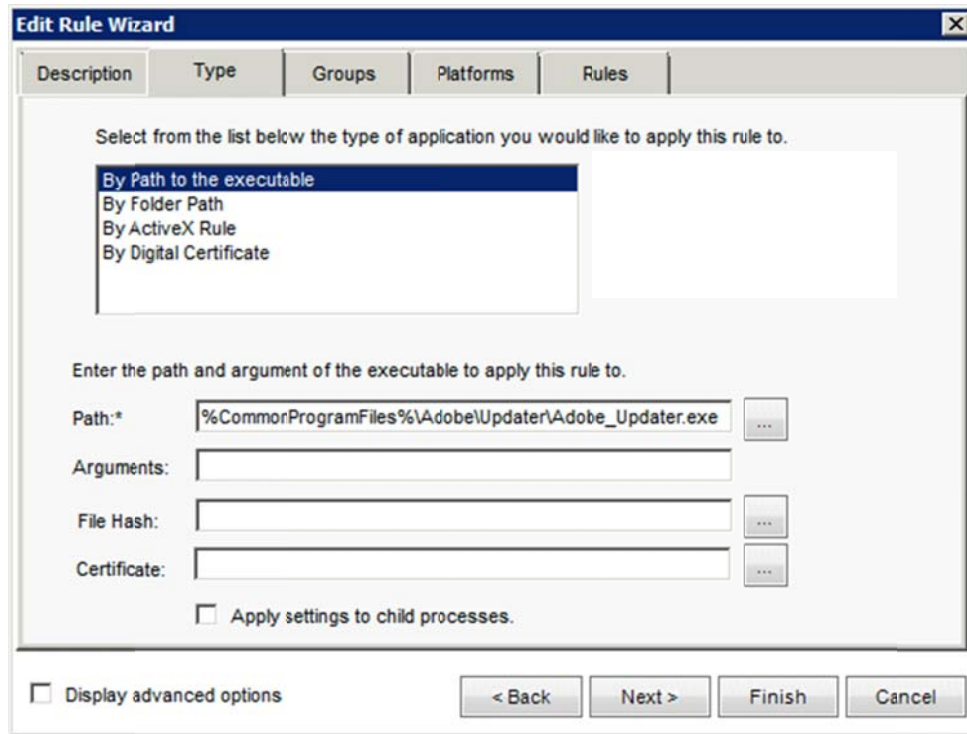


Figure 3: Creating a rule to allow an application.

Just installing a privilege management solution doesn't automatically bring Least Privilege to the network. Any solution is a framework within which rules must be created that map users to policy-approved actions. Chris' situation is fairly common during this period of rule construction: You the IT professional want to create a whitelist of only the applications approved for execution and/or elevation. Figure 3 shows one way such a rule might be created.

In that figure you can see how the application's executable can be referenced by path, file hash, or digital certificate. Each has its pros and cons that must be weighed when building the rule. In Chris' case, denying all applications by default and then specifically allowing those that are approved means preserving his sales application while shutting down his file sharing security hole.

Yet even the smallest business in operation today uses many different applications. A business of ten people might require fifty applications to get the job done. *Getting the rules that work sometimes requires the assistance of an entire community.*

That's why the privilege management solution you want should include some way to share your rules with others. With a clearinghouse of effective rules, populated by others in similar situations, you can quickly identify those that have worked for them. With businesses worldwide looking to eradicate their administrator problem, why reinvent the wheel all by yourself.

Privilege Management is the Death of Administrator

Evolving your network's security approach to one that follows Least Privilege is a worthy goal. Getting you there are privilege management solutions that wrap around Microsoft Windows and Active Directory to enable the granularity you've read about here. A solution you'll want starts with a full catalog of actions from which to create rules. That same solution integrates with your Active Directory data to identify users and groups. It also includes collaboration tools that help shorten deployment time and increase rule quality.

These solutions exist today. Implementing them starts by killing Administrator.

About the Author:

Greg is a Senior Partner and Principal Technologist with Concentrated Technology. He is a Contributing Editor for TechNet Magazine and Redmond Magazine, and a Series Editor for Realtime Publishers. Greg is a sought-after and top-ranked speaker, seen regularly at conferences like TechMentor, Tech Ed, VMworld, and more. He is a multiple recipient of Microsoft "Most Valuable Professional" award with has received VMware's vExpert award.

Sponsored by *Privilege Authority* from



Now you can grant user accounts the least privileges necessary according to best practices, yet elevate specific applications and ActiveX controls as needed. Determine user rights for just those applications, features, and controls you choose with the low-cost solution to Elevating User Rights.

Visit www.scriptlogic.com/pa for more information.