

Realtime
publishers

"Leading the Conversation"

The Definitive Guide[™] To

Building a Windows Server 2008 Infrastructure



Greg Shields

Chapter 3: Active Directory Design & Domain Controller Management	53
A Good AD Design Solves Many Problems.....	54
Understanding the AD and Domain Controllers.....	56
The AD Forest.....	56
The AD Domain.....	56
Domain Controllers.....	56
Flexible Single Master Operation Roles.....	57
Functional Levels.....	58
Sites.....	59
Organizational Units.....	59
Domain Name Service	59
Best Practices in AD Design.....	60
Installing Domain Controllers.....	61
Installing the DNS Server Role.....	61
Promoting a Member Server with DCPROMO	63
Promoting Additional Domain Controllers.....	67
Upgrading Domain Controllers.....	73
Updating the Schema	74
Promoting a Member Server with DCPROMO	75
Relocating FSMO Roles	75
Demoting and Rebuilding Domain Controllers	76
Relocating FSMO Roles	77
Functional Levels.....	77
Read-Only Domain Controllers	77
AD Backup and Restore.....	80
Backing Up the AD Database	80
Restoring Individual AD Objects.....	80
Restoring Full Domain Controllers.....	81
AD Is a Central Part of Your Windows Infrastructure	81

Copyright Statement

© 2008 Realtimerepublishers.com, Inc. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtimerepublishers.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 Realtimerepublishers.com, Inc or its web site sponsors. In no event shall Realtimerepublishers.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.

Realtimerepublishers.com and the Realtimerepublishers 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 Realtimerepublishers.com, please contact us via e-mail at info@realtimerepublishers.com.

[**Editor's Note:** This eBook was downloaded from Realtime Nexus—The Digital Library. All leading technology guides from Realtimepublishers can be found at <http://nexus.realtimepublishers.com>.]

Chapter 3: Active Directory Design & Domain Controller Management

In our first two chapters, we've discussed the topics of server installation and management from the perspective of a single server. Chapter 1 dealt with the needs of installing an operating system (OS) to a particular set of server hardware. Chapter 2 discussed the management needs of individual servers, specifically using the new Server Manager tool that arrives with the release of Server 2008.

But in order to fully recognize your Windows Server 2008 infrastructure, it is likely that you'll be installing multiple servers in your environment. When the number of computers in an environment grows much beyond one or two, the need for a centralized mechanism for security, authentication, and authorization grows necessary. With Windows systems, that centralized mechanism is Active Directory (AD).

AD is at its core a directory of objects, much like a phone book directory. The directory contains information about the computers, users, and other configuration objects that are useful for its users. AD is also a source of information control and security. As AD becomes the database of record associated with these objects, it also serves as a location where these objects can identify themselves (identification), provide information that proves who they are (authentication), and request use of resources managed by AD (authorization). Thus, one of AD's major tasks is to provide the structure whereby resources such as files, folders, and registry keys among others are accessed in a controlled manner. AD provides a central location where security principals such as users and computers are assigned rights and privileges to access resources.

In this chapter, we'll be taking a high-level and introductory approach to describing the structure and function of AD as well as the process for installing AD into your computing environment. We'll discuss some best practices associated with the design of AD, and we'll conclude with one critical topic associated with the management of AD's Domain Controllers—ensuring proper backups and successful restores.

A Good AD Design Solves Many Problems

With its near-universal presence within business networks, it is likely that you've already come into contact with AD at some point in your past. Since its introduction with Windows 2000, AD has become the standard for directory services in nearly all Windows-based environments. Thousand-page books have been written on the subject of AD, each including deep descriptions of its various components and internal workings. So in a short chapter like this, we must be judicious about the types of topics we tackle. That being said, the intent with this chapter is to give you an overview of the structure, best practices, and installation steps associated with building AD as part of your Windows 2008 infrastructure.

AD is a complicated beast, with many configurations required in order to put it successfully in place. The good part about Microsoft's implementation of AD is that much of that configuration is automated and simple while at the same time retains the ability to scale to a worldwide presence. Installing a simple AD with a single site and few internal objects is relatively easy and can involve only a few minimal steps. At the same time, installing a large-scale AD with multiple sites, worldwide reach, and large levels of objects is also possible. Ultimately, the complexity of the environment dictates the level of customization required.

As stated in the header for this section, a good AD design will solve—or at least prevent—many IT problems right from the get-go. By incorporating good design into your AD, it is less likely to cause problems with authentication and user's loss of access to necessary resources.



As with many things, with AD, the simplest design is often the best solution.

In the early years of AD, with the release of Server 2000, many IT organizations saw an explosion of AD domains. Zealous IT administrators found themselves with a new tool to play with, and in many places, IT found AD domains as a tool for laying the groundwork for IT team responsibilities.

Unfortunately, the widespread creation of AD domains was not and is not necessarily the best solution with the business and its users in mind. Users and the business desire the greatest levels of transparency possible with authentication and resource security. Accessing resources across domain boundaries can involve extra steps and headaches for users when not properly set up. Because of the issues that large domain structures can pose, after the initial period of domain hypergrowth, many IT administrators found themselves later consolidating domains into fewer and fewer numbers.

Considering this, a good AD design provides the following benefits to the organization (Source: <http://technet2.microsoft.com/windowsserver2008/en/library/23d96652-a0d9-4f70-9742-514110c99da61033.msp?mfr=true>):

- Simplified management of Microsoft Windows–based networks that contain large numbers of objects. Objects can be managed through a unified interface, which eliminates management duplication and enhances the ability to manage the environment with fewer human resources.
- A consolidated domain structure and reduced administration costs. As stated previously, when domain structures are consolidated into as few as possible, this reduces the overall operational expenses associated with managing those constructs.
- The ability to delegate administrative control over resources, as appropriate. When objects are available within a centralized directory, the ability to secure those objects and assign rights and permissions to them becomes much easier.
- Reduced impact on network bandwidth. The correct network positioning of AD's Domain Controllers is critical to ensuring the lowest levels of network usage.
- Simplified resource sharing. Resource sharing information can be stored within AD's database, which allows users to easily search for and find resources they need.
- Optimal search performance. Searching of managed objects is also improved when they are stored within a single database of record.
- Low total cost of ownership. In contrast with environments in which each computer is managed independently, the centralized control improves the ability for administrators to manage through policies and reduces the cost to operate the environment.

As you'll learn in the next few sections, creating the best user experience for the users of a Windows infrastructure is very much a function of creating AD based on best practices. When an AD design is properly created based on the needs of its users, those users will be able to easily authenticate, locate information and resources, and ultimately do their jobs.

Understanding the AD and Domain Controllers

Without too many details, let's take a few minutes to discuss AD's major components. Understanding these components in relation to each other is critical to understanding how best to design your AD.

The AD Forest

An AD forest is a collection of one or more AD domains. These domains share a common logical structure, directory database schema and configuration, and scope of search. Domains, which we'll discuss next, are integrated into a forest when there is interest in sharing resources and respecting authentications between them. Whereas domains are commonly used as administrative boundaries, forests are available so that users across those boundaries can easily share information and work collectively. As domains within a forest implicitly trust each other, groups that assign resource privileges can be shared across domain boundaries.

The AD Domain

Domains are considered partitions of an AD forest. They are the boundary of user and resource administration and are the construct in which objects reside that are exposed to the user. A domain is the location where user identities are stored and where object authentication occurs. Users login to their respective domains and make use of objects within that domain. Once an object is authenticated, it can make use of approved resources within that domain as well as other resources as specifically identified in other domains within the forest.



Many domains can make up a forest, and each domain can belong to only one forest. A forest can contain either a single domain or multiple domains. Domains also can be linked to each other to create a hierarchy within the forest. The structure of how domains are arranged within a forest is usually based on intra-IT responsibility and the level of trust between IT organizations. It can also be set based on geographical considerations or along lines of business.

Domain Controllers

Each domain requires a minimum of one Domain Controller to perform the management functions associated with the domain. Each Domain Controller hosts a copy of the AD database and handles the responsibilities of authentication and authorization. To ensure that the AD database is consistent across all Domain Controllers, each Domain Controller engages in replication with other Domain Controllers within the domain. Domain Controllers are replicated as peers without multiple-level hierarchies among any Domain Controllers within a domain. This replication process ensures that the database is loosely consistent between every Domain Controller within the domain, and that any Domain Controller can be used as a source of authentication or resource authorization.

Because of this replication, it is possible for any Domain Controller to be used as the location whereby users login to computers on the domain. As domains can be widely dispersed—even globally—this replication process allows users to use any domain controller anywhere to search for and work with resources.

Flexible Single Master Operation Roles

In moving AD to peer-to-peer replication, five specific functions were isolated as needing to run on a single, dedicated host. It was not possible to run these five functions in the same distributed concept as is done with the other components of AD and its peer-to-peer Domain Controller interrelations.

The five functions are called the Flexible Single Master Operation (FSMO) Roles. They're flexible because any Domain Controller can fulfill the role. They're single master because, as we discussed earlier, they cannot operate in a peer-to-peer concept. The five FSMO Roles, as defined by Microsoft (Source:

http://www.microsoft.com.nsatc.net/technet/prodtechnol/windows2000serv/reskit/distrib/dsfl_utl_pavr.msp?mfr=true):

- **Schema Operations Master**—There is a single schema operations master role for the entire enterprise. This role allows the operations master server to accept schema updates. There are other restrictions on schema updates.
- **Relative ID Master**—There is one relative ID master per domain. Each domain controller in a domain has the ability to create security principals. Each security principal is assigned a relative ID. Each Domain Controller is allocated a small set of relative IDs out of a domain-wide relative ID pool. The relative ID master role allows the Domain Controller to allocate new subpools out of the domain-wide relative ID pool.
- **Domain-Naming Master**—There is a single domain-naming master role for the entire enterprise. The domain-naming master role allows the owner to define new cross-reference objects representing domains in the Partitions container.
- **PDC Operations Master**—There is one Primary Domain Controller (PDC) operations master role per domain. The owner of the PDC operations master role identifies which domain controller in a domain performs Windows NT version 4.0 PDC activities in support of NT 4.0 Backup Domain Controllers (BDCs) and clients using earlier versions of Windows.
- **Infrastructure Master**—There is one infrastructure master role per domain. The owner of this role ensures the referential integrity of objects with attributes that contain distinguished names of other objects that might exist in other domains. Because AD allows objects to be moved or renamed, the infrastructure master periodically checks for object modifications and maintains the referential integrity of these objects.

Functional Levels

Each iteration of AD—starting with Windows 2000, through Windows Server 2003, and now with Windows Server 2008—has added additional functionality to AD’s core set of features. Enabling that added functionality is done by setting the Forest Functional Level and Domain Functional Level to the correct level. Raising the functional level requires that all Domain Controllers within that level’s scope (either in the domain or in the forest) be running at that level. Hence, in order for the Domain Functional Level to be raised to Windows Server 2008, all Domain Controllers within the domain must be running Server 2008. No Server 2003 Domain Controllers can remain in the environment.

Raising any functional level requires a manual “switch” to be flipped to enable the change. This separate step is required so that higher-level Domain Controllers can be introduced incrementally into a lower-level AD domain. As we’ll explore later in this chapter, this allows for a rolling upgrade of Domain Controllers until all are at the updated OS.

With Server 2008, there are actually no new features gained in upgrading the Forest Functional Level to Windows Server 2008. However, individual domains do gain new functionality:

- The SYSVOL replication engine is updated from using the File Replication System (FRS) to the new and more reliable Distributed File System (DFS).
- The Advanced Encryption Standard (AES) in both AES 128 and AES 256 is supported for the Kerberos protocol.
- Last interactive logon information is logged.
- Fine-grained password policies are enabled, which provide the ability to create multiple password policies within a single domain.

Sites

When domains grow geographically large, that replication process also can grow complex. Sites are an AD construct that allows users within a localized area of high network connectivity to ensure that they are authenticating and requesting resource access from Domain Controllers in an area of the network close in proximity. Sites are intended to be arranged by geographical area and are specifically linked to the network subnets that correspond to that geographical area.

Three main elements are linked to site membership.

- **Replication**—Replication between Domain Controllers is configured based on their site membership. Domain Controllers within the same site are assumed to have high network connectivity between them. Thus, their replication occurs more often with less consideration for network performance and capability. Domain Controllers in different sites replicate at lesser intervals so as not to impact the network between them.
- **Authentication**—Clients that attempt to authenticate will first look to Domain Controllers within their local site. This ensures that clients can complete the login process as quickly as possible. When Domain Controllers are unavailable in the local site, clients can then seek elsewhere for a Domain Controller to complete the process.
- **Client Configuration**—Using Group Policy, clients can be configured based on their site membership. This allows for clients to receive necessary configurations based on the site in which they currently reside.

Organizational Units


Organization Units (OUs) provide useful constructs to simplify management of AD objects. OUs collect AD objects into groups so that policy-based configurations can be applied to those objects. The greatest use of OUs is for assignment of Group Policies to objects that reside within the OU. Any AD object can only reside in a single OU.

 Chapter 6 will discuss OUs in detail.

Domain Name Service

Network resolution of the elements that make up AD is critical so that clients can find servers and Domain Controllers and Domain Controllers can locate each other. Domain Name Service (DNS) is the network protocol that handles the resolution of servers and services. Using DNS, clients are able to locate Domain Controllers within the domain.

Windows DNS and AD also make use of Service (SRV) records. These special records are used by clients to identify specific domain services and the servers that host those services. Because SRV records are hosted by DNS, and because DNS has the ability for dynamic updates, Domain Controllers can update the location of available services on the fly by manipulating their presence in DNS resolution.

 Microsoft's Web site includes hundreds of pages of additional information about the basic functionality of each of the components within the structure of Windows Server 2008. For more information, check out <http://technet2.microsoft.com/windowsserver2008/en/library/b1baa483-b2a3-4e03-90a6-d42f64b42fc31033.mspx>.

Best Practices in AD Design

In many ways, there are as many potential AD designs as there are ADs in the world. Thus, giving hard and fast advice as to the configuration of AD for your particular environment is challenging. That being said, there are a number of best practices associated with the design of AD that make sense across all instances. These best practices can assist you with creating an AD that works best for both administrators and users.

Consider the following short list as a set of guidelines to assist you with creating your domain structure. There are numerous other resources available on the Internet that can provide additional best practices focused toward specific case studies.

Forest and Domain Creation:

- Resources can be accessed across domain and forest boundaries but with additional steps to the user. Thus, minimizing the overall number of domains in the forest assists with providing best-possible access for users to their needed resources. When organizationally possible, a single domain structure is the best possible technical solution.
- The same holds true for forests. Consolidating resources into a single forest when organizationally possible is similarly the best possible technical solution.
- Due to a way in which Domain Administrator privileges can be maliciously elevated across domain boundaries, AD's boundary of security is the forest. If there are concerns about privilege escalation between Domain Administrators in separate domains, consider the use of multiple forests.
- When multiple domains are required, consider creating domains based on geographic scopes as these are less likely to change than organizationally based domains. Windows domains are long-lasting entities, and thus organizationally based domains are more likely to require change considering the long-term flow of business and that domain restructures tend to be expensive activities.

Site Creation:

- An AD site is intended to be bounded by a region of high network connectivity. Establish as a site every geographic area of high network connectivity based on IP subnet addresses.
- Place at minimum one Domain Controller in every site and make at least one Domain Controller in each site a Global Catalog (GC).
- Windows and the Knowledge Consistency Checker (KCC) service have the ability to automatically determine the best site topology. Manually creating site links stops this automated process. It is a best practice to allow the KCC to manage site links automatically rather than to do so manually.

OU Creation:

- When possible, create separate OUs for user and computer objects. This assists with the deployment of Group Policy.
- Consider creating as few OUs as possible. Create additional groupings only when Group Policy targeting mandates the group creation.

Installing Domain Controllers

Architecting a good AD design is in many ways the “hard part.” Once an AD design is established and ready for implementation, the actual installation of Domain Controllers can be a relatively trivial task. That being said, there are a few steps you need to know in order to properly install your first and subsequent Domain Controllers. In this section, we’ll go through a lengthy step-by-step process of installing a new Domain Controller to create a new domain as well as the needed additional Domain Controllers put in place for high availability. This section will include a few extra steps done manually that could otherwise have been automated so that you gain an understanding of the entire unaided process associated with domain creation.

Also in this section, as many organizations already have AD in place, we’ll discuss the procedure to upgrade an existing AD from Windows Server 2003 R2 to Windows Server 2008.

Installing the DNS Server Role

Although the Domain Controller promotion process can automatically install and configure DNS for you, it is usually a good idea to start any domain creation with a manual installation of DNS. With previous versions of Windows Server, the DCPROMO process historically has not done a good job of fully configuring all the pieces of DNS for operation with an AD domain. That process has gotten quite a bit better with the release of Server 2008, but even with its new capabilities, it is a good idea for you to understand the process so that you have the skills you need for later troubleshooting.



For the purposes of this chapter and this guide, we will be creating a domain and associated forward and reverse DNS zones named *realtime-windowsserver.com* on the two Domain Controllers *w2008a* and *w2008b* and for the *192.168.0.0/24* subnet. These DNS zones and their associated domain will host each of the resulting servers and services that we discuss throughout this guide.

To install the DNS Server Role, use the following steps. First, from Server Manager, right-click the Roles node and choose to Add Roles. From the resulting wizard, make the selection to install the DNS Server Role. The DNS Server Role has no additional Role Services, so its installation via Server Manager involves no additional configuration. Restart Server Manager after the installation of the role.

Once the DNS Server Role is installed, we'll then need to prepare it for use by AD. In this step, we will be configuring the DNS server as well as creating and configuring the *realtime-windowsserver.com* zone. From Server Manager, click Roles | DNS Server | DNS | w2008a, and select Properties. Many of the server settings for DNS remain the same from their default configurations. However, we'll want to pay special attention to a few.

- Forwarders Tab—If this server finds that it cannot resolve a particular request, we can instruct the server to ask another server for an answer. This process is called Forwarding. Note that this is not necessarily needed for Internet forwarding except in the case where this server is incapable of contacting other Internet-based DNS servers. Click the Edit button here to enter any servers to be used for forwarding.
- Advanced Tab—DNS in Windows 2008 is often configured to support dynamic updates. This allows clients to update their DNS records when their IP address changes. One result of this is that some DNS records can grow stale over time if they are not properly updated. When stale records are not cleaned out of the DNS database, they cause problems with proper resolution. Selecting the check box to enable automatic scavenging of stale records will enable the DNS server to remove records that have aged past a certain number of days.

Once we've completed configuring the DNS server itself, we'll need to create a forward lookup zone. This will resolve fully-qualified DNS names (FQDNs) to IP addresses. To create the forward-lookup zone, right-click Forward Lookup Zones, and select New Zone. When prompted, choose to create a new Primary Zone with the name *realtime-windowsserver.com*. Use the default for the zone file name and configure the zone to allow both nonsecure and secure dynamic updates.

We'll also need to create a reverse lookup zone to resolve IP addresses to FQDNs—the reverse of the zone created previously. To do so, right-click Reverse Lookup Zones, and select New Zone. When prompted, choose to create a new Primary Zone of type IPv4 Reverse Lookup Zone. Use the Network ID 192.168.0, select the default for the zone file name, and configure the zone to allow both nonsecure and secure dynamic updates.

Next, we'll need to populate this new zone with the information about our server w2008a. To do so, we need to ensure that the full computer name for this server is set to *w2008a.realtime-windowsserver.com*. Do this by right-clicking Computer, and choosing Properties. From the resulting screen, click the link for Change settings under Computer, name domain, and workgroup settings. Click Change in the resulting screen and then the More button to bring forward the DNS Suffix and NetBIOS Computer Name screen. Enter *realtime-windowsserver.com* as the Primary DNS suffix of this computer. Changing the primary DNS suffix will require the computer to restart. Once the computer has restarted, return to Server Manager and verify that the entries are configured in the forward and reverse lookup zones similar to what you see in Figure 3.1.

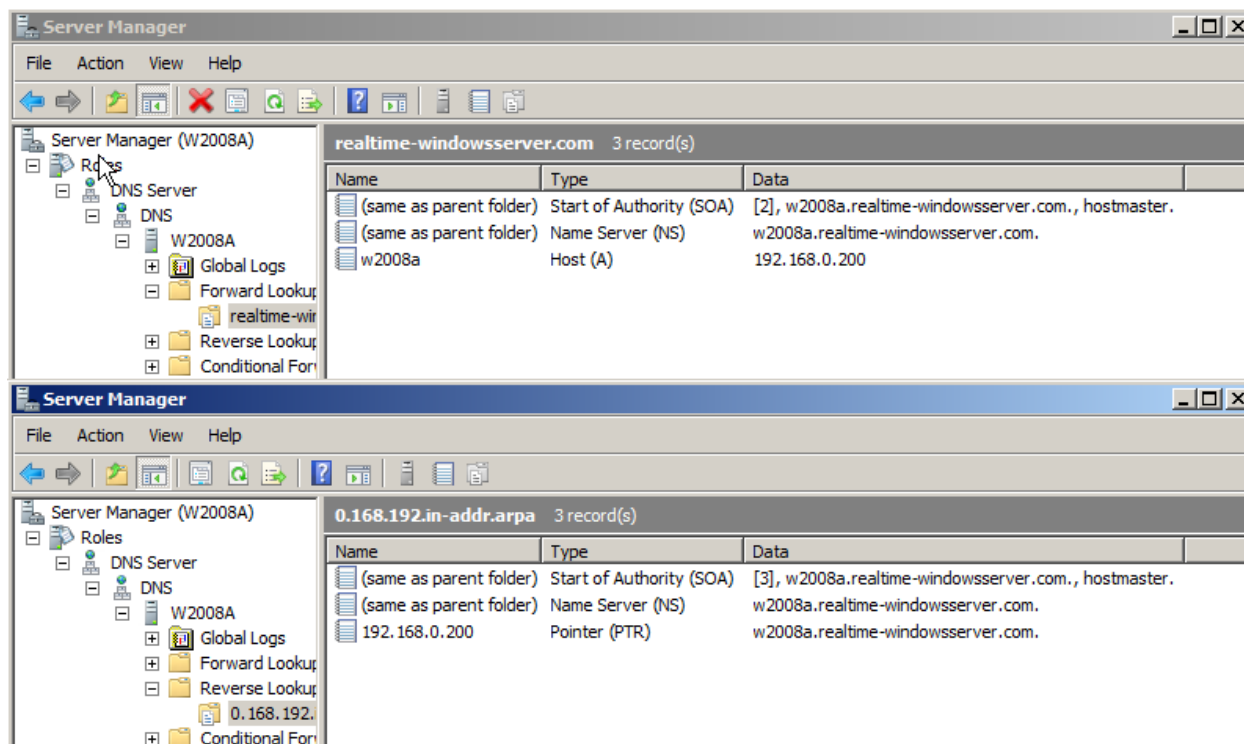


Figure 3.1: Two iterations of Server Manager, showing the correct forward and reverse lookup zones needed to start creating our Windows domain.

Promoting a Member Server with DCPROMO

Once we've completed this process, we can begin the process of elevating our member server to a full Domain Controller. Doing so involves the use of the DCPROMO command. Though AD Domain Services is considered a role in Server 2008, its installation can only be done through DCPROMO rather than through Server Manager as it is with other roles. To elevate w2008a, we'll use the following process:

- From a command prompt, run the command *dcpromo*. The ADDS binaries will be installed, which will take a period of time.
- Once complete, the ADDS Installation Wizard will be presented. It is always a good idea to mark the check box to *Use advanced mode installation* so that we are presented with all the possible options for creating a new domain. Do this now and click Next.
- When asked to Choose a Deployment Configuration, select *Create a new domain in a new forest*. Then, enter *realtime-windowsserver.com* as the FQDN of the forest root domain. In our example, we are creating a new forest that contains only a single domain. If we were creating a down-level domain in an existing forest, we would instead need to include the root-level domain here.
- For Domain NetBIOS name, use the name *REALTIME*. This is the NetBIOS name for the domain and is what is shown at the logon screen when any member attempts to logon. For an example of how the logon screen will look for Server 2008 clients once they join the domain, see Figure 1.1 in Chapter 1.

- In the screen titled Set Forest Functional Level, we will set the level to Windows Server 2008. As this is a brand new domain with only one member, we do not need to worry about down-level Domain Controllers.
- The screen titled Create DNS Delegation will ask whether we want to create a DNS Delegation. This screen is used when we are attempting to create a domain and DNS zone that is a child of an existing DNS zone. In our case, our domain and DNS zone are equivalent to the zone we just created, so this screen is effectively unnecessary. For now, click Yes and then enter administrator credentials into the resulting box to force the promotion process to attempt the process anyway. You will see an error during the domain creation process related to this selection that you can safely ignore.
- In very large domains and forests, the AD database, log files, and SYSVOL can grow to become very large. They can also require a very fast disk subsystem to ensure best performance. In the case of a very large domain and forest, the next screen provides the ability to relocate these files to another location or disk drive. In our case, our domain will be very small, so we can leave these settings untouched.
- The Directory Services Restore Mode Administrator Password in the next screen is used in restoring all or portions of the AD database after an accidental deletion or a disaster. Enter a password in the boxes here and ensure this password is kept in a safe location.
- The final Summary screen discusses all the settings configured within the DCPROMO wizard. You'll also see here a button titled *Export settings*. Clicking this button forces the DCPROMO process to export a text file that contains the settings configured in the past screens. This file can be especially handy when creating Domain Controllers on Server Core instances, which we'll discuss in Chapter 5. For now, click Next to begin the creation of the domain and the installation of ADDS.

At this point, the ADDS Installation Wizard will begin the process of installing ADDS and creating the domain. A check box titled *Reboot on completion* can be marked to instruct the process to reboot the computer once complete. In any case, a reboot is required to complete the domain creation and installation of ADDS.



Figure 3.2: The ADDS Installation Wizard going through its process of installing ADDS and creating our domain.

Once the reboot is complete, we will have successfully created the realtime-windowsserver.com domain. We can double-check this in a number of ways. First, after the reboot, you will want to log on to the domain as REALTIME\Administrator using the correct password. Once there, check the event log for an Event ID 29223 from source LsaSrv that occurs just before the reboot occurs. The text of this event should read *This server is now a Domain Controller* similar to the image in Figure 3.3. There should be few if any errors in the event log.

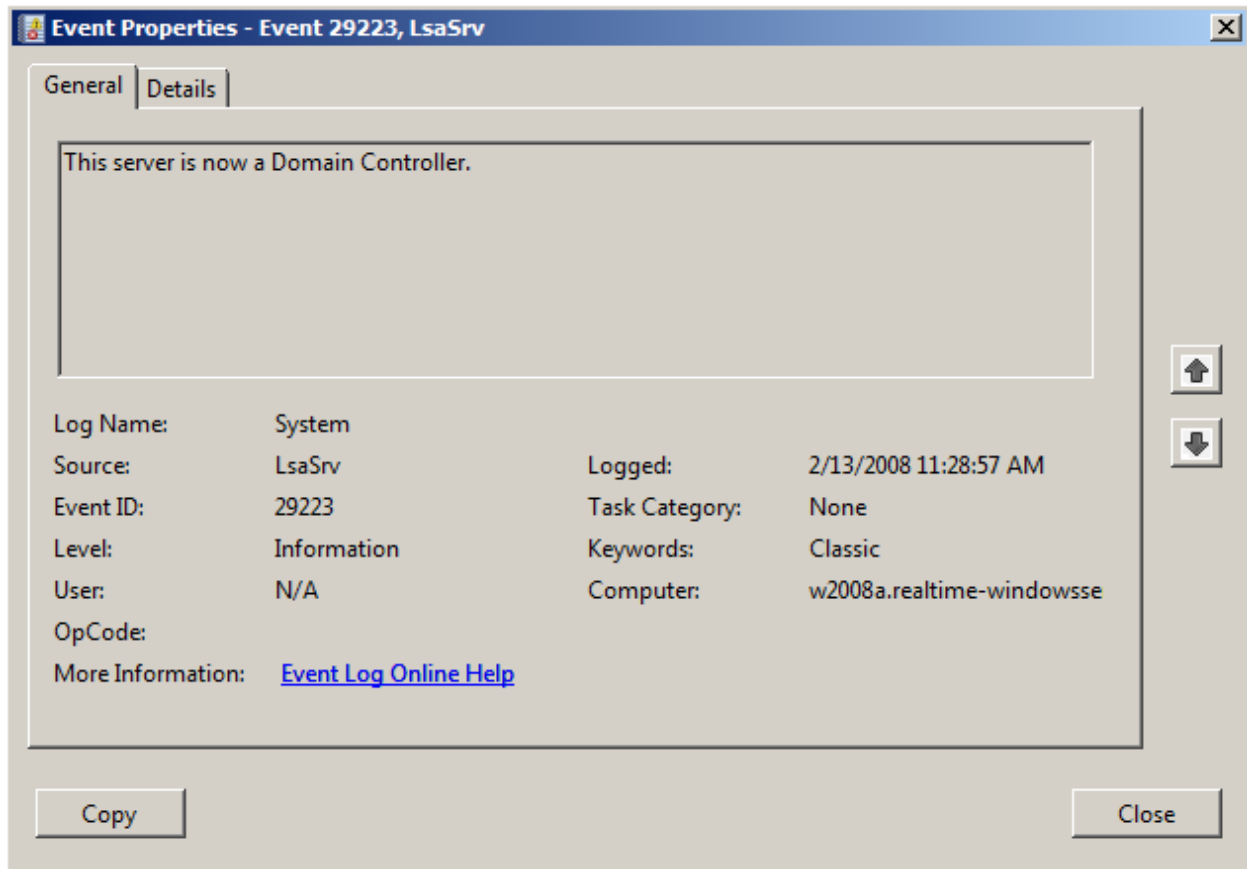


Figure 3.3: The event log will provide an event showing the successful promotion of the member server to a Domain Controller.

Another test to verify the success of the domain creation is to look in DNS for the presence of the domain's SRV records. These records are necessary for clients to find domain-related services. You'll see five new sub-zones under the realtime-windowsserver.com forward zone, similar to those shown in Figure 3.4. You'll find that under each of these five sub-zones are numerous additional zones. Without getting into too much of the detail of each of the zones, a good test is to walk through each of the individual zones present and look for anything out of the ordinary with their presentation. As our new domain only contains a single Domain Controller (and, really, a single server) named w2008a, any entries should point to this server.

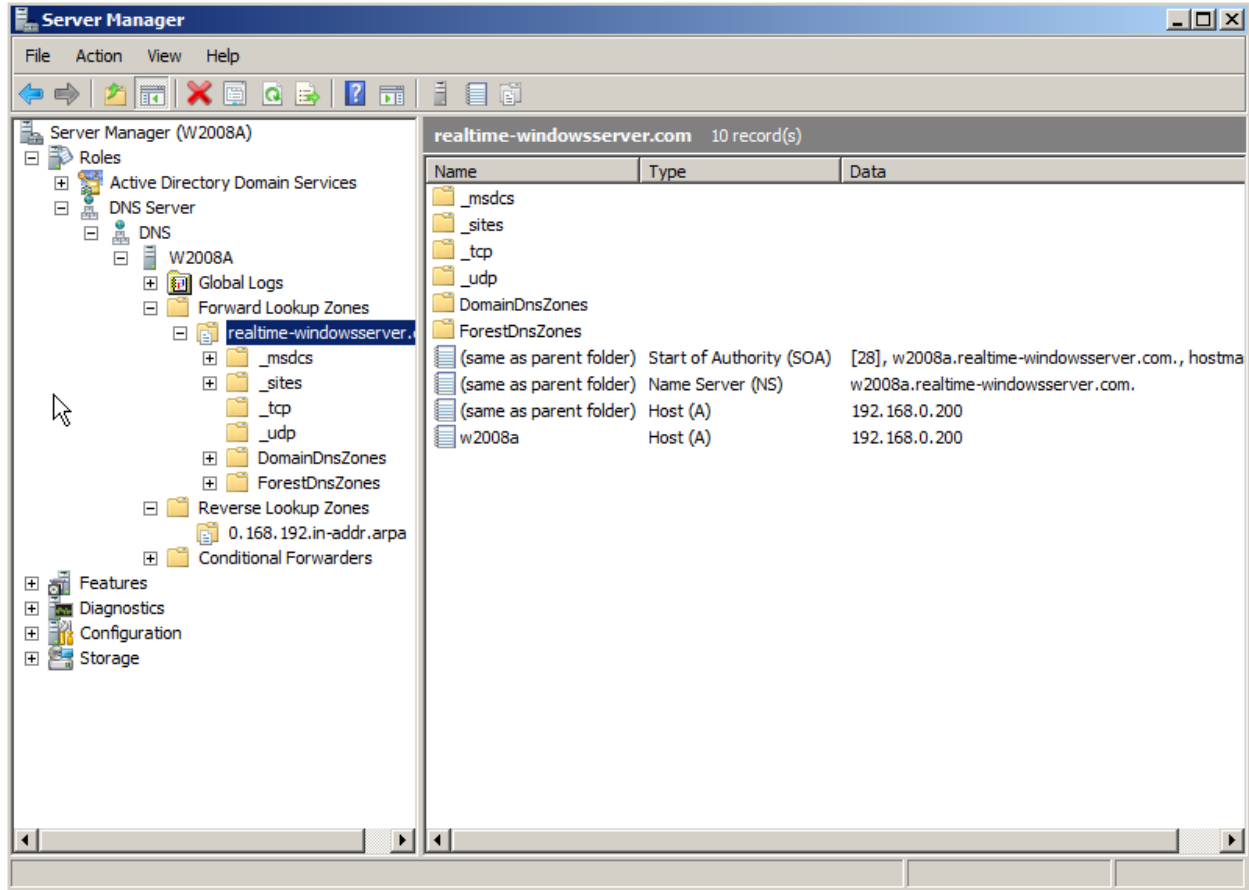



Figure 3.4: Upon the creation of the domain, a number of new sub-zones will be found under our initially created zone. These are for use by clients in identifying domain-related resources.

Promoting Additional Domain Controllers

Because AD is the backbone upon which all our data and applications reside, it is always a good idea to include no less than two Domain Controllers for every domain. This ensures that should one Domain Controller go offline or incur problems, there is always another that can service clients and manage the operations of the domain.

 Many very small networks have chosen not to build and manage two Domain Controllers due to cost but have later on experienced significant outages due to the loss of their only Domain Controller. It is very important to always plan for a minimum of two Domain Controllers per domain.

Once we've installed our first Domain Controller, we will install our second for redundancy. The process to install the second Domain Controller is much easier than the first, as the second Domain Controller gathers much of its configuration requirements from the first Domain Controller. To do this, first install the Server 2008 OS onto another computer to create a member server. For this example, configure the second server, w2008b, to point to w2008a as its DNS server and add it to the realtime-windowsserver.com domain.

In our example, both of our Domain Controllers will also act as DNS servers to provide redundancy for both services. So to begin, login using an account with Domain Administrator privileges and install the DNS Server Role using the same method as outlined earlier. Once complete, we'll set up this server to operate as a secondary DNS zone:

- For server-specific settings, set the configuration of the Forwarders and Advanced tabs in the same way as done for the first DNS server.
- We will need to create secondary forward and reverse zones on w2008b that point to w2008a. To do so for the forward zone, right-click Forward Lookup Zones and select New Zone. When prompted, select to create a Secondary Zone and enter realtime-windowsserver.com as the zone name. In the Master DNS Servers screen, enter the IP address for w2008a. If correct, the column titled Validated will display OK and a green checkmark will appear to the left of the IP address.
- For the reverse zone, right-click Reverse Lookup Zones and select New Zone. When prompted, select to create a Secondary Zone, an IPv4 Reverse Lookup Zone, and enter 192.168.0 as the Network ID. In the Master DNS Servers screen, enter the IP address for w2008a. If correct, the column titled Validated should display OK and a green checkmark will appear to the left of the IP address.
- Upon completing the previous two tasks, you'll immediately see that the zone cannot be loaded and displays an error. This occurs because of a security feature with DNS. DNS zone transfers are usually configured to be explicitly allowed, a setting we have not yet configured on w2008a. To allow w2008b to transfer the zone, return to Server Manager on w2008a and on both the previously created forward and reverse zones, select the Name Servers tab. On each, click the Add button. Enter the FQDN for w2008b and its IP address and click OK. For both the forward and reverse zones, the resulting tab should resemble Figure 3.5. Note that you may see an error message when attempting to do this. That error message can be safely ignored.
- After a few minutes, navigate back to Server Manager on w2008b and hit the F5 key to refresh the zone. If everything is correct, after a short delay, the zone should appear on w2008b similar that is an exact match to what is seen on w2008a.

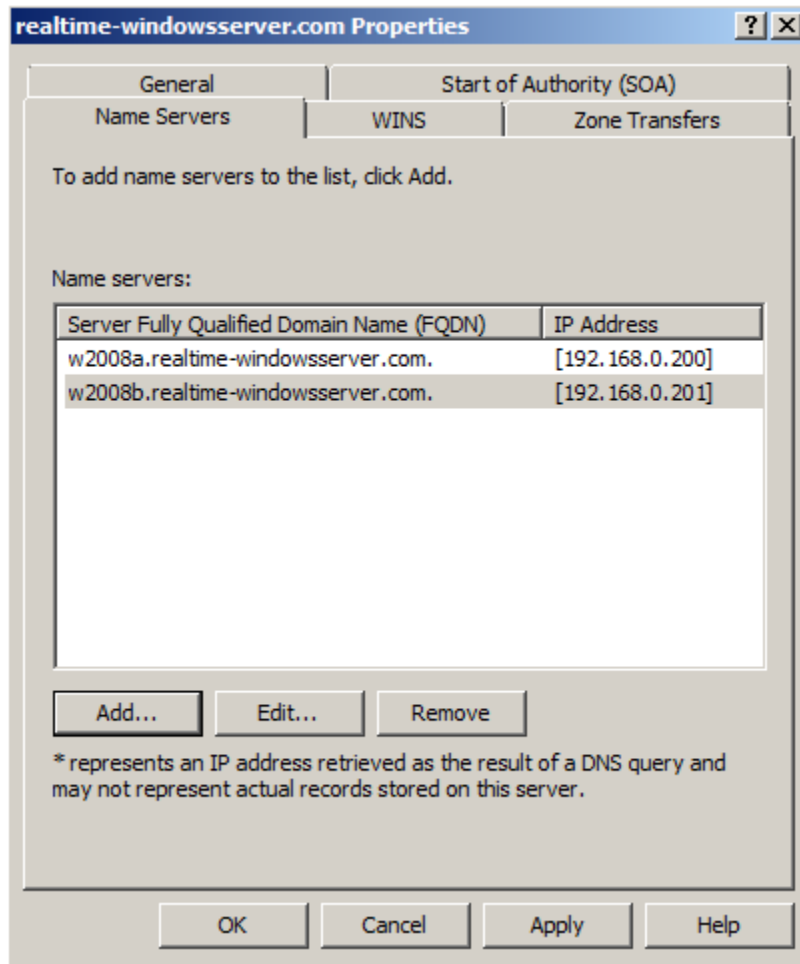


Figure 3.5: The Name Servers tab allows other servers to perform zone transfers.

Our next step is to complete the promotion process using DCPROMO. To start this process, from a command prompt, enter DCPROMO to bring forward the wizard and complete the following steps:

- As before, ADDS will start by installing its needed binaries. When control is returned, mark the checkbox to *Use advanced mode installation*, and click Next.
- Select to create this Domain Controller in an *Existing forest* and to *Add a domain controller to an existing domain*.
- Because we have already added this server to the realtime-windowsserver.com domain, in the next screen, the domain will have already been populated. We can also choose to use our existing credentials because we have logged in as a Domain Administrator. In the following screens, choose the realtime-windowsserver.com domain and the site named Default-First-Site-Name.
- At the Additional Domain Controller Options screen, ensure that the box is checked to make this server a Global Catalog. Do not select the box to make this a Read-only Domain Controller.

- The next screen titled *Install from Media* allows you to choose how you want to replicate the domain data to this new Domain Controller. In the case of a very large domain with a large database and a very slow network connection, this option allows you to replicate data using offline media such as a DVD. This is very handy when the replication of domain data could saturate our network connection. In our case, our domain is small and well-connected, so choose to *Replicate data over the network from an existing domain controller*.
- The following screen titled *Source Domain Controller* allows us to select the Domain Controller from which to replicate data. In dispersed networks with many Domain Controllers, it is possible for this promotion to choose a Domain Controller in a far removed site, which can increase the time to complete the process and/or saturate the network link between this site and the remote site. In that case, selecting a Domain Controller in close network proximity reduces the effect of the replication. Since our Domain Controllers are close in network proximity to each other, we can safely choose to *Let the wizard choose an appropriate domain controller*.
- The next two screens allow us the option to relocate the database, log, and SYSVOL files and set the Directory Services Restore Mode password. We'll use the same settings and password as before in these two screens.
- At the Summary screen, click *Next* to start the ADDS installation process to promote our second member server to a Domain Controller. Once the installation and subsequent reboot is complete, we will have successfully installed ADDS onto a second server.

Once the Domain Controller promotion is complete and the reboot has occurred, log back into the server as a Domain Administrator and check the event log as before to verify that this server has successfully promoted.

Once we have completed this process, we need to make a few modifications to DNS to move its database into AD, lock down dynamic updates, and enable record scavenging on our individual zones. This ensures that DNS is operating in conjunction with AD in the best and most secure way possible:

- Navigate to the DNS Server node of Server Manager on w2008a and bring forward the Properties screen for our forward lookup zone.
- On the General tab, click the *Change* button. In the resulting screen, mark the box for *Store the zone in Active Directory (available only if DNS server is a domain controller)*. By doing this, we are moving the storage of DNS records out of files on the server to the AD database itself. This allows DNS information to be replicated throughout the domain through standard Domain Controller replication. Click *Yes* when asked *Do you want this zone to become Active Directory integrated?* Click the *Apply* button to convert the zone.
- For the drop-down box next to *Dynamic updates*, change the selection to *Secure only*. This forces clients to authenticate to the DNS server prior to updating records which prevents a rogue client from maliciously manipulating client DNS data.
- Click the button titled *Aging*. In the resulting screen, mark the box for *Scavenge stale resource records*. The aging and scavenging process we discussed earlier requires both a server-based and zone-based configuration. Selecting this check box fully configures the server to start the aging and scavenging process for this zone.

- Complete the previous three steps on the reverse zone to complete its configuration.
- Now, navigate to the DNS Server node of Server Manager on w2008b. Here, for both our forward and reverse zones, click the Change button to change the zone type from Secondary to Primary and click OK. Click the Change button again and mark the box for *Store the zone in Active Directory (available only if DNS server is a domain controller)*. Also, as before, on both forward and reverse zones set *Dynamic updates* to *Secure only* and click the Aging button and mark the box for *Scavenge stale resource records* to enable aging and scavenging for these zones.
- Lastly, change the network settings for both servers so that each points to itself as a primary DNS server with the other as a secondary DNS server.

This completes the process of building our sample domain. Now be aware that technically we've done this the hard way. Because the DCPROMO process can do some of the DNS server configuration for us, this process could have been a little easier. However, DCPROMO doesn't configure all the little settings we discussed earlier. More so, the value in seeing the extended process gives you the understanding of the relationship between DNS and AD as well as how AD relies on DNS for the resolution of its necessary services. Also, by creating our secondary DNS server and later elevating it to an AD-integrated server, we get to see the differences in how both types of DNS configurations affect AD.

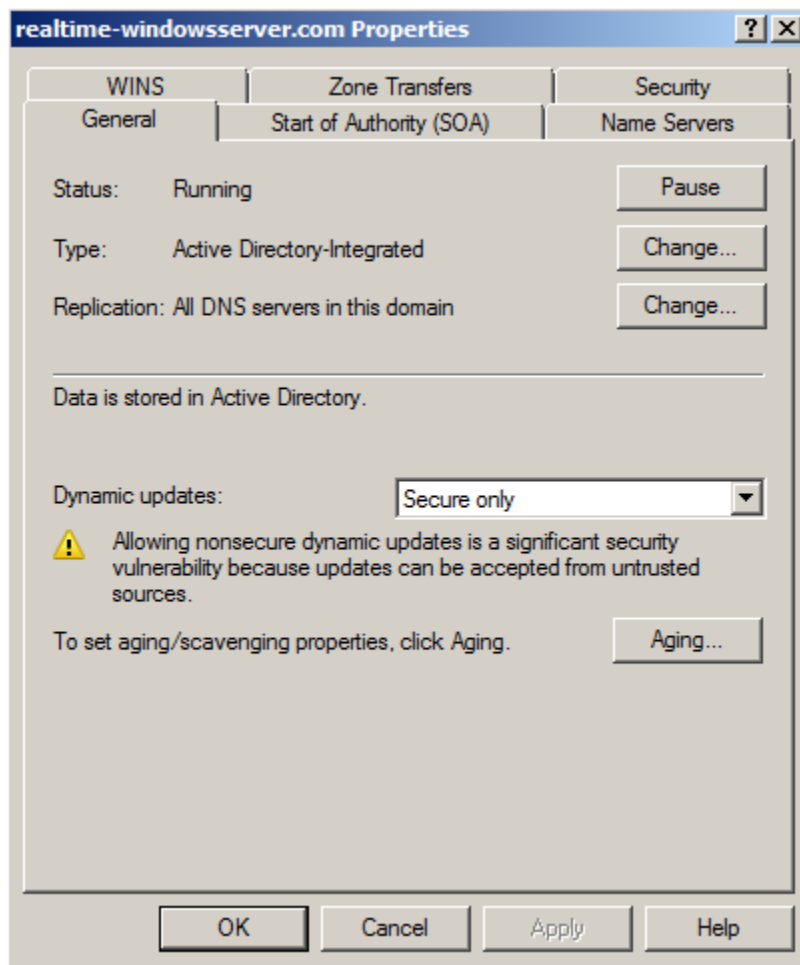



Figure 3.6: Switching the DNS zone to AD-integrated allows DNS information to replicate in the same way AD information replicates throughout the domain.

Upgrading Domain Controllers

The truth of the matter with Windows domains is that in a lot of cases you may find yourself upgrading an existing domain rather than installing a brand new one from scratch. At the very least, this will be the case with your production environments. The best way to upgrade Domain Controllers from Server 2003 to Server 2008 is really not an upgrade at all but a rolling reinstallation of the OS on each of your Domain Controllers in such a way that the AD database remains up and operational during the migration.

 Why is this wholesale reinstall process arguably the best method? Consider the average age of service of your existing Domain Controllers. If they've been around for a number of years, it's likely that they've accumulated a little extra girth around the waist. You may have installed unnecessary applications to them or uninstalled others that left pieces behind. Internet browsing may have left undesired resident code on them that you will want to clean up as part of the upgrade process. All of these bits accumulate to become a great reason to make your 2003-to-2008 upgrade a good time for a fresh installation rather than carrying forward the "extras."

In this section, we'll talk about the upgrade process to get a domain from running on Server 2003 to Server 2008. We'll assume that the desired goal for the project is to ensure that all upgraded Domain Controllers are running freshly installed copies of Server 2008.

To illustrate the process, we will be discussing a relatively simple example. This example is important as it is likely the one that resembles the majority of Windows domains. Here, we will assume that our forest is comprised of only a single domain named `realtime-windowsserver.com` with two domain controllers named DC1 and DC2. Those Domain Controllers run Server 2003 R2, are fully patched, and serve as DNS servers. The FSMO roles on these servers are all housed on DC1 and both servers are GC servers. The process we will use to upgrade to Server 2008 will include the following general steps: We'll first update the AD schema to support Server 2008. We will then add a third Server 2008 member server named DC3 into our domain and promote that server to become the third Domain Controller for the domain. If this is your production environment, consider using a virtual machine for this third Domain Controller as it will be used only temporarily during the upgrade. Once that server is added to the domain, we will transfer the FSMO roles to DC3. We can then rebuild each of our Server 2003 Domain Controllers to Server 2008, one at a time. To do so, we will first demote each server back to a member server to ensure its Domain Controller information is removed from AD.

Once DC1 and DC2 have been rebuilt as Server 2008, we will transfer the FSMO roles back to DC1 and decommission DC3. Finally, we will upgrade the Forest Functional Level to Server 2008.

Updating the Schema


The schema update process can in many ways be one of the most difficult parts of the entire process. Schema updates involve large-scale changes to the structure of the AD database. Thus, the update could break needed functionality in environments which have made customizations to the AD database in support of custom applications or complex arrangements.

That being said, the schema update process is fairly transparent. If you look on the Server 2008 media in the \sources\adprep folder, you'll see a series of files with an .LDF extension that are readable in any text editor. Opening any of these files will show you text similar to what is seen in Listing 3.1. In this listing, we see one example of a schema change.

```
dn: CN=ms-DS-isGC,CN=Schema,CN=Configuration,DC=X
changetype: ntdsSchemaAdd
objectClass: attributeSchema
ldapDisplayName: msDS-isGC
adminDisplayName: ms-DS-isGC
adminDescription: For a Directory instance (DSA), Identifies the state
of the Global Catalog on the DSA
attributeId: 1.2.840.113556.1.4.1959
attributeSyntax: 2.5.5.8
omSyntax: 1
isSingleValued: TRUE
systemOnly: FALSE
searchFlags: 0
schemaIdGuid:: M8/1HeUPnkmQ4eLLQnGKRg==
showInAdvancedViewOnly: TRUE
systemFlags: 20
```

Listing 3.1: One example of a schema change as done by the Server 2008 upgrade process.

There are hundreds of additions and changes just like what is shown in Listing 3.1 associated with the Server 2008 schema upgrade. So, for environments that have either a heavy reliance on direct schema access or have incorporated custom changes, a thorough review of the files in this folder is in order. Doing so will assist with ensuring that the update will not cause problems with the production environment.

 It's worth mentioning that a schema update has the potential for causing major problems. Because of this, a full backup of the domain and AD database should be completed prior to starting this process. An even better idea prior to starting the schema update is to power down one Domain Controller in your environment. AD backups are notoriously difficult to use for restorations. Conversely, if a Domain Controller is powered off during the upgrade process, it will not receive the updates and can later be used as the "backup" database should the upgrade fail.

The actual schema upgrade process has three steps, with one step being optional. Each of these steps is done using the `adprep.exe` tool also found on the Server 2008 media in the `\sources\adprep` folder. This is a command-line tool that is run on specific Domain Controllers in the environment:

- `Adprep /forestprep`—The first step in the schema update is to update the forest schema using this command. Running this command must be done before any Server 2008 Domain Controllers can be introduced into the forest and must be run on the forest's Schema Operations Master, which in our example is DC1. This step needs to be performed only once for the entire forest.
- `Adprep /domainprep`—This second step must be done once for each domain within the forest after the forest schema has completed. As our forest has only one domain, we need to accomplish this only once.
- `Adprep /rodcprep`—This optional step can be run after the earlier two steps in domains where Read-Only Domain Controllers will be used. Our domain will later make use of RODCs, so we'll also run this step.



For each of these commands, after running the command, ensure that a full replication has completed prior to moving to the next step. Both Domain Controllers in our example are in the same site, so replication effectively occurs immediately.

Promoting a Member Server with DCPROMO

Once the schema has been updated, we're ready to install our first Domain Controller into our domain. This process is basically the same as the process we performed earlier in the section Promoting Additional Domain Controllers, so we won't go over it again in detail. Be sure when building this third Domain Controller that it is configured to be a GC.

Be aware that the addition of a Server 2008-based Domain Controller does not change the Forest Functional Level or Domain Functional Level. The Server 2008 Domain Controller will operate at the level of the current functional levels until such time as an administrator-initiated change is made. We'll perform that activity last in this process.

Relocating FSMO Roles

Once we've incorporated a third "temporary" Domain Controller into the environment that is running on Server 2008, we can then begin the process of rebuilding our production Domain Controller's. Prior to accomplishing this, there are five FSMO roles within any Windows domain that must remain up and operational for the full functionality of the domain. We need to transfer the owner of those roles from DC1 to DC3 before starting any upgrades. Though this transfer can be done through the GUI, it is actually easier to accomplish this with a single line at the command prompt.

To transfer the FSMO Roles to the new Domain Controller, log in as an account with Enterprise Administrator privileges and enter the following, all on one command line:

```
Ntdsutil roles connections "connect to server dc3" quit "transfer
domain naming master" "transfer infrastructure master" "transfer
pdc" "transfer rid master" "transfer schema master" quit quit
```

This command line actually runs a series of steps within NTDSUTIL to connect to DC3 and individually transfer each of the roles to the new server. If you'd like to verify the success of this command, you can do so by entering this all on one command line:

```
Ntdsutil roles connections "connect to server dc3" quit "select
operation target" "list roles for connected server"
```

If the role movement was successful, you should see a result that looks similar to Listing 3.2.

```
select operation target: list roles for connected server
Server "dc2" knows about 5 roles
Schema - CN=NTDS Settings,CN=DC3,CN=Servers,CN=Default-First-Site-
Name,CN=Sites,CN=Configuration,DC=realtime-windowsserver,DC=com
Domain - CN=NTDS Settings,CN=DC3,CN=Servers,CN=Default-First-Site-
Name,CN=Sites,CN=Configuration,DC=realtime-windowsserver,DC=com
PDC - CN=NTDS Settings,CN=DC3,CN=Servers,CN=Default-First-Site-
Name,CN=Sites,CN=Configuration,DC=realtime-windowsserver,DC=com
RID - CN=NTDS Settings,CN=DC3,CN=Servers,CN=Default-First-Site-
Name,CN=Sites,CN=Configuration,DC=realtime-windowsserver,DC=com
Infrastructure - CN=NTDS Settings,CN=DC3,CN=Servers,CN=Default-First-
Site-Name,CN=Sites,CN=Configuration,DC=realtime-windowsserver,DC=com
Select operation target:
```

Listing 3.2: This result from the “list roles for connected server” command shows that all FSMO roles have been moved to the server DC3.

Demoting and Rebuilding Domain Controllers

We can now safely demote the existing Server 2003 instances to remove their Domain Controller-related information from the AD database. Once we've completed that step, we will then rebuild them each as a Server 2008 instance.

To complete the demotion, from a command prompt, run DCPROMO. Upon starting the Active Directory Installation Wizard, an error message will appear noting that *This domain is a Global Catalog server*. This error message warns us to ensure that at least one Domain Controller is also a GC, which is the case as we have ensured in this example that all Domain Controllers are also GC servers. Click OK to clear the error. Next, ensure that the box is not marked for *This server is the last domain controller in the domain*. At the following screen, enter the Administrator password to begin the demotion process.

The demotion process will remove all references for the server in the AD. After completing this process, you can safely rebuild the server as a Server 2008 instance. Upon completing the installation of Server 2008, re-run the DCPROMO process and promote the member server back to a Domain Controller.

Relocating FSMO Roles

Once DC1 and DC2 have been demoted, rebuilt to Server 2008, and subsequently promoted, we are ready to relocate our FSMO roles back to their permanent home. As before, we'll use a command line to move them all at once from their temporary storage location on DC3 to their permanent location on DC1. Do this by entering the following all on one command line:

```
Ntdsutil roles connections "connect to server dc1" quit "transfer
domain naming master" "transfer infrastructure master" "transfer
pdc" "transfer rid master" "transfer schema master" quit quit
```

The same verification as earlier can be used to ensure that the transfer occurred successfully.

Functional Levels

We now have three Domain Controllers happily operating on Windows Server 2008, though our Windows domain and forest are still running under the old functional level. To upgrade the functional level for the forest and domain and receive the benefits associated with the new functional levels, navigate to Start | Administrative Tools | Active Directory Domains and Trusts. Right-click the top-level node of the resulting console and choose Raise Forest Functional Level. Under *Select an available forest functional level*, select Windows Server 2008 in the drop-down menu and click Raise.

In our example, because our forest has only a single domain, raising the Forest Functional Level automatically also raises the Domain Functional Level. In forests with multiple domains, this will need to be a separate process for each domain. In order to do this in that circumstance, right-click the domain name in AD Domains and Trusts and select Raise Domain Functional Level. Under *Select an available domain functional level*, select Windows Server 2008 and click Raise. Your Domain Controller upgrade process is now complete.


Read-Only Domain Controllers

Server 2008 introduces a new special kind of Domain Controller called a Read Only Domain Controller (RODC). This Domain Controller is in some ways similar to the NT-style BDC in that it contains a read-only copy of the AD database. That copy can be authenticated against by any clients.

Different than BDCs, however, administrators can select which AD objects are replicated down to an individual RODCs. Objects that are replicated down to the RODC can be used for authentication. If an object is not present on the local RODC, an upstream RODC or Domain Controller can be used for authentication.

RODCs are designed specifically to be used in remote site or branch office situations in which physical security for Domain Controllers may not be assured. With full read/write Domain Controllers, it is possible for a would-be attacker with console access to a Domain Controller to access the entire contents of AD. By stealing a single Domain Controller from an unsecured branch office, the attacker would have access to all data stored within the AD. This means that the theft of a single Domain Controller could require the repermissioning of all objects within AD—an expensive and time-consuming activity.

Since objects are replicated down to an RODC only when they are identified by an Administrator, only a subset of the total AD database is present at the remote site. Should an attacker attempt to break into or steal that RODC, they will have access only to that limited subset of objects. Thus, repermissioning will only be required on those objects that were replicated to the lost RODC.

 As stated earlier in our section on the adprep.exe tool, the /rodcprep schema update must be performed on each domain prior to creating any RODCs.

Creating an RODC is nearly exactly the same as creating a regular Domain Controller. To create an RODC, follow the same steps as shown earlier using DCPROMO in Advanced Mode. The major difference is in the wizard titled Additional Domain Controller Options. Here, check the box for Read-only domain controller (RODC) as shown in Figure 3.7.

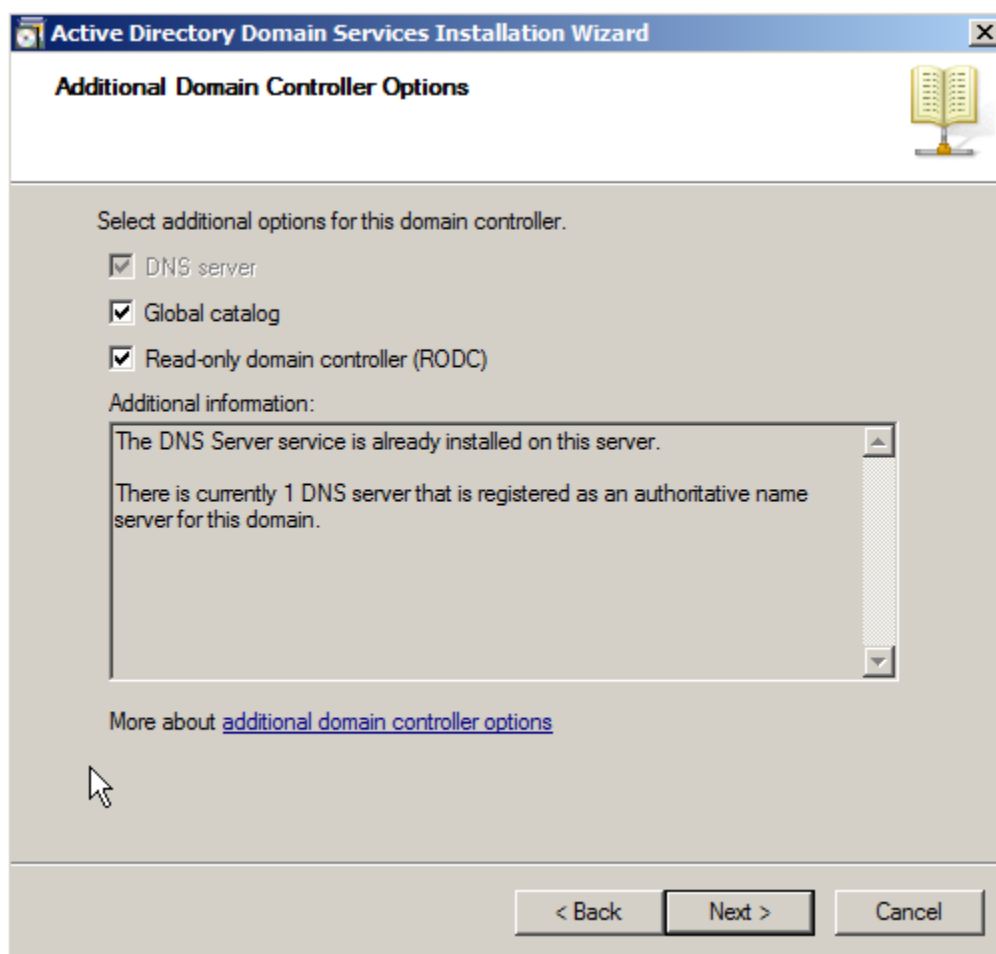


Figure 3.7: Checking the RODC box in the DCPROMO step will create the Domain Controller as an RODC rather than a regular Domain Controller.

After checking the box for RODC, the next step in the DCPROMO wizard will be to Specify the Password Replication Policy. This is the location where individual users or AD groups are identified whose members should be replicated down to the RODC. Users and groups can be configured to either allow or deny the replication of their information to the RODC. By providing this ability in both directions, highly secure accounts such as Domain Administrator can be specifically prevented from RODC replication.

The final additional step will be the Delegation of RODC Installation and Administration. Since an RODC is a tightly controlled version of a regular Domain Controller at a remote site, it is feasible for a separate individual or group to be granted administrative access to the server. As Domain Administrative privileges are typically required to manage a standard Domain Controller, this separation allows a local individual or group to handle the management and maintenance of the RODC without needing to add them to the Domain Administrators group.

Once the DCPROMO process is complete, it is possible to further access the Password Replication Policy for the RODC by navigating to Start | Administrative Tools | Active Directory Users and Computers, and then locating the RODC's computer object. Right-click the computer object and choose Properties followed by the Password Replication Policy tab to see a window similar to Figure 3.8.

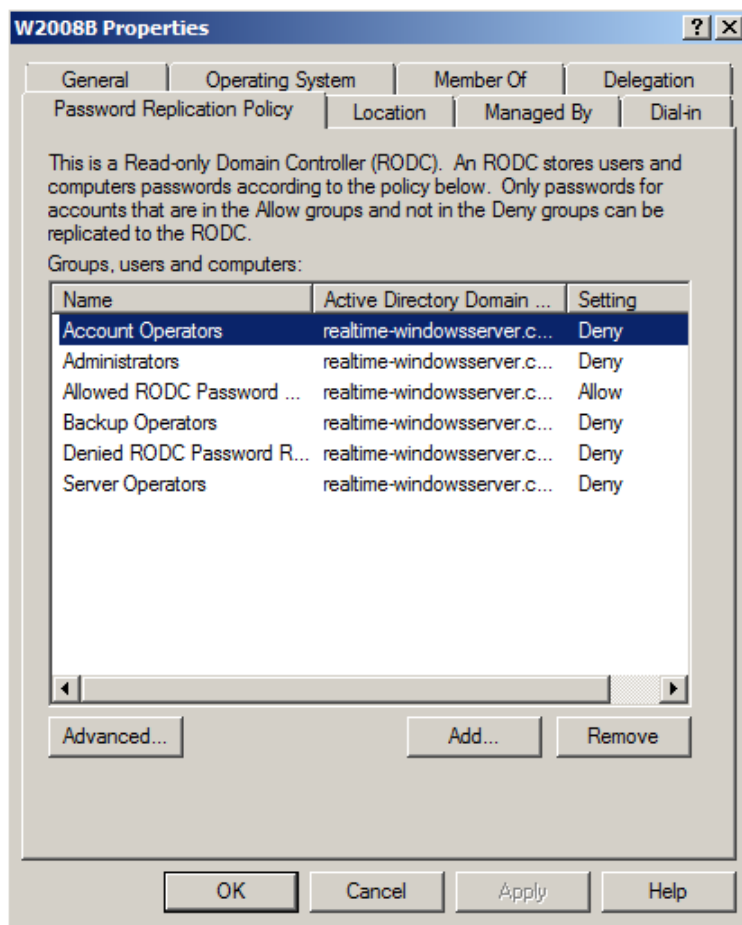



Figure 3.8: Password Replication Policy can be manipulated after the installation through the RODC's computer object in Active Directory Users and Computers.

AD Backup and Restore

Our last topic in this chapter deals with the dual concepts of AD backup and restore. Taking consistently good backups of the AD database ensures the highest chance of a successful restoration should a problem occur. The Server 2008 native tool to accomplish this task is called Windows Server Backup, much of which we've already discussed back in Chapter 2.

Backing Up the AD Database

Backing up the AD Database using Windows Server Backup is trivial. With Windows Server Backup, select the drive letter of the volume to be backed up within its wizard. If that volume includes an instance of AD, it will be backed up as well. There is no ability to separately backup just the system state as with previous versions. This will have the tendency of increasing the size of backups but will also tend to increase the certainty that a restoration will occur with success.


 If you have selected to move portions of AD's database, logs, or SYSVOL to other volumes, ensure that those volumes are backed up as well.


In this example, we'll assume that all AD components are configured to reside on the C volume. To create a backup, right-click Windows Server Backup and select Backup Once. In the resulting window, choose *Different options* to view the complete list of options. Then, select a *Full server backup*. This backup will capture all data as well as the necessary system state information that includes the AD database and other needed components.

The next screen allows you to configure the storage of the backup. Local drives are available as well as remote shared folders. Select an option with enough storage space to store the backup. For the advanced option, it is possible to choose whether the Volume Shadow Copy Service a full or a copy backup. The choice here is dependent on whether another backup product is being used to backup applications. At the final screen, click the *Confirm* button to start the backup process.

Restoring Individual AD Objects

Restoring individual objects in Server 2008 is relatively unchanged from in previous versions. The process still involves switching the server to run in Directory Services Restore Mode and marking objects from previous backups to authoritative. That process is a cumbersome process that doesn't get better with the release of Server 2008.

 You can find more information on the process to restore individual AD objects on the Microsoft Web site at <http://technet2.microsoft.com/windowsserver/en/library/690730c7-83ce-4475-b9b4-46f76c9c7c901033.msp?mfr=true>.

 One part of this process that does help with the restoration of individual objects is the use of Server 2008's DSAMAIN tool to capture a backup's snapshot of the AD. Though use of DSAMAIN is out of scope for this chapter, this tool can be used to view parallel instances of AD to find in which backup the deleted object was captured. DSAMAIN itself cannot restore the deleted object. It is only capable of showing a read-only view into the copy of the database.

Restoring Full Domain Controllers

While restoring individual objects isn't all that much easier with Server 2008, the restoration of full Domain Controllers—and in the same vein full servers—is much improved with Windows Server Backup. As we talked about in Chapter 2, Windows Server Backup and its tie into Volume Shadow Copies allows for the creation of backups that support bare metal restoration of a server.

By backing up a Domain Controller using Windows Server Backup, it is possible using the procedures we discussed in Chapter 2 to perform a bare metal restore of that server to similar hardware after a failure. Once the server is rebooted and brought back online after restoration, ADDS will recognize that it has recovered from a backup and will begin an integrity check and re-index on the AD database. Any objects that have changed since the time of the backup will be updated through normal AD replication.

Because the bare metal restoration process is so easy using Complete PC Restore, the process of returning a Domain Controller back on-line after a server failure can be completed relatively quickly.

AD Is a Central Part of Your Windows Infrastructure

In environments that use it, all of the data, applications, and people that make up a Windows environment rely heavily on AD. AD's authentication, authorization, and management functions make it a critical component of any Windows network. Thus, a good AD design is important to ensuring that users have the best possible access to their needed resources with a minimum of downtime.

In this chapter, we've talked about some of the design aspects of AD as well as the down-and-dirty steps necessary to get it freshly installed as well as upgraded from previous versions. We've discussed how AD relies on DNS as well as reviewed some examples of how that reliance can be configured.

Next up, we'll drill down our focus away from AD's all-encompassing reach to talk specifically about one role that is present in nearly all Windows environments—the venerable file server. Though the process of serving files hasn't changed much from OS version to version, the tools we have to manage it have. With Server 2008, we get a new role dedicated to the serving of files, and a suite of new tools that improve our ability to manage that data that is critical to our business.

Download Additional eBooks from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this eBook to be informative, we encourage you to download more of our industry-leading technology eBooks and video guides at Realtime Nexus. Please visit <http://nexus.realtimepublishers.com>.