Realtime publishers

The Essentials Series: Tackling Active Directory's Four Biggest Challenges

Quickly Recovering Deleted Active Directory Objects

sponsored by

QUEST SOFTWARE®

by Greg Shields

Copyright Statement

Quickly Recovering Deleted Active Directory Objects

It's happened to your domain before. You walk back in after a long lunch or notice something peculiar as you're packing up for a long weekend. Users are calling in to complain that they can't access their resources. They're seeing errors all across the board when attempting to accomplish anything on their desktops. It's like someone has…deleted…their accounts. Or, even worse, someone has deleted an entire Organizational Unit (OU) full of them. At this point, you find yourself setting down your leftovers or your coat, realizing that you're about to be at work quite a bit later than you expected this evening.

For a platform as powerful and pervasive as Active Directory (AD), it's a wonder how easy it remains to accidentally destroy huge sections of it. Whether you're a low-level Help desk administrator or an all-powerful Domain Admin, most IT professionals have enough rights to create this kind of havoc. Deleting a user account or an OU full of them is a particularly big problem. A problem that even today doesn't have a good resolution using native Windows tools alone.

The Native Solution

If this has happened to you, here's a glimpse of the process you're forced to undertake:

- **Step1:** Locate a Domain Controller that is also a Global Catalog (GC) and disconnect this server from the network.
- **Step 2:** Reboot the server into Directory Services Restore Mode and login using that server's unique DSRM password. On servers previous to Windows Server 2008, this can be done by repeatedly pressing the F8 key during the boot cycle to bring forward the boot menu. The process changes in Windows Server 2008, requiring you to run the *MSCONFIG* utility before rebooting. Once launched, navigate to the *Boot* tab and under the *Boot Options* selection box, choose to reboot into *Safe boot | Active Directory repair*.
- **Step 3:** Restore the AD database—also known as a System State Restore—to the Domain Controller from a recent backup before the deletion event. This non-authoritative restore will not actually restore any deleted objects. It only enables old objects that have been deleted to be selectively (authoritatively) restored in the next step.
- **Step 4:** Perform an authoritative restore on the object or objects that have been deleted. This involves launching the NTDSUTIL tool with the command *ntdsutil "authoritative restore"*. Then, restore the specific object with the command *restore subtree <objectDn>*. The value for *<objectDN>* will be the distinguished name (DN) of the object or container that was deleted.

- **Step 5:** Reconnect the Domain Controller to the network and reboot it back into normal mode. Ensure that the restored object or objects replicate correctly to all Domain Controllers in the domain.
- **Step 6:** As the Domain Controller reboots, it will create a series of .LDF files. These files include the necessary "back-link" information, which can be used to restore the groups of which the deleted object was a member. Restore those back-links for each object using the command *ldifde –i –k –f <ldfFile>*. The file name structure of these .LDF files will resemble *ar_<date>-<time>_links_<domainName>.ldf*.

This process appears relatively trivial until you realize one piece of missing information. To properly restore a deleted object in Step 4, you must know the DN of that object. Thus, in order to know its DN, you must first know which objects were deleted. If an entire OU of objects was deleted, you'll need to know each of the objects to individually restore. For a large swath of deleted objects, this process can be complex to the point of absurdity.

Even Windows Server 2008 R2's Recycle Bin Is Insufficient

At first blush, Windows Server 2008 R2's new Recycle Bin feature might seem like a good solution. The new AD Recycle Bin provides a mechanism to bring back objects that have been deleted from your AD. However, it's important to know that AD's Recycle Bin isn't like the one you see on your desktop. Enabling its functionality requires upgrading your entire AD forest to the Windows Server 2008 R2 forest functional level. This means that each and every Domain Controller must also be at Windows Server 2008 R2, a process that can take a long time to complete. Even more challenging, actually using the AD Recycle Bin requires the use of scripting to restore an object. So, the less-experienced admins who are more likely to cause a deletion problem are less likely to be able to fix it.

The Need for Better Tools

Organizations that base their computing infrastructure on AD cannot afford the lengthy and error-ridden process of manually restoring objects after an accidental mouse click. Yet this manual process is the only way to solve the problem with Windows' native tools. With the right third-party tools in place, finding and restoring deleted AD objects can be as easy as the few mouse clicks that deleted them in the first place. These tools take and catalog regular snapshots of AD structure and objects, allowing administrators to look backwards in time. They provide a way to compare previous with current snapshots to quickly find and automatically restore lost objects.

Yet the problem doesn't stop there. Best-in-class third-party tools provide much-needed further protections. They enable the additional capability to quickly restore your entire AD domain or forest in the case of corruption or malicious activity. Having this emergency restore capability in your back pocket is critical because every deleted object is equal to a worker's lost productivity, and every downed domain means an entire business going down. As such, implementing the right tools for comprehensive deletion protection is critical to turning major problems into minor restores.