

Realtime
publishers

"Leading the Conversation"

The Essentials Series

Active Directory 2008 Operations

sponsored by

SCRIPTLOGIC

by Greg Shields

Article 1: Understanding Active Directory Auditing in Windows Server 2008.....	1
Enabling Auditing in Windows Server 2008	1
Windows Server 2008’s New Auditing Subcategories.....	3
Configuring Audit Subcategories	5
Auditing for AD Changes	6
Targeted Auditing Data Is Better Auditing Data	7
Article 2: Understanding Active Directory Recovery in Windows Server 2008.....	8
Backing Up AD.....	8
Full Server Recovery of a Domain Controller	10
Restoring Deleted AD Objects.....	12
Locating Deleted Objects with DSAMAIN	13
Recovery Knowledge Is as Important as Backup Knowledge.....	14
Article 3: Understanding the Security Implications of Server 2008 RODCs	15
History Repeats Itself.....	15
Introducing the RODC	17
Creating an RODC	19
RODCs Protect the Domain.....	21

Copyright Statement

© 2008 Realtimepublishers.com, Inc. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtimepublishers.com, Inc. (the "Materials") and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtimepublishers.com, Inc or its web site sponsors. In no event shall Realtimepublishers.com, Inc. or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtimepublishers.com and the Realtimepublishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtimepublishers.com, please contact us via e-mail at info@realtimepublishers.com.

Article 1: Understanding Active Directory Auditing in Windows Server 2008

Throughout the history of the Windows operating system (OS), the features available to enable and monitor auditing for Active Directory (AD) have been relatively limited. Nine general categories of auditing have traditionally been available, all of which result in a fairly coarse level of logging to the Windows Event Log. By including only a small number of log categories, the result of enabling logging is a vast amount of excess data that must be managed in order to capture auditable actions of interest. At the same time, auditing requirements brought about by industry and governmental compliance regulations have increased the criticality for effective and consistent logging in many network environments.

With Microsoft's release of Windows Server 2008, audit logging gains new levels of granularity associated with configurable event categories and subcategories, while a new Windows Event Log improves the process of filtering for and locating events of interest. AD itself gains four new logging subcategories that assist with the monitoring of configuration changes and replication in addition to object accesses.

This white paper will discuss the new audit capabilities specific to AD gained through an upgrade to Windows Server 2008. It will provide specific guidance and step-by-step instructions to assist you, the administrator, with making best use of AD's new auditing features.

Enabling Auditing in Windows Server 2008

The process to enable auditing in Windows Server 2008 arrives relatively unchanged from its implementation in previous OS versions. Enabling the basic auditing of AD events on domain controllers is most often performed using Group Policy through modification of the native Default Domain Controllers Policy. Enabling auditing in this manner ensures that auditing settings are configured consistently across all domain controllers. Figure 1 shows a configured policy as seen within the Group Policy Management Editor.

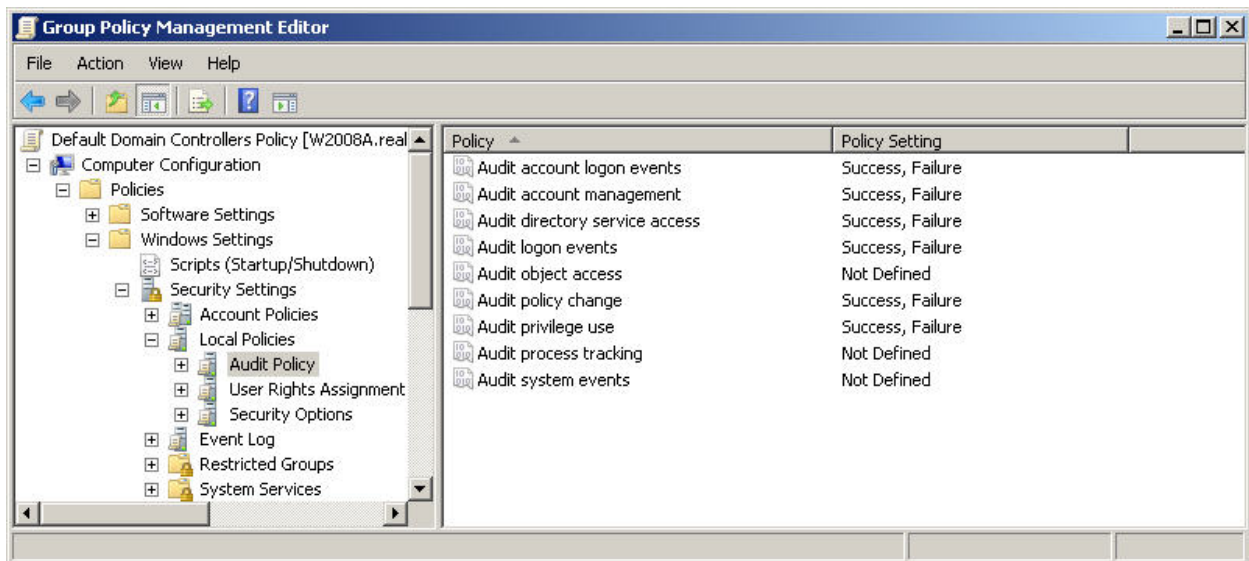


Figure 1: An example of AD auditing as enabled through the Default Domain Controllers Policy.

As you can see, a number of auditing categories are exposed through Group Policy. Each category has the ability to enable auditing for both Success and Failure events. Categories and event types are enabled based on the types of events you are interested in logging to the Windows Security Event Log. Though these categories are not new to Windows Server 2008, there remains some confusion about their use. Let's take a look at each:

- Audit account logon events—This category generates an event when a user attempts to login or log out of a computer using a domain account.
- Audit account management—This category audits the creation, change, renaming, or deletion of user accounts or groups. It also audits the setting or change of a password.
- Audit directory service access—This category audits the attempt by users to access AD objects. Individual AD objects to be monitored must have their System Access Control List (SACL) configured to be monitored. The process of enabling this will be discussed shortly.
- Audit logon events—This category generates an event when a user attempts to login or log out of a computer using a local computer account.
- Audit object access—This category audits the attempt by a user to access an object, such as files, folders, registry keys, or printers, among others. Individual AD objects to be monitored must have their SACL configured to be monitored.
- Audit policy change—This category generates an event when a user attempts to change a user rights assignment policy, audit policy, or trust policy.
- Audit privilege use—This category audits the attempt by users to exercise the use of their assigned user rights.
- Audit process tracking—This category audits highly detailed tracking information about program activation, process exit, handle duplication, and indirect object access. This level of auditing is often employed by developers and during deep troubleshooting.
- Audit system events—This category generates an event when a user restarts or shuts down a computer or attempts to modify system security or the security log.

Although most of these auditing categories globally enable their type of auditing, two in particular require the configuration of object SACLs to enable auditing. Those two are *Audit directory service access* and *Audit object access*. For these two categories, the individual objects to be audited must also be configured if they are to be audited. For objects such as files, right-clicking the file and selecting Properties brings forward the properties dialog box. Selecting the Security tab, then Advanced, followed by the Auditing tab presents a dialog box used to configure which users and types of accesses should be audited.

The same process is used to audit the configuration of AD objects within Active Directory Users and Computers. To configure the SACL for an AD object, launch Active Directory Users and Computers, then click View | Advanced Features. Doing so enables the Security tab to be seen for individual objects. Right-click an object of interest, and select the Security tab, then click Advanced, and then select the Auditing tab. Figure 2 shows an example of configuring auditing for the Everyone group on the Computers organizational unit (OU).

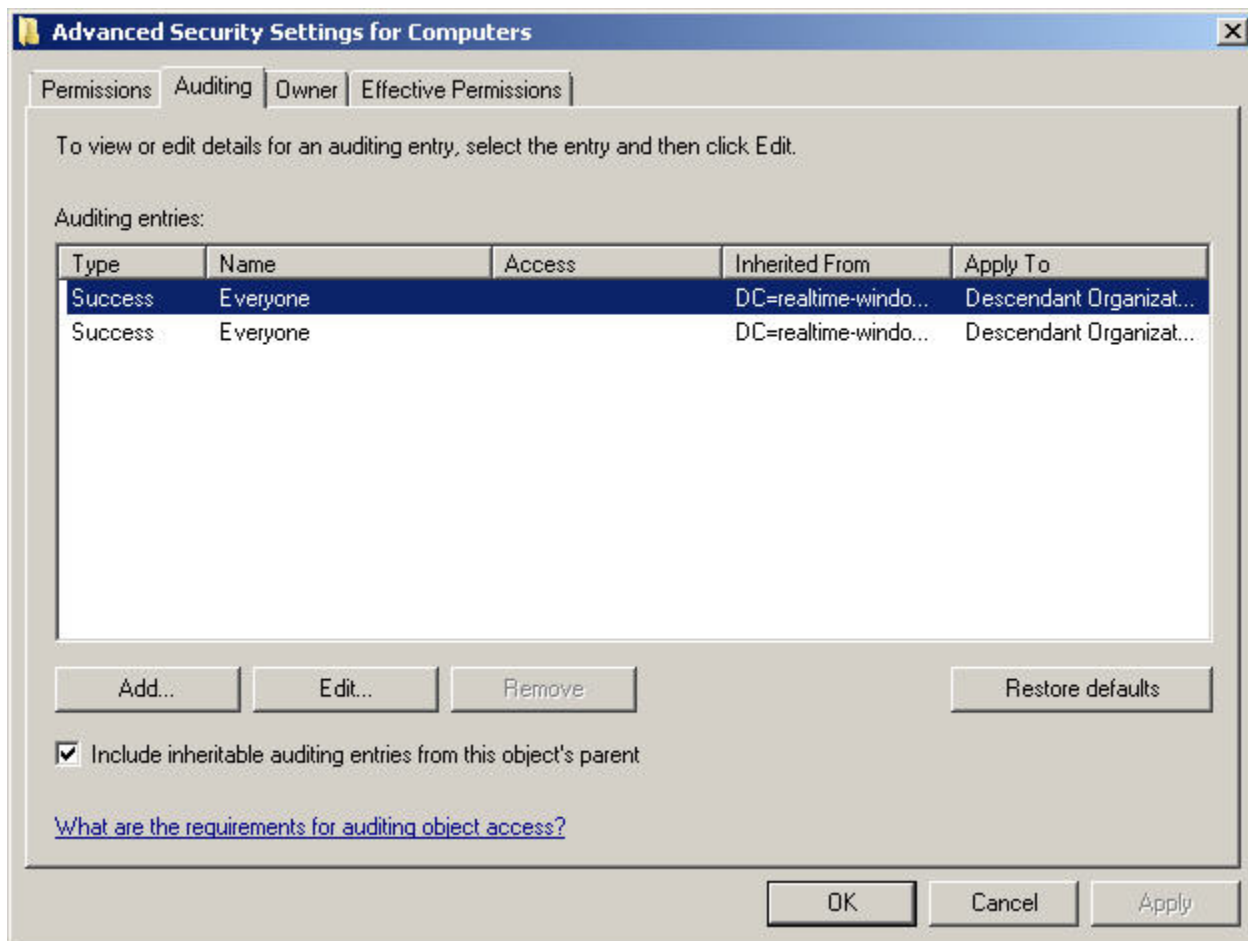


Figure 2: Configuring object-level auditing within Active Directory Users and Computers.

Windows Server 2008's New Auditing Subcategories

The problem with these nine categories in previous versions of the Windows OS was that they didn't provide the level of granularity needed by many administrators. Enabling the *Audit account management* category effectively turned on auditing for all types of account management activities. If you were interested in only auditing for user account management and had no interest in computer account management, you were stuck with wading through the extra data associated with its Event Log entries.

With Windows Server 2008, the original nine categories are broken into 50 audit policy subcategories. These subcategories allow for precise control over the types of events logged into the Security Event Log. Table 1 highlights each of these new subcategories and their relation to the original nine audit policies. As you'll learn, knowing the name of each subcategory and its relation to its category is important for the command-line tool used to enable them.

Audit Category Name	Associated Audit Subcategories
System events	<ul style="list-style-type: none"> • Security state change • IPsec driver • Security system extension • System integrity • Other system events
Login events	<ul style="list-style-type: none"> • Logon • Logoff • Account Lockout • IPsec main mode • IPsec quick mode • IPsec extended mode • Special login • Other logon/logoff events
Object access	<ul style="list-style-type: none"> • File system • Registry • Kernel object • SAM • Certification services • Application generated • Handle manipulation • File share • Filtering platform packet drop • Filtering platform connection • Other object access events
Privilege use	<ul style="list-style-type: none"> • Sensitive privilege use • Non-sensitive privilege use
Process tracking	<ul style="list-style-type: none"> • Process creation • Process termination • DPAPI activity • RPC events
Policy change	<ul style="list-style-type: none"> • Audit policy change • Authentication policy change • Authorization policy change • MPSSVC rule-level policy change • Filtering platform policy change • Other policy change events
Account management	<ul style="list-style-type: none"> • User account management • Computer account management • Security group management • Distribution group management • Application group management • Other account management event

Directory service access	<ul style="list-style-type: none"> • Directory service access • Directory service changes • Directory service replication • Detailed directory service replication
Account logon events	<ul style="list-style-type: none"> • Kerberos service ticket operations • Credential validation • Kerberos authentication service • Other account logon events

Table 1: A list of the new audit subcategories and their relation to the original nine audit categories.

 You can find detailed information about the Event IDs and descriptions associated with each of the new subcategories at <http://support.microsoft.com/default.aspx/kb/947226/en-us>.

Configuring Audit Subcategories

Unlike with the nine categories, the implementation of these new subcategories is not done through Group Policy. Nor are they enabled through Local Security Policy. Rather, the only mechanism currently available for enabling specific subcategories is through the command-line tool `auditpol.exe`.

The `auditpol` command is used to enable and disable individual subcategories on individual machines. As `auditpol` does not leverage Group Policy for its assignment, it must be configured individually on each machine. Auditpol is equipped with a number of switches that are used to set and verify policy assignment. For example, to configure success auditing for the account management category with all subcategories, use the command

```
Auditpol /set /category:"account management"
```

It is also possible to use `auditpol` to set specific subcategories, one per command. Do so by using the `/subcategory` switch. To enable both success and failure auditing on only the *Computer account management* and *User account management* subcategories of the account management category, use the following two commands:

```
Auditpol /set /subcategory:"user account management"
/success:enable /failure:enable
```

```
Auditpol /set /subcategory:"computer account management"
/success:enable /failure:enable
```

The process of verifying set policies is also done through the same command-line tool. Listing 1 highlights an example of using the /get switch to verify the configuration after running the two previous commands.

```
C:\Users\Administrator>auditpol /get /category:"account management"

System audit policy
Category/Subcategory      Setting
Account Management
  Computer Account Management      Success and Failure
  Security Group Management        No Auditing
  Distribution Group Management    No Auditing
  Application Group Management     No Auditing
  Other Account Management Events  No Auditing
  User Account Management          Success and Failure

C:\Users\Administrator>
```

Listing 1: The result of using the /get switch to verify correct audit subcategory configuration.

Auditing for AD Changes

As you can see in Table 1, AD auditing gains four new subcategories that provide further granularization of audit data. One subcategory that is of particular use in AD environments is Directory Service Changes. This new audit subcategory enables a new type of auditing associated with the configuration of individual AD objects. In environments that must support regulatory compliance, this new audit subcategory enables critical information about the configuration of AD itself. Once enabled, as an administrator attempts to make a change to an AD configuration, that change is logged to the Security Event Log. What makes this new subcategory particularly powerful is that information about what was changed along with the old and new values are now stored in the log entry itself.

For each change, two events are logged. The first event shows the attribute's "old" value related to the configuration change. The second event shows the attribute's "new" value. Because of this split, determining what happened requires a bit of sleuthing to match the two entries together. Four possible Event IDs can be logged:

- 5136 - Modify—An attribute for an existing object has been modified
- 5137 - Create—A new object has been created
- 5138 - Undelete—An object has been undeleted
- 5139 - Move—An object has been moved

It is possible to control which objects are audited through the same SACL modification process as was used earlier. Also possible with this subcategory alone is the disabling of certain attributes directly within the AD schema. You can do so by launching ADSI Edit and navigating to the Schema naming context. In the resulting tree is a list of each attribute in your AD. Double-click an attribute you do not want to be audited. On the Attribute Editor tab of the resulting window, change the value of the searchFlags attribute to 256. As Figure 3 shows, this sets the value for the attribute to NEVER_AUDIT_VALUE.

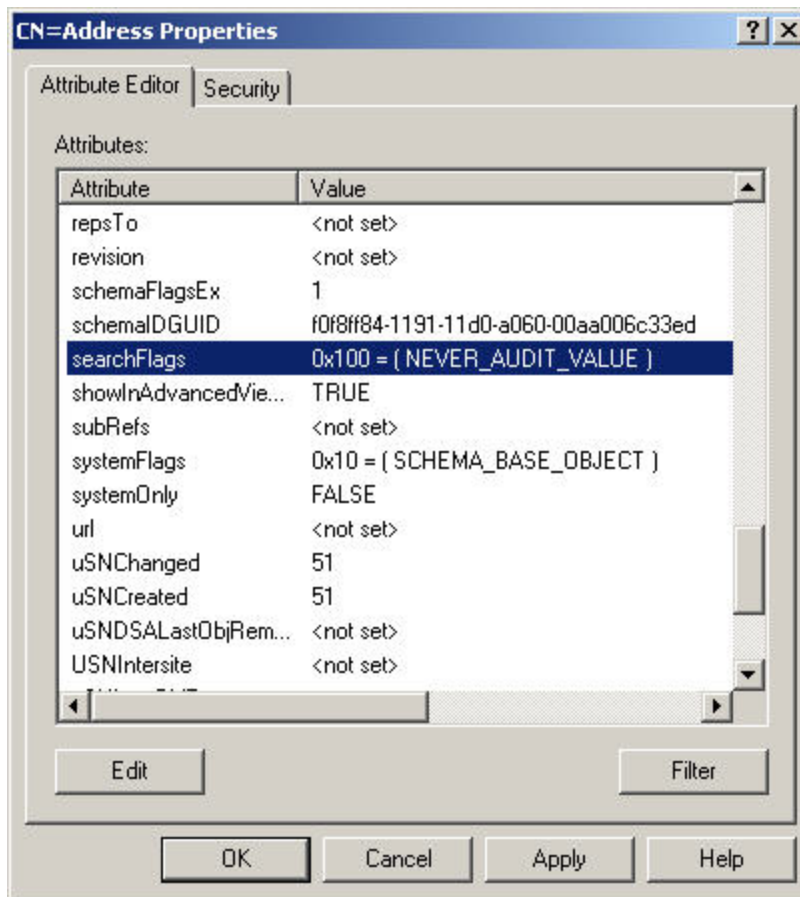


Figure 3: Configuring an AD attribute's changes to not be audited.

Targeted Auditing Data Is Better Auditing Data

Auditing has historically been a challenging process in the Windows OS, in many ways due to its previous limitations on auditing categories. As you can see, those limitations have been lifted somewhat through the incorporation of greater granularity and higher-quality information. The ability to watch for and report on AD configuration changes enables IT to better fulfill the needs of regulatory compliance in monitoring the actions of users and administrators on the network.

Article 2: Understanding Active Directory Recovery in Windows Server 2008

The deceptive ease of backing up Windows' Active Directory (AD) has often lulled Windows administrators into believing that its restoration is similarly easy. Although native tools such as NTBackup and the new Windows Server Backup available in Windows Server 2008 make the backup process relatively easy, the process to restore individual objects remains somewhat complex.

Thankfully, while individual object restores remain a multi-step process, Windows Server 2008 includes new tools that ease the task of completing a bare-metal recovery of a failed domain controller. Using Windows Server 2008's combination of Windows Server Backup and Windows Complete PC Restore, the process to resurrect a failed domain controller has now become fairly trivial.

This white paper will discuss the processes needed to properly backup and restore individual Active Directory objects as well as entire DC's once they have been upgraded to Windows Server 2008. It will provide specific guidance and step-by-step instructions to assist you the administrator with understanding and best completing this critical recovery task.

Backing Up AD

In order to back up AD, the Windows Server Backup feature must first be installed to the domain controller of interest. In contrast to the NTBackup tool used in previous versions, the Windows Server Backup feature is not installed by default. To install the feature and prepare it for first use, launch Server Manager, right-click the Features node, and select Add Features. Select the check box next to Windows Server Backup Features, and click Next, Install to install the feature.



Another available feature selection is Command-line Tools. These tools are useful for scripting or creating batch files to initiate backup or restore operations and are necessary to complete the System State restore necessary to restore individual AD objects.

In Windows Server 2003 as well as earlier versions, backing up AD was done via a backup of the System State. Backing up the System State of a domain controller captured the proper components to ensure AD could be restored successfully onto the same computer. The problem with System State backups was that they did not capture the entire composition of the domain controller. Instead, only a small portion of the server—such as boot files, registry, and COM+ class registrations—were captured to the backup in addition to AD's NTDS database and the SYSVOL. Because of this shortcoming, restoring a failed domain controller meant rebuilding a new server instance, upgrading that instance to the same service pack and patch level, and restoring the System State over the top of the core installation. This mechanism for completing a restoration required less storage space for backups but a longer amount of time required to complete a restore as well as a reduced chance of a successful restore.

With Windows Server 2008, System State backups are deprecated in favor of what are called critical volumes. Critical volumes are those volumes that are required to recover the AD. They include the operating system (OS) files, the registry, the NTDS database and log files, and the SYSVOL. The critical volumes required to be backed up can be as few as a single volume in the case where all AD components are installed to the same drive, or they can be multiple volumes if AD components were separated at installation to different drives.

To back up AD, launch Server Manager and navigate to the Storage | Windows Server Backup node. Select Backup Once to begin a single backup instance. You will need to answer questions in the following screens of the Backup Once Wizard:

- *Backup options*—If options have been previously selected for the Backup Schedule Wizard, those options will be selected here. Otherwise, choose *Different options* to select a new set of options.
- *Select backup configuration*—The option to back up the *Full server* is provided as well as a Custom option to select volumes of interest. When selecting volumes to back up, ensure that all critical volumes are selected.
- *Select backup items*—If the Custom option was selected in the previous screen, the screen shown in Figure 1 enables the selection of volumes to back up. A check box is also available to *Enable system recovery*. This check box must be selected in order to perform a bare-metal restore.
- *Specify destination type*—Backed up data can be stored either to an available local drive or to a remote shared folder. Backups cannot be stored to critical volumes, but it is possible to back up full volumes to DVD media. As a workaround, it is possible to store backups to a shared folder on the local machine by referring to the full shared folder path on the local machine.
- *Specify remote folder* or *Specify backup destination*—Depending on the selection in the previous screen, one of these two options will be displayed. Select either an available local drive or a remote folder. If you select a remote folder, you are given the option to select privilege inheritance on the target folder.
- *Specify advanced configuration*—In this screen, VSS backup behavior is determined, which configures the backup to retain or clear application log files as well as update file backup history. This selection is important when other backup products are being used to back up applications.
- *Confirmation*—Click Backup to start the backup.

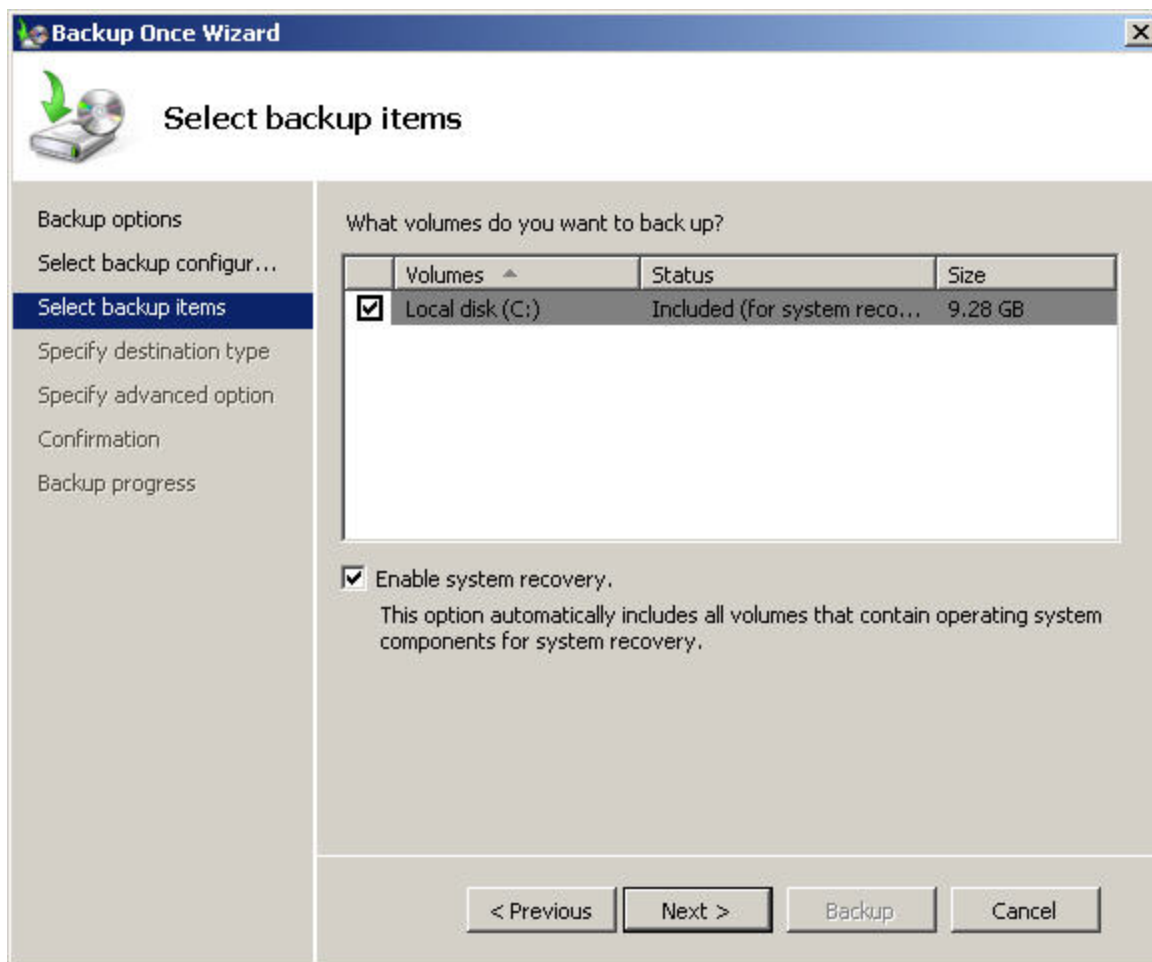


Figure 1: Configuring the volumes to back up in the Backup Once Wizard.

Full Server Recovery of a Domain Controller

In Windows 2003 and earlier, the native Windows restoration process required the installation of an OS prior to starting a restore. This added time and complexity to the restoration process. With Windows Server 2008, once a domain controller backup has been completed, it is then possible to restore that domain controller directly onto bare metal. The target computer need not be the exact same computer as the source of the backup, but it must have the exact same hardware composition.

This capability for bare-metal restoration speeds the restoration process while providing a greater assurance of a successful restore. It relies on the use of Windows Server 2008's pre-installation environment. WinPE natively includes many of the necessary networking and storage drivers as well as a graphical OS to assist with the restoration process.

To complete a bare-metal domain controller restoration, boot the target server with the Windows Server 2008 media DVD, and click Next when prompted. In the resulting screen, click the link titled *Repair your computer*, and then Next at the following screen. When the System Recovery Options screen appears, select Windows Complete PC Restore. The system will attempt to scan the local machine's drives for a current backup. If backup files are stored elsewhere on the network, click Cancel. In the resulting *Restore your entire computer from backup* wizard, select *Restore a different backup*, and click Next.

WinPE includes a set of common network drives that function with many network cards. Because of this native support, it is possible to connect WinPE to Windows shares elsewhere on the network that contain backup files. In the screen titled *Select the location of the backup*, click Advanced, and select *Search for a backup on the network*. Choose Yes when asked to verify that the network is a trusted network. In the resulting box, enter the full path to the network share containing the backup of interest. Figure 2 shows an example of the dialog box that appears when the wizard successfully connects to remote backup files.

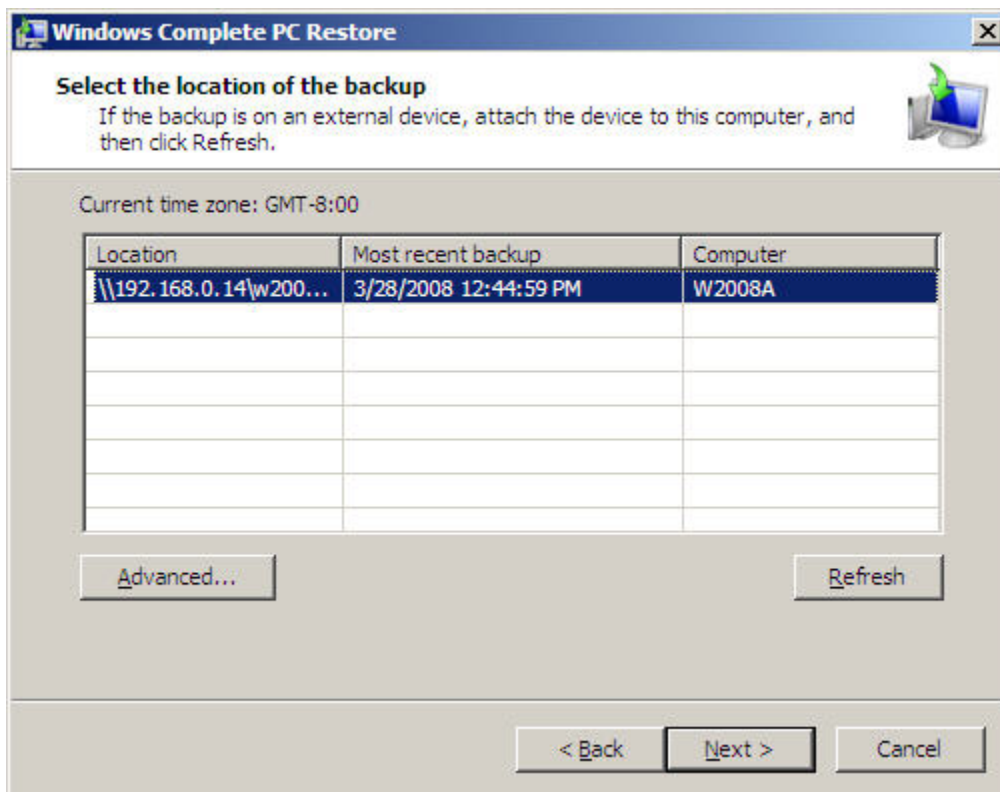



Figure 2: Windows Complete PC Restore successfully connecting to a remote share for backup files.

Select the location of the backup, and click Next. In the screen that follows, select the backup of interest in that location, and click Next again. The resulting screen provides the ability to exclude disks from the restore, install any necessary storage drivers, restart the computer after the restore, and complete automatic checking and updating of disk error information. Click Next, then Finish to start the restoration.

 Be aware that the target disk must be at least as big as the source disk. This is a requirement even if the backup file size is smaller than the total size of the disk. If the target disk is larger than the source disk, a volume will be created on the target disk during restoration that is equal to the original size of the source disk. Using Disk Management, it is possible to later extend the volume to consume the extra space if desired.

Restoring Deleted AD Objects

In the introduction to this paper, it was mentioned that the process to restore deleted AD objects remains complex even upon the upgrade to Windows Server 2008. Such remains the case because much of the process of completing an object restoration in Windows Server 2008 is effectively the same as in previous OS versions.

To begin the process, you will need to reboot the server into Directory Services Restore Mode (DSRM) and complete a non-authoritative restore of AD. To boot into DSRM, hit F8 during the initial boot cycle. In the resulting screen, select Directory Services Restore Mode, and hit the Enter key. After the machine boots into DSRM, login with `.\administrator` as the username along with the DSRM password.

Although System State backups are deprecated in Windows Server 2008, System State restores are still used for restoring objects in AD. System State restores are supported only through the Windows Server Backup command-line tool `wbadmin`. Two steps are necessary. The first step identifies the correct backup from which to restore data. The second step begins the non-authoritative restore. The first step is shown below:

```
wbadmin get versions -backuptarget:<targetDrive>:  
-machine:<backupComputerName>
```

In this step, `<targetDrive>` identifies the location where the backup media is currently stored, while `<backupComputerName>` identifies the name of the computer where you want to recover the backup. Listing 1 shows an example of a result from running this command.

```
C:\Users\Administrator>wbadmin get versions -backuptarget:\\SRV1\share  
-machine:w2008a  
  
wbadmin 1.0 - Backup command-line tool  
(C) Copyright 2004 Microsoft Corp.  
  
Backup time: 3/28/2008 2:44 PM  
Backup target: Network Share labeled \\SRV1\share  
Version identifier: 03/28/2008-20:44  
Can Recover: Volume(s), File(s), Application(s), Bare Metal Recovery,  
System State  
  
C:\Users\Administrator>
```

Listing 1: The result from running the `get versions` switch of the `wbadmin` command.

The version identifier previously shown is needed for input into the second command. In Listing 1, the version identifier is shown as *03/28/2008-20:44*. This string is used to replace *<version>* in the command below:

```
wbadmin start systemstaterecovery -version:<MM/DD/YYYY-HH:MM>
-backuptarget:<targetDrive>: -machine:<backupComputerName>
```

The values of *<targetDrive>* and *<backupComputerName>* remain the same as for the previous command. After running this command, hit the Y key to start the System State recovery operation.

Once this is complete, restart the server into normal mode. To complete the restore of the deleted object, an authoritative restore is required. This process is completely unchanged from Windows Server 2003, so for more information about the process of completing an authoritative restore, consult <http://go.microsoft.com/fwlink/?LinkId=68564>.

Locating Deleted Objects with DSAMAIN

A complexity with the object restore process is that the commands to complete the process require the distinguished name of each deleted object. One of the major pain points with using the Windows native tools in restoring deleted AD objects is locating exactly what has been deleted and determining their distinguished names.

The good news is that a new feature has been added to Windows Server 2008 that indirectly assists with this process. The DSAMAIN tool enables the creation and later mounting of AD database snapshots in parallel with the currently running instance. This parallel instance is then compared with the current instance to locate the deleted object as well as its distinguished name. The snapshots are not so much backups that can be restored. Rather, they are view-only representations of the database that are mounted as AD Lightweight Directory Services partitions using the NTDSUTIL command.

Before a snapshot can be mounted, it must first be created. To create a snapshot, use the command:

```
Ntdsutil snapshot "activate instance ntds" create quit quit
```

Among other information, the output of this command will be a GUID that is used to identify the partition when later mounted. If you need to list the available snapshot GUIDs, use the command:

```
Ntdsutil snapshot "list all" quit quit
```

To mount a previously created snapshot, replace *<GUID>* in the following command with the GUID of your snapshot of interest:

```
Ntdsutil snapshot "mount <GUID>" quit quit
```

The result of this command will look similar to Listing 2. Note in Listing 2 the location in which the snapshot has been mounted within the file system. This information is used in the next step.

```
C:\Users\Administrator>ntdsutil snapshot "mount {837e3bbc-dd34-2fed-8cb6-88832ef7658c}" quit quit

ntdsutil: snapshot
snapshot: mount {837e3bbc-dd34-2fed-8cb6-88832ef7658c}
Snapshot {837e3bbc-dd34-2fed-8cb6-88832ef7658c} mounted as
C:\$SNAP_200803281403_VOLUMEC$\
snapshot: quit
ntdsutil: quit

C:\Users\Administrator>
```

Listing 2: The result from mounting an AD snapshot using NTDSUTIL.

Once the snapshot is mounted, the DSAMAIN tool can be used to start the mounted snapshot as a parallel AD instance. Do this with the following command:

```
DSAMAIN -dbpath {pathToMountedAd}\WINDOWS\NTDS\ntds.dit -ldapport
{newLdapPort}
```

In this command, you will need to replace {pathToMountedAd} with the path shown in Listing 2. In the example there, the path is C:\\$SNAP_200803281403_VOLUMEC\$. You will also need to replace {newLdapPort} with an available network port that is not currently in use. Typically, a very high numbered port is used, such as 41000.

Upon starting the mounted snapshot, the snapshot operates much like a running instance of AD. You can use AD manipulation tools such as LDP to search within the snapshot to locate information about deleted entries. This information becomes useful in completing the authoritative restore step in the AD restore process.

Recovery Knowledge Is as Important as Backup Knowledge

Although some of the processes for completing AD backups and restores have become easier, others remain complex and involve plenty of command-line experience. Being aware of the success of backups and knowing the restore process is critical to getting your Windows domain back online after an accidental deletion incident occurs. The information presented here provides a good start towards assisting you with that critical restoration knowledge and experience.

Article 3: Understanding the Security Implications of Server 2008 RODCs

Technology changes. Processes change. Even business changes. But the venerable Windows domain remains. Microsoft's long-lived mechanism for consolidating authentication, security, and configuration control has seen a number of iterations in its life cycle. And yet the Windows domain has remained a near-constant within business IT environments since its inception.

What is of particular interest when one looks at the history of this network operating system (OS) is how history has a tendency to repeat itself. Servers that once were highly utilized later become underutilized as software struggled to keep up with advances in hardware. Later yet, the situation swings as virtualization consolidates low-use servers onto a single physical host. Disruptive technologies such as Terminal Services bring client/server computing back into the data center in ways much like the mainframes of yesteryear. With the release of Windows Server 2008, Microsoft presents us with a return of non-writeable domain controllers. Previously called Backup Domain Controllers (BDC), these new Windows Server 2008 constructs are now referred to as Read-Only Domain Controllers (RODCs).

History Repeats Itself

A strict comparison between a BDC and an RODC isn't complete without a look at the thought processes that have gone into their development. Windows NT BDCs were originally developed because of the need to ensure a single copy of the Active Directory (AD) database in a time before multi-master replication. In the traditional Windows NT network, the BDC housed a read-only copy of the AD database. It was intended for load balancing of incoming requests while housing an up-to-the-second backup should the Primary Domain Controller (PDC) go offline. Because the role of BDC was nothing more than a read-only version of the PDC, these capabilities were among its few benefits to the IT environment.

With the release of Windows 2000 Server and Windows Server 2003, Microsoft implemented multi-master replication to domain controller communication, which brought about the end for the BDC. Each domain controller was now an equal to every other, all containing a loosely contiguous copy of the AD database. This everything-to-everyone philosophy was a boon for large environments, as each domain controller could now both accept incoming authentication requests and make changes to the database itself.

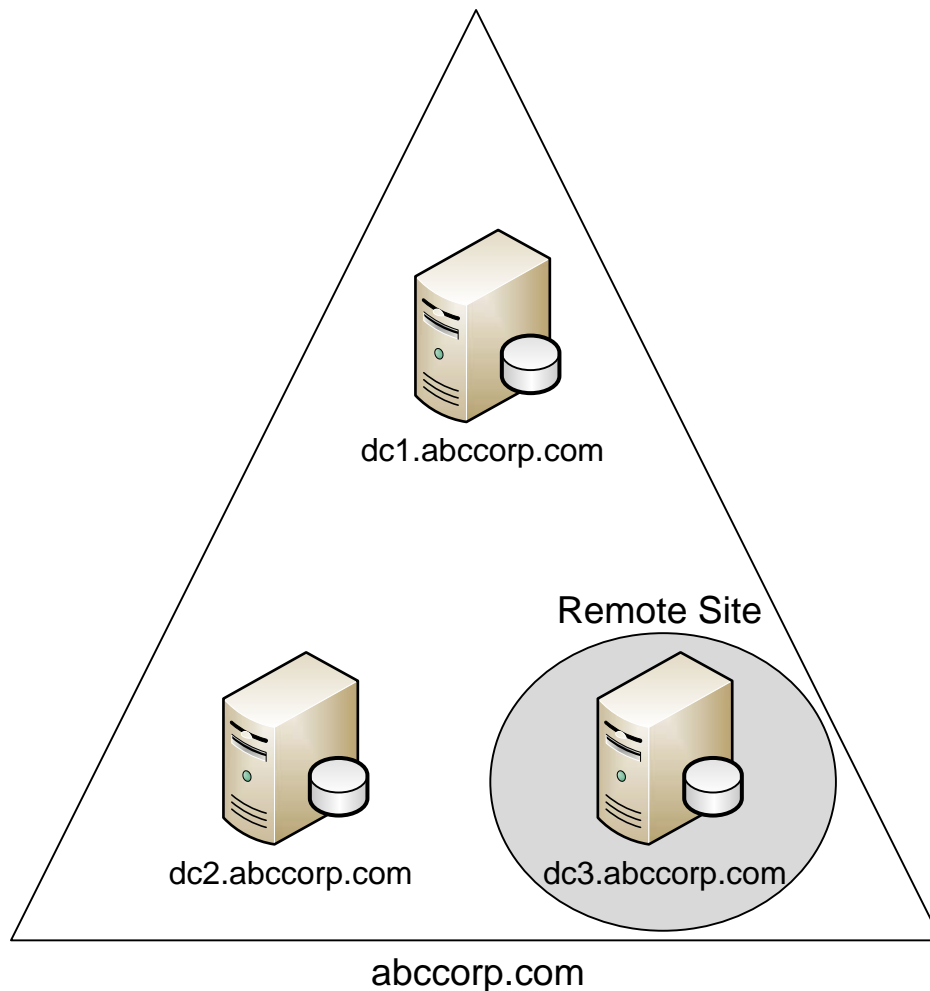


Figure 1: An AD domain with the directory database present on each domain controller.

But with these changes also came a problem. When each domain controller contains a full and complete copy of the AD database, as Figure 1 shows, the loss of even a single domain controller means the potential disclosure of an entire domain's worth of AD information. When this happens, usernames and passwords are lost, as is personnel information in situations in which AD records have been populated with HR information. With AD originally intended to be the database of record for employee information, both technical and personal, any domain controller loss would be a significant security incident for an IT organization.

The good news is that this problem isn't necessarily a high risk for all environments. In most, the physical theft of a domain controller is somewhat difficult. Data centers are typically highly secured locations, often requiring multiple mechanisms for entry. A would-be perpetrator would require substantial effort to physically enter a corporate data center and expect to get a domain controller's hard drives out the door without arousing some level of suspicion.

But not all IT environments store every domain controller in a locked-down data center. Quasi-secured locations and those with branch offices of only a few personnel are a particular concern. In a situation in which data center-grade physical security doesn't make financial sense, yet local domain controller access is needed, the result is often a domain controller that ends up under an employee's desk or stored in a closet. Although the physical removal of a domain controller from a data center is difficult, such situations bring to light a greater potential for data loss.

Introducing the RODC


Due to this problem, with Windows Server 2008, the venerable BDC makes a repeat appearance. This time, however, it arrives with new capabilities that make it a compelling fit for environments like those previously discussed. A Windows Server 2008 RODC is indeed a read-only copy of the AD database, but RODCs are different in that a Domain Administrator can choose which accounts are replicated to the RODC.

By selecting only those accounts that are local, the risk of deploying domain controllers to remote or quasi-secured locations significantly lessens. Looking back at Figure 1, if *abccorp.com* chose to deploy *dc3.abccorp.com* as an RODC, the Domain Administrator would have the added ability to replicate only the accounts appropriate to its remote site. If that domain controller is later lost, the only accounts with the potential for compromise—and therefore re-permissioning or re-creation—are those specific to the remote site.

Another problem with traditional domain controllers is their reliance on Domain Administrators for local administration. Unlike all other Windows servers, an administrator on a traditional domain controller must be a Domain Administrator. This requirement is to protect the AD database from error or compromise by a down-level administrator. But it also introduces a management headache for Domain Administrators who do not want or do not have time for the management responsibilities of the domain controller such as patching, configuration, application installation, and other administrative activities.

This headache is particularly challenging in the same remote site situations discussed earlier. In those sites, administrative responsibility for local servers is often assigned to a semi-trusted individual local to the site. This assignment allows the local person to complete patching and other operations without requiring central office IT assistance. Because of the security architecture intrinsic to domain controller alone, the only way to effectively allow this delegation to occur is to promote the part-time remote site administrator to a full Domain Administrator.

In developing the RODC, Microsoft has addressed this additional problem and created a new group designed to assist. On any deployed RODC, it is possible to grant local administrator rights to a user or group without needing to grant Domain Administrator privileges. Doing so enables far-reaching organizations to specify a local administrator with the privileges necessary to triage, troubleshoot, and otherwise administer the local RODC without needing additional Domain Administrator access. Figure 2 shows a screenshot from the DCPROMO process where this user or group can be identified.

 Be aware that the read-only nature of RODCs extends to DNS as well. If DNS zones are stored within AD, updates to those zones will need to occur on a full domain controller instance. The same holds true for applications that require writing to AD as part of their daily operations, such as Microsoft Exchange. These applications also will require a full domain controller in order to complete the writing and updating processes required for their operation.

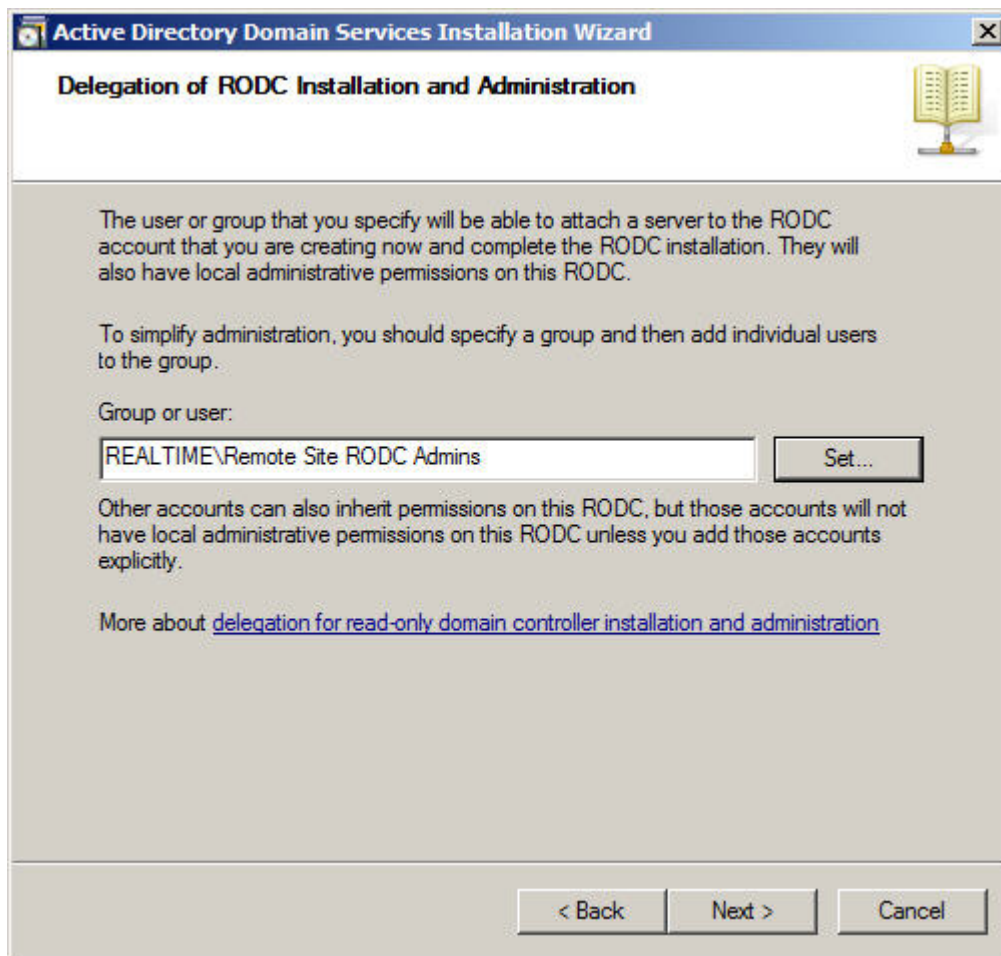



Figure 2: The DCPROMO process where you can grant local administrator rights to a user or group.

Creating an RODC

The process to create an RODC is nearly the same as the process to create a standard domain controller. Starting the process begins similarly as well. By entering *dcpromo* at the command prompt, the Active Directory Domain Services Installation Wizard starts. At the first screen, ensure that the check box is selected for *Use advanced mode installation*.

 Be aware that there are a few restrictions associated with RODCs. First, prior to the installation of any RODC, the domain schema must be modified to support their use. Do so by navigating to the `sources\adprep` folder on Windows Server 2008 media, and from a command prompt enter `adprep.exe /rodcrep`. Once complete, allow replication to occur between the domain controllers in your domain. In addition, an RODC cannot be the first domain controller created in a domain and cannot be the first Windows Server 2008 domain controller added to an existing domain. So, prior to starting down the path for RODCs, ensure that you have a full Windows Server 2008 domain controller in place first.

Complete the installation as you would for a typical domain controller promotion, entering in the pertinent information as requested by the wizard. When the Additional Domain Controller Options window appears, ensure that the *Read-only domain controller (RODC)* check box is selected.

The next screen within the DCPROMO wizard is titled Specify the Password Replication Policy (see Figure 3). It is here that the initial decisions are made about which passwords to replicate to this particular RODC. Individual users or groups can be added to the list and set to Allow or Deny. As with NTFS configurations, the Deny attribute overrides any Allow attributes selected for users within any groups. Because of this, high-risk accounts such as Administrators, Server Operators, Backup Operators, and Account Operators are by default added to the list with the Deny attribute set. By clicking Add, it is possible to add users or groups to the list.

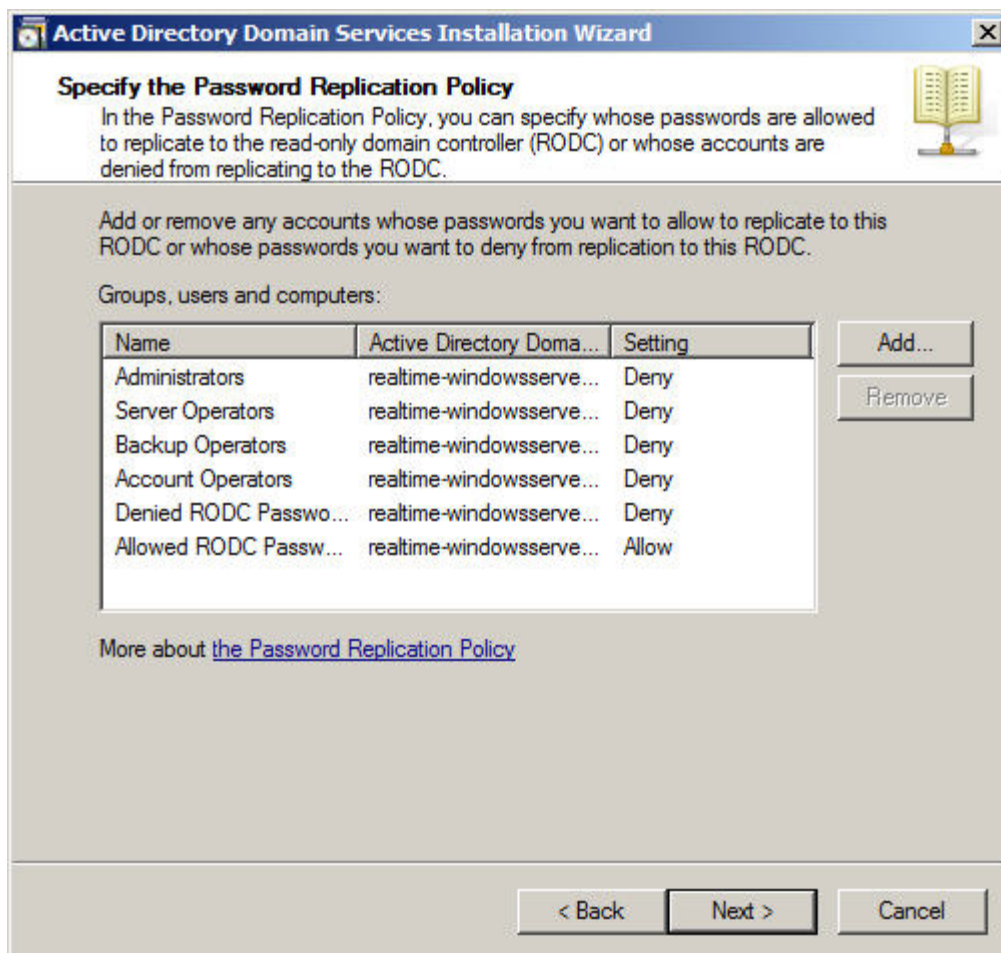


Figure 3: The DCPROMO process includes the initial creation of password replication policy.

As with NTFS permissions, it is a best practice to identify a Global Group for account replication rather than specific users. By default, the Allowed RODC Password Replication Group is created and configured to Allow. The Denied RODC Password Replication Group is a default group that is configured to Deny.

Clicking Next through this screen brings forward the window that we saw in Figure 2. There, you can select which user or group will have local administrator access to manage the RODC. Each of the remaining settings within DCPROMO is similar to those seen in a standard domain controller creation.

Once the RODC has been created, further management of its password replication policy is done through Active Directory Users and Computers. To do so, navigate to the RODC's computer object in Active Directory Users and Computers and choose Properties. In the resulting window, select the Password Replication Policy tab to bring forward a window that looks very similar to what we saw in Figure 3.

This window provides a few additional functionalities if you click Advanced. There, as Figure 4 shows, it is possible to view and export the users and computers whose accounts have been replicated to the local RODC. It is similarly possible to prepopulate passwords associated with those accounts by clicking the *Prepopulate passwords* button. Also possible is the generation of a Resultant Policy report available by selecting the Resultant Policy tab. All these added capabilities are present to further assist you with setting and ensuring that the right accounts are replicated to the RODC.

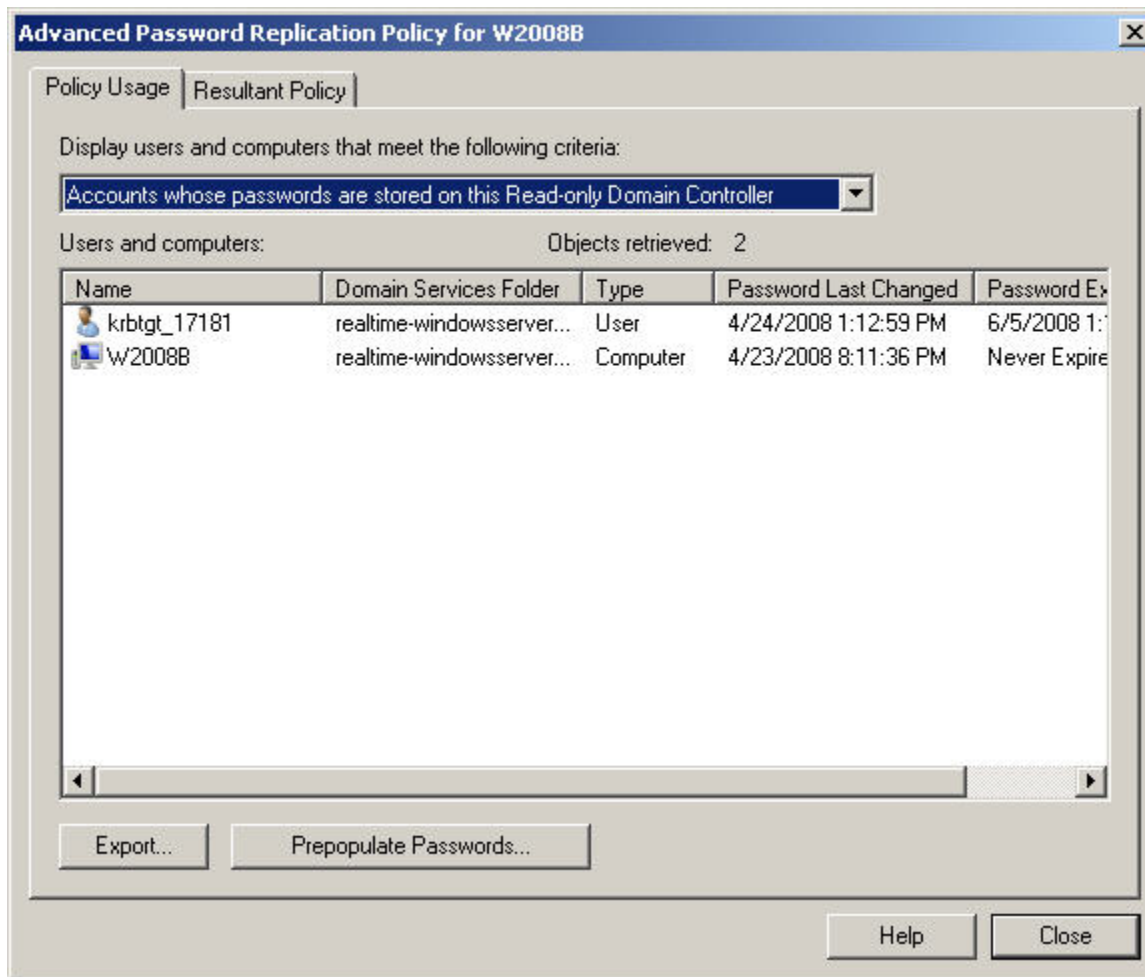


Figure 4: Additional advanced configuration of password replication policy is possible by viewing the properties of the RODC's computer object in Active Directory Users and Computers.

RODCs Protect the Domain

As you can see, RODCs are designed with the specific goal in mind of protecting the domain against the possibility of data disclosure. This additional level of protection is designed to limit the impact of a loss, ensuring that the vast majority of domain accounts remain intact should a single quasi-secured domain controller become lost. If your AD domain includes sites where domain controllers do not have the proper level of physical security, this feature makes Windows Server 2008 a compelling upgrade for its RODC capabilities.