

Organize Your Active Directory Objects

Tony Murray

(Reprinted from WindowsItPro Magazine)

Do you have a good handle on which objects in your directory are no longer used? Do you know exactly who you need to contact when making changes to the content or structure of your forest?

As a consultant specializing in Active Directory (AD), I come across many AD implementations that have grown organically over time. Typically, these implementations contain a large number of unused objects, as well as objects that are obviously in use, but who or what is using them isn't clear. It's costly having objects in this state: Periodic cleanups of AD become labor intensive and expensive, AD restructures or migrations become more complex, and even simple change management becomes more difficult.

To gain control over your AD environment, you need to deal with three key elements of object lifecycle management. The first is determining the appropriate way to provision, re-provision, and de-provision objects. The second is setting up controls so that all new objects conform to the provisioning methodology. The third is the sometimes arduous and time-consuming work of cleaning up existing objects so they either conform to the methodology or can be deleted from AD.

In this article I provide advice and tips that will assist you with the first two aspects by introducing the concept of guardianship of AD objects. By associating real people (guardians) with AD objects, you can gain greater control over your AD environment. I also offer some examples to assist you with the clean-up task.

Clarifying Terminology

The "guardian" for an AD object is the human being directly responsible for, or most closely associated with, that object. A better term might be "contact," but I'll avoid that because it's already used to represent a specific type of object in AD. Another term might be "owner," but this too has meaning in AD security in the context of the creator/owner of an object.

It would be handy if there were an AD attribute named "guardian" that we could use for setting guardianship of different types of AD objects. Unfortunately, there isn't so we must either create a new attribute (which involves extending the schema), or using an existing attribute from the default AD schema. For simplicity and because most organizations have a healthy aversion to extending the schema, I use existing attributes as described in the sections below.

Benefits of Guardianship

Identifying and removing unused objects in AD can be a thankless and time consuming task. Some helpful tools can assist you with finding unused objects (the Windows command-line tool dsquery is one; AdFind and OldCmp from Joeware are others), but because object deletion is potentially damaging to systems and applications that leverage AD you need to be 100 percent sure that that you're dealing with an unused object before you delete it. Typically you'll need to check with the person currently responsible for that object.

In many cases this person isn't easily identifiable from the object's attributes. You might have only the object name to work with (e.g., a group named "OKP100 Staff"). This is fine if OKP100 means something to you, but otherwise it's no help at all. The object's description might contain some

Organize Your Active Directory Objects

Tony Murray

(Reprinted from WindowsItPro Magazine)

information (e.g., “See JP Carter before making changes”), but what if JP Carter no longer works for the organization?

As you can see, no magic, in-built feature automatically links a human owner to an AD object. It's something that you have to implement for yourself. This is where the guardianship concept can help you.

Guardianship can also assist you when working with active objects. For example, when processing a request to add a user to a group, your operational staff can refer to the guardian to approve or decline the request.

The suggestions I make for setting guardianship of objects described below all assume that you will use AD as the repository for guardianship information. The same concepts (but clearly different methodology) apply if you already have a tool in place for provisioning AD objects and that tool is capable of storing the required guardianship information.

Setting Guardianship for User Objects

Organizations use user objects for a range of different purposes. Aside from standard user accounts directly associated with a warm body, user objects can be created for shared accounts, resource accounts (for mailboxes such as meeting rooms), service accounts, and secondary accounts for administrative purposes.

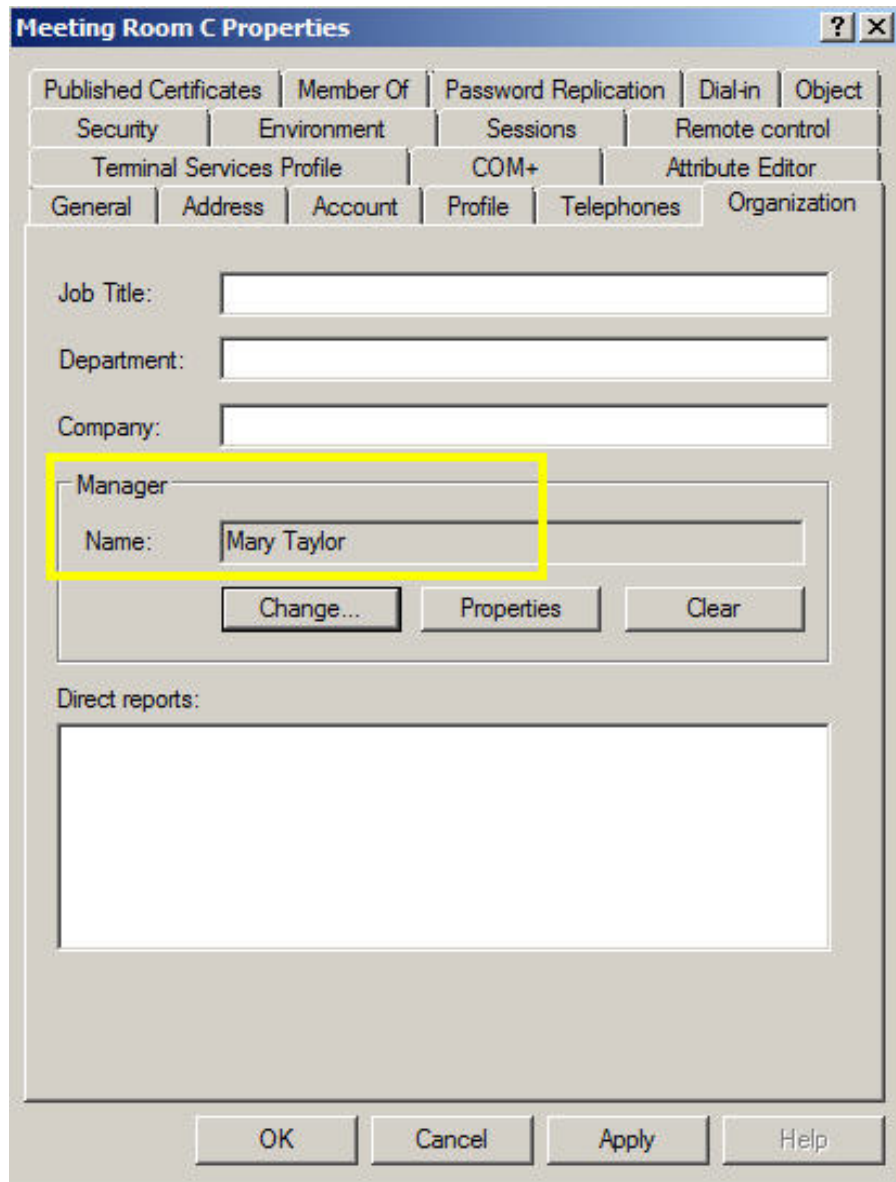
For all types of user objects, I recommend associating a guardian by setting the value of the manager attribute. Let's look at an example in which we have a resource account for a meeting-room mailbox named Meeting Room C. We want to set the guardian to be Mary Taylor.

From within the Active Directory Users and Computers MMC snap-in, find Meeting Room C, open up the properties and select the Organization tab. From here, click Change within the Manager section and use the object picker to find and add Mary Taylor's user account, as Figure 1 shows.

Organize Your Active Directory Objects

Tony Murray

(Reprinted from WindowsItPro Magazine)



The manager attribute is a linked attribute. The manager attribute is the forward link, while directReports is the corresponding back-link attribute.

Because the attributes are linked, when I set Mary Taylor as Meeting Room C's manager, Mary Taylor's user object shows Meeting Room C as a direct report, which Figure 2 shows. The main advantage of using a linked attribute is that the link object can be renamed or moved within AD, and the link remains intact. The link can only be broken if either the forward or back-link object is deleted. Another advantage of the linked attribute is that it lets you search AD for the relationship using either the guardian or the object(s) for which the guardian is responsible.

Organize Your Active Directory Objects

Tony Murray

(Reprinted from WindowsItPro Magazine)

The screenshot shows the 'Mary Taylor Properties' dialog box with the 'Organization' tab selected. The 'Direct reports' list is highlighted with a yellow box and contains the following items:

- Conference Room 1
- Meeting Room A
- Meeting Room B
- Meeting Room C

Below are examples of such searches using the AdFind tool from www.joeware.net. The first example shows a search for all user accounts for which Mary Taylor is the guardian:

```
C:\>adfind -list -b "CN=Mary Taylor,OU=Standard User Accounts, DC=ad,DC=fisheagle,DC=net" directReports
```

Organize Your Active Directory Objects

Tony Murray

(Reprinted from WindowsItPro Magazine)

Figure 3 below shows the results of that search:

Figure 3: Results from a search of user accounts:

CN=Meeting	Room	C,OU=Resource	Accounts,DC=ad,DC=fisheagle,DC=net
CN=Meeting	Room	B,OU=Resource	Accounts,DC=ad,DC=fisheagle,DC=net
CN=Meeting	Room	A,OU=Resource	Accounts,DC=ad,DC=fisheagle,DC=net
CN=Conference Room 1,	OU=Resource	Accounts,	DC=ad,DC=fisheagle,DC=net

The second example shows a search for the guardian of a meeting room:

```
C:\>adfind -list -b "CN=Meeting Room C,OU=Resource  
Accounts,DC=ad,DC=fisheagle,DC=net" manager
```

And the results of that second search look like this: CN=Mary Taylor,OU=Standard User Accounts,DC=ad,DC=fisheagle,DC=net.

Setting Guardianship for Group Objects

The manager and directReports linked-attribute pair isn't available for use with groups. Instead, I recommend using a similar pair of linked attributes named managedBy and managedObjects.

Let's look at an example in which we have a group named Consulting Team. We want to set the guardian to be Mary Taylor. To do this, locate the group within Active Directory Users and Computers, open the properties and select the Managed By tab. From here, click change within the Name section and use the object picker to find and add Mary Taylor's user account, which you can see in Figure 4.

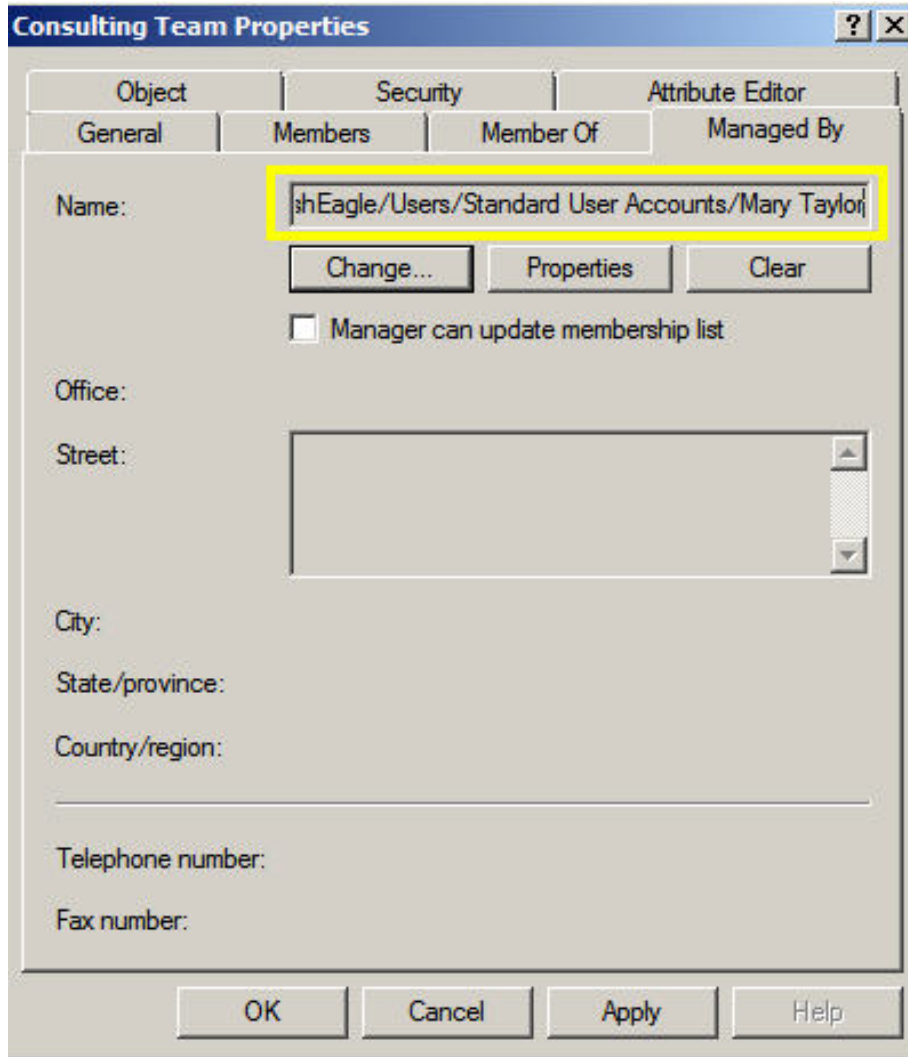
When you make the change in Active Directory Users and Computers, AD sets the value of managedBy on the group object to be the distinguished name (DN) of Mary Taylor's account. Note that when setting the Managed By value, you have the option to select Manager can update membership list, as Figure 4 shows. If you want only to assign guardianship for informational purposes, then you probably don't want to select the option, but it otherwise provides a shortcut method of assigning delegated management of the group membership to the guardian.

Be aware that the managedObjects back link isn't visible in Active Directory Users and Computers in the way that directReports is. To view the back link, you need to use LDAP queries or tools such as ADSIEdit or the new Attribute Editor in Windows Server 2008.

Organize Your Active Directory Objects

Tony Murray

(Reprinted from WindowsItPro Magazine)



As with the manager and directReports user attributes, you can search for the relationship between the group and its guardian by querying AD using the guardian, as this AdFind example shows:

```
C:\>adfind -list -b "CN=Mary Taylor,OU=Standard User Accounts,DC=ad,DC=fisheagle,DC=net" managedObjects
```

Figure 5 below shows the results:

```
Figure 5: Results from search querying AD using the guardian
CN=Management Team,OU=Groups,DC=ad,DC=fisheagle,DC=net
CN=Project Management Team,OU=Groups,DC=ad,DC=fisheagle,DC=net
CN=Consulting Team,OU=Groups,DC=ad,DC=fisheagle,DC=net
```

Organize Your Active Directory Objects

Tony Murray

(Reprinted from WindowsItPro Magazine)

Or you can query AD by using the object for which the guardian is responsible, as the AdFind example below shows:

```
C:\>adfind -list -b "CN=Consulting  
Team,OU=Groups,DC=ad,DC=fisheagle,DC=net" managedBy
```

Here are the results of that search: CN=Mary Taylor,OU=Standard User Accounts,DC=ad,DC=fisheagle,DC=net.

Setting Guardianship for Other Object Types

You can extend the concept of ownership to include any other type of AD object. For example, computer and organizational unit (OU) objects spring to mind as likely candidates. Both of these support the use of the managedBy and managedObjects linked attributes, so you can use the same method shown for groups above to define the guardian relationship.

Guardianship and Object Lifecycle Management

The guardianship concept works best if it's incorporated into your organization's provisioning and de-provisioning procedures. It's important to set the guardian whenever you provision an AD object that you want to keep track of.

Similarly, when you de-provision a standard user account, you should ensure that any guardianship relationships associated with that user are either removed or transferred to another user account. For example, if Mary Taylor from our scenario above leaves the company, I would need to consider what to do with the objects for which she is guardian. In the case of groups and meeting rooms, I would probably transfer the guardianship to another account. If Mary has a secondary account for administrative purposes for which she is guardian I would probably de-provision that at the same time.

Also bear in mind that that people often change roles within an organization. When this happens, your re-provisioning procedures should reflect the fact that someone may no longer be the appropriate guardian for AD objects.

Cleaning Up Your Existing AD Infrastructure

With a working guardianship in place for newly provisioned objects, you need to address the task of identifying existing objects that have no guardian, then configure the appropriate guardian relationships. Before you do that, however, you should remove any objects from AD that are no longer required.

Within AD, objects can be created for a specific reason and simply forgotten. For example, accounts and groups might be created to support a new document-management system. If the organization decommissions the document-management system a few years down the line, the associated accounts and groups might not be removed in parallel. These stale objects could linger in AD indefinitely, or until someone questions their existence.

Below are some examples of using command-line tools to find unused objects. These are by no means comprehensive, but should provide you with a starting point. I use AdFind in all three

Organize Your Active Directory Objects

Tony Murray

(Reprinted from WindowsItPro Magazine)

examples for consistency, but I could also have used OldCMP or the built-in dsquery tool for the inactive user and computer object searches.

Finding Inactive User Objects

The example below, written as one line, uses AdFind to search for user accounts that have either never logged on to the domain or haven't logged on to the domain since the beginning of 2008. The output is in CSV format.

```
adfind -csv -default -tdca -utc -binenc
-bit -f "(&(samaccounttype=805306368)
(|(lastLogonTimestamp<={{utc:
2008/01/01}}))(!(lastLogonTimestamp=*))
(!(userAccountControl:AND:=2)))" last
logontimestamp pwdlastset account
expires whencreated
```

The search excludes disabled users because most organizations tend to leave de-provisioned user objects in a disabled state for a period of time prior to deleting them. The search uses the lastLogonTimestamp attribute, a replicated attribute that gets updated periodically (and which is consequently not as accurate as the non-replicated lastLogon attribute), and lets you detect stale objects by querying a single domain controller (DC) rather than attempting to consolidate the lastLogon results from all DCs in the domain. The lastLogonTimestamp attribute is available with Windows Server 2003 and later.

Finding Inactive Computer Objects

Similar to the search for inactive user objects, the example below, written as one line, uses AdFind to search for computer objects that have either never logged on to the domain or have not logged on to the domain since the beginning of 2008:

```
adfind -csv -default -tdca -utc -binenc
-f "(&(objectcategory=computer)(|(last
LogonTimestamp<={{utc:2008/01/01}}))(!
(lastLogonTimestamp=*)))))" name operat
ingSystem operatingSystemServicePack la
stlogontimestamp pwdlastset whencreated
```

Finding Groups with Empty Membership

It's very difficult to determine whether a group is still in use within AD. At least with user and computer objects you can query for when the user or computer last set the password (using the pwdLastSet attribute) or when the user or computer account last logged on (using the lastLogonTimestamp attribute). But groups don't have passwords and don't log on to AD, so there are no useful attributes to help you determine whether a group is still in use.

A group with empty membership might be a good indication that it isn't in use, but realistically the only reliable method is to set a guardian. (You could then set up a process periodically requesting

Organize Your Active Directory Objects

Tony Murray

(Reprinted from WindowsItPro Magazine)

confirmation of the guardians that a group is still in use. If a confirmation isn't received within XX days, the de-provisioning process of a group could be initiated.)

In the example below I use AdFind to search for any groups that have no members, which is one indicator that the group might not be in use. Note that I exclude critical system objects (e.g., Enterprise Admins, built-in groups) as these can be legitimately empty and should never be removed:

```
adfind -csv -default -f "(&(object category=group)
(!member=*)(!isCritical SystemObject=TRUE))"
samaccount name description managedby
```

It's important that you question the validity of the results of your searches. An LDAP search against AD for the information (such as those in the examples shown above) is just one aspect of the overall task. You should qualitatively assess each result.

For example, it might be entirely valid for a user object corresponding to a resource mailbox (e.g., a meeting room) not to have logged on to the domain for 12 months or more. Another example is a group that has no members but is required to be present for a specific application to function.

Minimal Effort, Maximum Reward

Whatever terminology you use—manager, owner, contact, or guardian—the concept of linking AD objects to real people isn't new. In fact, Microsoft makes some provision in AD for defining the relationship through the managedBy and managedObjects linked-attribute pair for use with certain object types.

I strongly recommend that you consider implementing the concept of guardianship in your environment. The effort involved in setting up the required procedures is low and far outweighs the cost of dealing with an uncontrolled environment. The sooner you do this the less effort you will need to spend on clean-up tasks at a later date.