AdminSDHolder, Protected Groups and SDPROP
John Policelli

At a Glance:

- Overview of AdminSDHolder, protected groups and SDPROP
- Controlling groups that are protected by AdminSDHolder
- Security Descriptor propagator

Contents

Active Directory Domain Services uses AdminSDHolder, protected groups and Security Descriptor propagator (SD propagator or SDPROP for short) to secure privileged users and groups from unintentional modification. This functionality was introduced in the inaugural release of Active Directory in Windows 2000 Server and it's fairly well known. However, virtually all IT administrators have been negatively impacted by this functionality, and that will to continue unless they fully understand how AdminSDHolder, protected groups and SDPROP work.

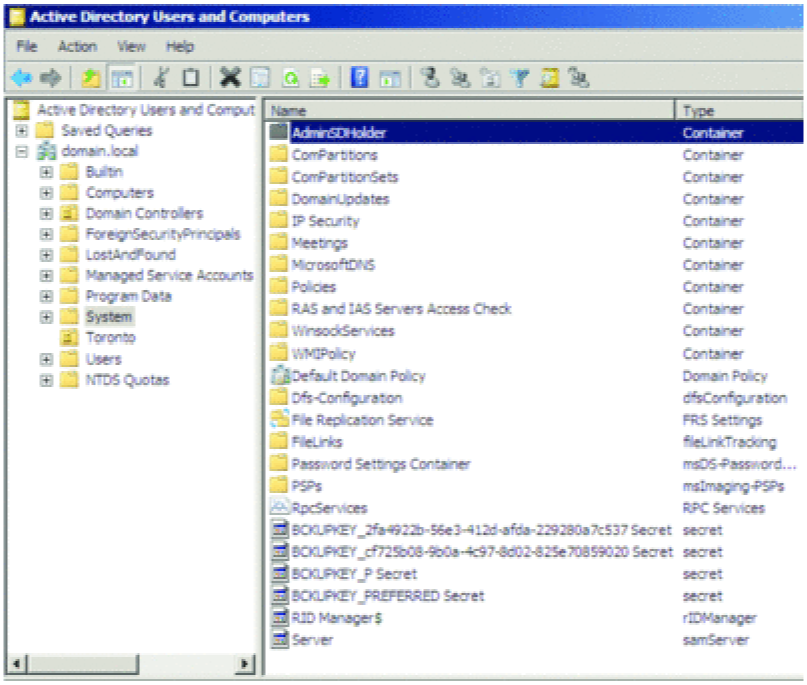
Each Active Directory domain has an object called AdminSDHolder, which resides in the System container of the domain. The AdminSDHolder object has a unique Access Control List (ACL), which is used to control the permissions of security principals that are members of built-in privileged Active Directory groups (what I like to call "protected" groups). Every hour, a background process runs on the domain controller that holds the PDC Emulator operations master role. It compares the ACL on all security principals (users, groups and computer accounts) that belong to protected groups against the ACL on the AdminSDHolder object. If the size or the binary string is different, the security descriptor on the object is overwritten by the security descriptor from the AdminSDHolder object..

As you can see, multiple layers of security are incorporated into this functionality. First, the permissions applied to users belonging to protected groups are more stringent than the default permissions applied onto other user accounts. Next, the default behaviour is that inheritance is disabled on these privileged accounts, ensuring that permissions applied at the parent level aren't inherited by the protected objects, regardless of where they reside. Finally, the background process running every 60 minutes identifies manual modifications to an ACL and overwrites them so that the ACL matches the ACL on the AdminSDHolder object.

See the sidebar "A Common Example of the Impact of AdminSDHolder, Protected Groups, and SDPROP" for a real-world look at this functionality.

The AdminSDHolder Object

As mentioned, each Active Directory domain contains an AdminSDHolder object that resides in the domain's System partition. The distinguished name of the AdminSDHolder object is "CN=AdminSDHolder,CN=System,DC=domain,DC=com," where DC=domain,DC=com is the distinguished name of the domain. **Figure 1** shows the AdminSDHolder object in a Windows Server 2008 R2 domain.



Name	Type
AdminSDHolder	Container
ComPartitions	Container
ComPartitionSets	Container
DomainUpdates	Container
IP Security	Container
Meetings	Container
MicrosoftDNS	Container
Policies	Container
RAS and IAS Servers Access Check	Container
WinsockServices	Container
WMIPolicy	Container
Default Domain Policy	Domain Policy
Dfs-Configuration	dfsConfiguration
File Replication Service	FRS Settings
FileLinks	fileLinkTracking
Password Settings Container	msDS-Password...
PSPs	msImaging-PSPs
RpcServices	RPC Services
BCKUPKEY_2fa4922b-56e3-412d-afda-229280a7c537 Secret	secret
BCKUPKEY_cf725b08-9b0a-4c97-8d02-825e70859020 Secret	secret
BCKUPKEY_P Secret	secret
BCKUPKEY_PREFERRED Secret	secret
RID Manager$	rIDManager
Server	samServer

Figure 1 **AdminSDHolder Object** (Click the image for a larger view)

Default ACL

Because the AdminSDHolder object is used in the process to secure privileged accounts, the default ACL on the AdminSDHolder object is more stringent than the ACL on other objects, such as the domain, OUs and containers.

In the default ACL for AdminSDHolder, the default Owner is the Domain Admins group, which is fairly unusual. Most Active Directory objects have the Administrators group as the default Owner. That's significant because an Owner can take control of an object and reset permissions.

Another important design factor for the AdminSDHolder object is that inheritance is disabled by default, which ensures that no parent-level permissions are inherited.

Finally, the Administrators, Domain Admins and Enterprise Admins groups are the groups that have the write permission to attributes on AdminSDHolder, which is more stringent than the default permissions applied to other Active Directory objects.

Protected Groups

As previously noted, AdminSDHolder permissions apply to security principals that belong to protected groups. The list of protected groups has expanded since the inaugural release of Active Directory in Windows 2000 Server. **Figure 2** shows the default protected groups and users from Windows 2000 Server to Windows Server 2008 R2.

Figure 2 Default Protected Groups

Windows 2000 Server RTM Windows 2000 Server with SP1 Windows 2000 Server with SP2 Windows 2000 Server with SP3	Windows 2000 Server with SP4 Windows Server 2003 RTM	Windows Server 2003 with SP1 Windows Server 2003 with SP2	Windows Server 2008 RTM Windows Server 2008 R2
Administrators	Account Operators	Account Operators	Account Operators
Domain Admins	Administrator	Administrator	Administrator
Enterprise Admins	Administrators	Administrators	Administrators
Schema Admins	Backup Operators	Backup Operators	Backup Operators
	Cert Publishers	Domain Admins	Domain Admins
	Domain Admins	Domain Controllers	Domain Controllers
	Domain Controllers	Enterprise Admins	Enterprise Admins
	Enterprise Admins	Krbtgt	Krbtgt
	Krbtgt	Print Operators	Print Operators
	Print Operators	Replicator	Read-only Domain Controllers
	Replicator	Schema Admins	Replicator
	Schema Admins	Server Operators	Schema Admins
	Server Operators		Server Operators

The list of protected groups consisted of four security groups in Windows 2000 Server RTM. In Windows 2000 Server SP4 and Windows Server 2003, several other groups were added, including the Administrator and Krbtgt accounts. In Windows Server 2003 with SP1 and later versions, Microsoft removed the Cert Publishers group from the default protected groups. In Windows Server 2008, Microsoft expanded this list to include the Read-Only Domain

Controllers group. The list of protected groups hasn't changed in the Release Candidate build of Windows Server 2008 R2.

A Common Example of the Impact of AdminSDHolder, Protected Groups and SDPROP

Most Active Directory administrators become aware of AdminSDHolder, protected groups and SDPROP through a scenario like this one:

You delegate permissions on an Organizational Unit (OU). You're later informed that the permissions are in place for most—but not all—user accounts in the OU. You determine that the ACL on the affected accounts is different from what you delegated and that inheritance is not enabled, so you enable inheritance to resolve the issue. Initially, this seems to work, but later the issue resurfaces. You again determine that the ACL is different and inheritance has been disabled.

I've seen individuals go through this seeming endless cycle over and over.

This situation actually occurs by design, however. It's caused by AdminSDHolder, protected groups and SDPROP.

The accounts affected by this issue belong to a protected group. As a result, the ACL on these accounts is inherited from the AdminSDHolder object in the domain, and inheritance is disabled by default. That's why the permissions that you delegated aren't applied to the affected user accounts. When you manually enable inheritance on these accounts, the delegated permissions are added to the ACL.

However, when the background process runs on the domain controller that holds the PDC Emulator operations master role—by default, every 60 minutes—the ACL is overwritten to match the ACL on the AdminSDHolder object and inheritance is disabled.

Controlling Groups that are Protected by AdminSDHolder

In my experience, a subset of these default protected groups causes problems with AdminSDHolder. For example, many organizations use the Print Operators group for Print Services management but not for Active Directory management. However, the Print Operators group is a protected group because it has elevated permissions on domain controllers by default. A best practice is to remove the elevated permissions that this group has on domain controllers. If you do follow this best practice (and you should!), you probably won't need to protect this group with AdminSDHolder.

You can exclude a subset of the default protect groups from the AdminSDHolder process, including:

- Account Operators
- Server Operators
- Print Operators
- Backup Operators

This ability to control groups that are protected by AdminSDHolder was introduced via hotfix for the RTM versions of Windows 2000 Server and Windows Server 2003 and is included in the most recent service pack for Windows Server 2003 and in the RTM versions of Windows Server 2008 and Windows Server 2008 R2. For more information on the hotfix, go to Delegated permissions are not available and inheritance is automatically disabled.

The ability to control groups protected by AdminSDHolder is enabled by modifying the dsHeuristic flag. This is a Unicode string in which each character contains a value for a single forest-wide setting. Character position 16 is interpreted as a hexadecimal value, where the left-most character is position 1. Therefore, the only valid values are "0" through "f". Each operator group has a specific bit, as shown in **Figure 3**.

Figure 3 dsHeuristic Operator Bits

Bit	Group to Exclude	Binary Value	Hexadecimal Value
0	Account Operators	0001	1
1	Server Operators	0010	2
2	Print Operators	0100	4
3	Backup Operators	1000	8

This situation becomes even more complicated when you're trying to exclude more than one group from AdminSDHolder, especially because you can have multiple combinations of exclusions—for example, Account Operators and Server Operators, or Account Operators and Backup Operators. To deal with this issue, simply add the binary value of each group and then convert the result to a hexadecimal value. For example, to exclude the Print Operators and Backup Operators groups, take the binary value for the Print Operators group (0100) and add it to the binary value of the Backup Operators group (1000), which equals 1100. You then convert the binary sum (1100) to the hexadecimal value (C).

To make this task a little easier, **Figure 4** lists all possible combinations in binary and hexadecimal format.

Figure 4 dsHeuristic Values for Excluding Combinations of Groups

Group(s) to Exclude	Binary Value	Hexadecimal Value
None (Default)	0	0
Account Operators	1	1
Server Operators	10	2
Account Operators Server Operators	0001 + 0010 = 0011	3
Print Operators	100	4
Account Operators Print Operators	0001 + 0100 = 0101	5
Server Operators Print Operators	0010 + 0100 = 0110	6
Account Operators Server Operators Print Operators	0001 + 0010 + 0100 = 0111	7
Backup Operators	1000	8
Account Operators Backup Operators	0001 + 1000 = 1001	9
Server Operators Backup Operators	0010 + 1000 = 1010	A
Account Operators Server Operators	0001 + 0010 + 1000 = 1011	B

Backup Operators		
Print Operators Backup Operators	0100 + 1000 = 1100	C
Account Operators Print Operators Backup Operators	0001 + 0100 + 1000 = 1101	D
Server Operators Print Operators Backup Operators	0010 + 0100 + 1000 = 1110	E
Account Operators Server Operators Print Operators Backup Operators	0001 + 0010 + 0100 + 1000 = 1111	F

After deciding which group(s) you want to exclude, you're ready to modify the dsHeuristics attribute. For details on that process, see the sidebar "How to Use the dsHeuristics Attribute to Exclude Groups from AdminSDHolder."

Modifying How Often the AdminSDHolder Background Process Runs

If the default frequency of 60 minutes for the background AdminSDHolder process to run isn't sufficient, you can change it by creating or modifying the AdminSDProtectFrequency entry in the HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\NTDS\Parameters subkey.

If this key doesn't exist, the default frequency (60 minutes) is used.

You can configure the frequency to anywhere between one minute and two hours. You must enter the number of seconds when creating or modifying the registry entry. The following command will configure SDPROP to run every 10 minutes (600 seconds):

```
REG ADD HKLM\SYSTEM\CurrentControlSet\Services\NTDS\Parameters /V
AdminSDProtectFrequency /T REG_DWORD /F /D 600
```

Note, however, that modifying this subkey isn't recommended because doing so can increase LSA (Local Security Authority) processing overhead.

Determining Whether a Security Principal Is Protected by AdminSDHolder

A fairly large number of default users and groups are protected by AdminSDHolder. One thing to keep in mind is that users are protected by AdminSDHolder if they have direct or transitive membership in a security or distribution group. Distribution groups are included because a distribution group can be converted to a security group.

Let's say a user belongs to a distribution list called Canada IT. The Canada IT DL is a member of the North American IT security group; the North American IT security group is a member of the Administrators group. Because the user's transitive group membership includes the Administrators group (by virtue of group nesting), the user's account is protected by AdminSDHolder.

There's an easy way to determine which users and groups AdminSDHolder protects in your domain. You can query the adminCount attribute to determine whether an object is protected by the AdminSDHolder object. The following examples use the ADFind.exe tool, which can be downloaded from joeware.net.

- To find all user objects in a domain that are protected by AdminSDHolder, type:

```
Adfind.exe -b DC=domain,DC=com -f
"&(objectcategory=person)(samaccountname=*)(admincount=1)" -dn
```

- To find all groups in a domain that are protected by AdminSDHolder, type:

```
Adfind.exe -b DC=domain,DC=com -f "&(objectcategory=group)(admincount=1)" -dn
```

Note: In the preceding examples, replace DC=domain,DC=com with the distinguished name of your domain.

Orphaned AdminSDHolder Objects

When a user is removed from a protected group, the adminCount attribute on that user account does not change; the value 1 remains. Furthermore, the status of inheritance is not changed. As a result, the user account no longer receives its ACL from the AdminSDHolder object, but it also doesn't inherit any permissions from parent objects, provided inheritance has not been enabled on the AdminSDHolder object. The common term for this issue is "orphaned AdminSDHolder objects." There is no automated mechanism to fix inheritance on objects that no longer belong to protected groups; you must deal with orphaned AdminSDHolder objects manually. Microsoft has developed and made available a VB Script that will assist you in re-enabling inheritance on user accounts that were previously members of protected groups. To find the VB Script, go to Delegated permissions are not available and inheritance is automatically disabled.

Security Descriptor Propagator

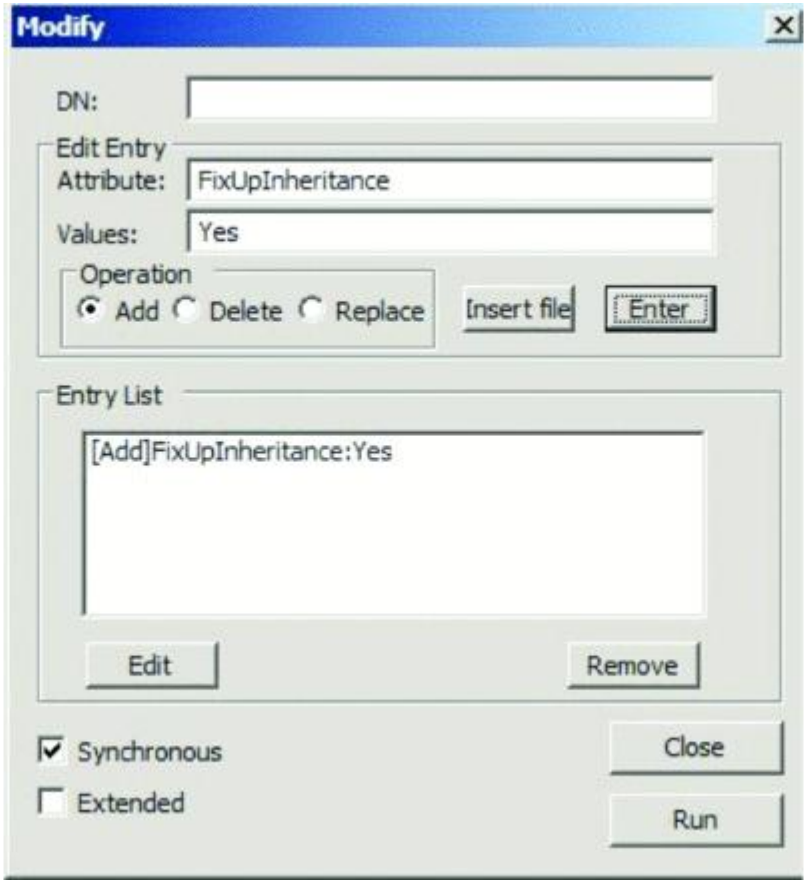
In order to propagate the changes of inheritable ACEs to descendent objects, domain controllers runs a background task called the Security Descriptor Propagator Update task. This task is triggered by a modification to the security descriptor for an object or when an object is moved.

Forcing SDPROP to Run

You can also force SDPROP to run in cases where you're testing changes or you can't wait for the configured interval. Forcing SDPROP to run involves manually initializing the SDPROP thread to evaluate inherited permissions for objects in Active Directory. This process can be achieved by taking the following:

1. Go to Start. Click Run. Type LDP.exe. Click OK.
2. On the Connection menu in the LDP console, click Connect.
3. In the Connect dialog box, type the server name you want to connect to in the Server field and ensure that 389 is listed in the Port field. Click OK.

4. On the Connection menu, click Bind.
5. In the Bind window, select the Bind as the currently logged-on user option, or select the Bind with Credentials option. If you selected the latter, enter the credentials that you want to bind with. Click OK.
6. On the Browse menu, select Modify.
7. In the Modify dialog box, leave the DN field empty. Type FixUpInheritance into the Attribute field. Type Yes into the Value field. Select the Add operation, then click Enter. **Figure 5** shows how the Modify window should look.
8. In the Modify dialog box, click Run. The Details pane will be similar to the highlighted text in **Figure 6**.



Modify

DN:

Edit Entry

Attribute: FixUpInheritance

Values: Yes

Operation

● Add ○ Delete ○ Replace

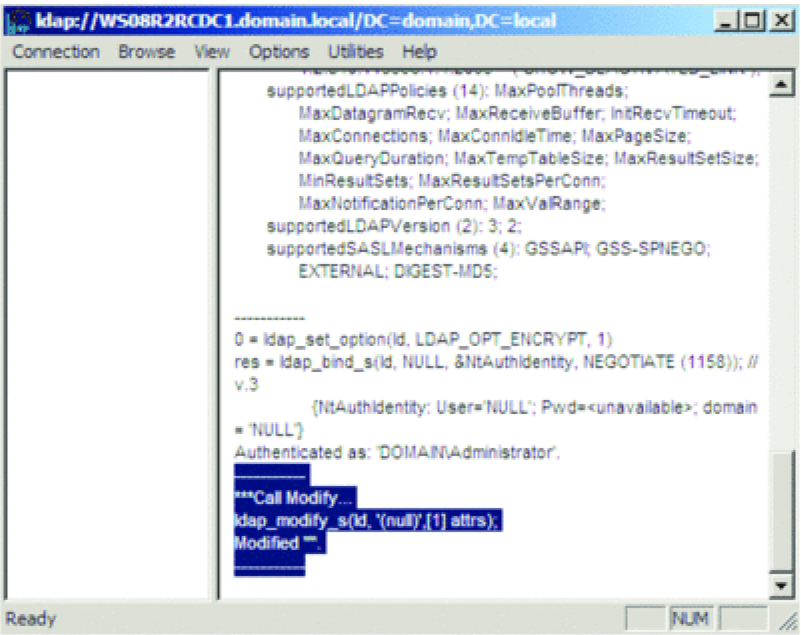
Entry List

[Add]FixUpInheritance:Yes

☑ Synchronous

□ Extended

Figure 5 **The Modify Window. When Forcing SDPROP to run in the Modify dialog box, click Run. The Details pane will be similar to the highlighted text in Figure 6.** (Click the image for a larger view)



```
    supportedLDAPPolicies (14): MaxPoolThreads;
        MaxDatagramRecv; MaxReceiveBuffer; InitRecvTimeout;
        MaxConnections; MaxConnIdleTime; MaxPageSize;
        MaxQueryDuration; MaxTempTableSize; MaxResultSetSize;
        MinResultSets; MaxResultSetsPerConn;
        MaxNotificationPerConn; MaxValRange;
    supportedLDAPVersion (2): 3; 2;
    supportedSASLMechanisms (4): GSSAPI; GSS-SPNEGO;
        EXTERNAL; DIGEST-MD5;

-----------
0 = ldap_set_option(ld, LDAP_OPT_ENCRYPT, 1)
res = ldap_bind_s(ld, NULL, &NtAuthIdentity, NEGOTIATE (1158)); //
v.3
          {NtAuthIdentity: User='NULL'; Pwd=<unavailable>; domain
= 'NULL'}
Authenticated as: 'DOMAIN\Administrator'.
-----------
***Call Modify...
ldap_modify_s(ld, '(null)',[1] attrs);
Modified "".
-----------
```

Figure 6 **Call Modify Operation in LDP.exe.** (Click the image for a larger view)

At this point, SDPROP should initialize. The amount of time the SDPROP process takes depends on the size of your Active Directory environment. The larger the environment, the longer it will take the process to run. You can monitor the DS Security Propagation Events counter in the NTDS Performance object to determine when SDPROP has completed, which will be indicated by a counter value of 0.

Wrapping Up

The AdminSDHolder is an important security feature in Active Directory. The AdminSDHolder, protected groups and Security Descriptor propagator help secure user accounts that contain elevated Active Directory permissions. The AdminSDHolder functionality has evolved from Windows 2000 Server to Windows Server 2008. During this evolution, Microsoft has expanded the number of objects that are secured by AdminSDHolder, introduced the ability to exclude certain groups from the AdminSDHolder and added the ability to control how often AdminSDHolder runs.

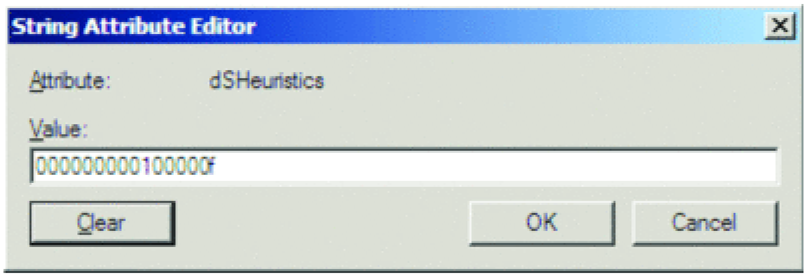
Most Active Directory administrators have been introduced to AdminSDHolder, intentionally or unintentionally. I've tried to provide you with a good understanding of what AdminSDHolder is, how it works and what cleanup is required when you remove a user from a protected group, along with some useful tweaks. I hope that this information will help prevent you from being caught off guard by the AdminSDHolder functionality the next time you delegate or remove Active Directory permissions.

How to Use the dsHeuristics Attribute to Exclude Groups from AdminSDHolder

The dsHeuristics attribute can be used to exclude certain groups from being protected by AdminSDHolder. The following instructions outline the steps for modifying the dsHeuristics attribute on Windows Server 2008 R2:

1. Log on to a domain controller or a member computer that has the Remote Server Administrator Tools (RSAT) installed.
2. Go to Start. Click Run. Type adsiedit.msc, then click OK.
3. In the ADSI Edit console, right-click on ADSI Edit in the console tree. Select Connect To.
4. In the Connection Settings window, select Configuration from the Select a Well-Known Naming Context drop-down. Click OK.
5. In the console tree, expand Configuration, expand Service, and expand Windows NT. Right-click on the Directory Service node, then select Properties.
6. In the CN=Directory Service Properties window, select dsHeuristics. Click Edit.
7. In the String Attribute Editor window, copy the existing value for dsHeuristics if it is set.
8. In the String Attribute Editor window, replace the dsHeuristics value with what you want to set, such as 000000000100000f to exclude Account Operators, Server Operators, Print Operators, and Backup Operators groups. **Figure A** shows the String Attribute Editor window.

   Note: Replace the zeros in the first part of the value with what you may already have in dsHeuristics. Make sure that you have the correct count of digits up to the "f" or whatever bits you want to set.
9. Click OK on the String Attribute Editor window. Click OK on the CN=Directory Service Properties window.



Attribute: dSHeuristics

Value:
000000000100000f

Figure A **String Attribute Editor Window** (Click the image for a larger view)

John Policelli, Microsoft MVP for Directory Services, MCTS, MCSA, ITSM, iNet+, Network+ and A+, is a solutions-focused IT consultant with more than a decade of combined success in architecture, security, strategic planning and disaster-recovery planning. For the past nine years, he has focused on identity and access management and providing thought leadership for some of Canada's largest installations of Active Directory. Policelli is the author of *Active Directory Domain Services 2008 How-To* (Sams Publishing, 2009).