1

What's New in Windows Server 2003 and ProLiant Architecture and Tools

Windows 2000 made a gigantic leap in technology from Windows NT in terms of scalability, stability, industry-standard protocols, and manageability. In the process, Windows 2000 allowed companies to simplify and consolidate their computing environments, which resulted in significant savings. Before merging with HP, Compaq eliminated hundreds of domain controllers and simplified an NT domain structure that consisted of 13 master account domains and about 1,700 resource domains to a single root domain with three child domains. In my book, *Windows 2000: Active Directory Design and Deployment*, I dedicated three chapters to identifying new features, benefits, and business justification.

Moving to Windows 2003 from Windows 2000 won't be nearly as dramatic as was moving from NT to Windows 2000, but there are some significant benefits. If you are migrating from NT to 2003, you can expect all the benefits of the migration to Windows 2000 plus the additional benefits provided by Windows Server 2003.

While we outline the significant new features and benefits of Windows Server 2003 in this chapter, we also will identify benefits throughout the remaining chapters of this book and discuss how you can apply them in the various aspects of your Windows Server 2003 design and deployment. In this chapter, we won't attempt to identify every single change or improvement made since Windows 2000. Rather, we will focus on significant features that will aid system architects or Administrators in identifying benefits to their organizations that might help solve a particular problem, aid in justifying a migration to Windows Server 2003, or provide information that will aid in the design or reconfiguration of the Active Directory (AD) enterprise.

In this chapter we focus on new features in Active Directory (AD), Networking, and Security. You can read about the new features in Exchange, Clusters, Virtual Private Networking (VPN), and Terminal Services (TS) in Chapters 12-15, respectively.

Active Directory (AD)

Some of the new features in AD fix problems with Windows 2000, whereas others add new functionality. A majority of the improvements that will make a difference in the Windows 2000 enterprise are in the area of replication and Global Catalog (GC) servers. In the sections that follow, we discuss some of the significant features and enhancements in this area. We explain some in detail, and summarize others because we detail them in subsequent chapters of this book.

Universal Group Membership Caching

Windows 2000 native mode, as well as Windows Server 2003, requires users to be authenticated by a Global Catalog (GC) server. This often causes a problem for remote sites that have no GC and must therefore authenticate across a WAN (Wide Area Network). Microsoft provided an option in the form of a Registry key that permitted authentication without contacting a GC. However, without GC contact, universal groups cannot be added to the user's token. Thus, if a deny ace was set for a universal group on a resource, and a member of that group was authenticated without a GC, that user would have access to the resource because the universal group would not be in the user's token.

In Windows Server 2003, a new feature called Universal Group Membership Caching (sometimes called GC Caching) gives users a new option for dealing with this situation. Because at the time of this writing this feature is in its infancy, it is not well documented. Thus, we've chosen to provide a fair amount of detail here as it could have tremendous benefit for enterprises that have small remote sites or sites that do not have a local GC.

Overview of Universal Group Membership Caching

Universal Group Membership Caching allows a user to log on, be authenticated by a local domain controller (DC), and retrieve global and universal group memberships from a "cache" located as attributes to the user object on the local DC—even though the DC is not a GC. Group Membership Caching is not enabled by default and is intended to provide more efficient logon performance for users at sites where a GC is not justifiable, typically at smaller sites. Group Membership Caching is intended as a stop-gap until the number of users at a site justifies installation of a GC, as caching will require use of resources on the DCs to support it. The term "cache" as used here is not defined as you might normally think of a cache—a temporary, volatile storage that is cleared on logoff or reboot. Rather, these values are stored as attributes of user objects in the AD and are purged only by an Administrator or by time-sensitive values.

You create the cache by enabling the caching feature on the *NTDS settings* object in a site. When caching is enabled and the user logs on and is authenticated by a DC in that site, three attributes are added to the user object:

- ***msDS-Cached-Membership***: This attribute contains all the universal and global groups the user is a member of as stored in the GC.

- ***msDS-Cached-Membership-Time-Stamp*:** This is the time that a user's cached membership was last updated. Certain triggers use this attribute to ensure that the group membership listed in the *msDS-Cached-Membership* attribute is up-to-date.
- ***msDS-Site-Affinity***: This attribute contains the Globally Unique Identifier (GUID) of the sites where the user has logged on and a time stamp when the attribute was last updated. This is the only attribute of the three that is replicated via AD replication.

note

Users who are authenticated by a DC that is also a GC will not have these attributes populated on that DC/GC. They will be populated on other DCs in the site that are not GCs. When you are authenticated by a GC, it always has the membership information and doesn't need to go anywhere else for the membership.

Functional Level Requirements

The Universal Group Membership Caching feature does not require any particular functional level. In fact, you can have Windows 2000 DCs in the domain and in the site where Universal Group Membership Caching is enabled. However, this can be a serious problem because Windows 2000 DCs don't have this functionality. Therefore, if a Windows 2000 DC validates the user, the group membership will be inconsistent because the Windows 2000 DC will have to always contact the GC. It is recommended that all DCs in a site where Universal Group Membership Caching is enabled are Windows Server 2003 DCs.

Facilitating Universal Group Membership Caching

Universal Group Membership Caching is a site attribute that must be enabled specifically in the properties of the NTDS Site Settings object for each site on which you want Universal Group Membership Caching functional. To enable this feature, follow these steps:

1. In the Sites and Services Snap-in, click on the site icon of the site you want to enable Universal Group Membership Caching for.
2. In the right pane of the snap-in, right-click on the NTDS Site Settings object and select Properties.
3. In the Properties window, select the box for Enable Universal Group Membership Caching, as shown in Figure 1.1.

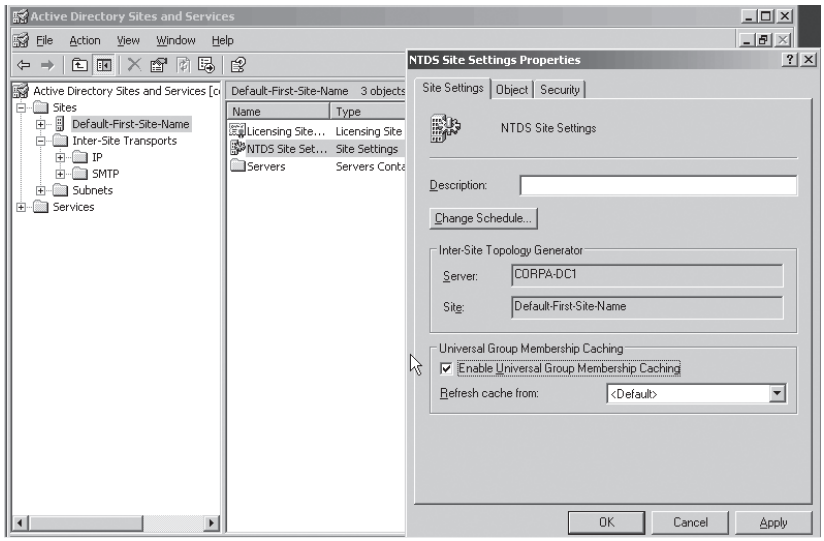

Figure 1.1 Setting the Enable Universal Group Membership Caching option in the NTDS settings object via the Sites and Services snap-in.

4. Click OK to close the Properties dialog box.
5. Repeat this process for each site that is to have Universal Group Membership Caching turned on.

note

Caching for any user happens only on the sites in which it is enabled. Thus, a user can log on and be authenticated by a DC using cached memberships at one site, then travel to another where Universal Group Membership Caching is not enabled, and be required to contact a GC to be authenticated.

Enabling Universal Group Membership Caching on Initial User Logon

After caching is enabled on a site, each user's group membership is stored on the authenticating DC in that site after the user's initial logon—after caching is enabled. The *msDS-SiteAffinity* attribute is populated with that site's GUID and is replicated to the other DCs in the site. This is accomplished in the following sequence of events. Refer to Figure 1.2 for the domain configuration in this and subsequent examples in this document.

1. Universal Group Membership Caching is enabled at the Atlanta site.

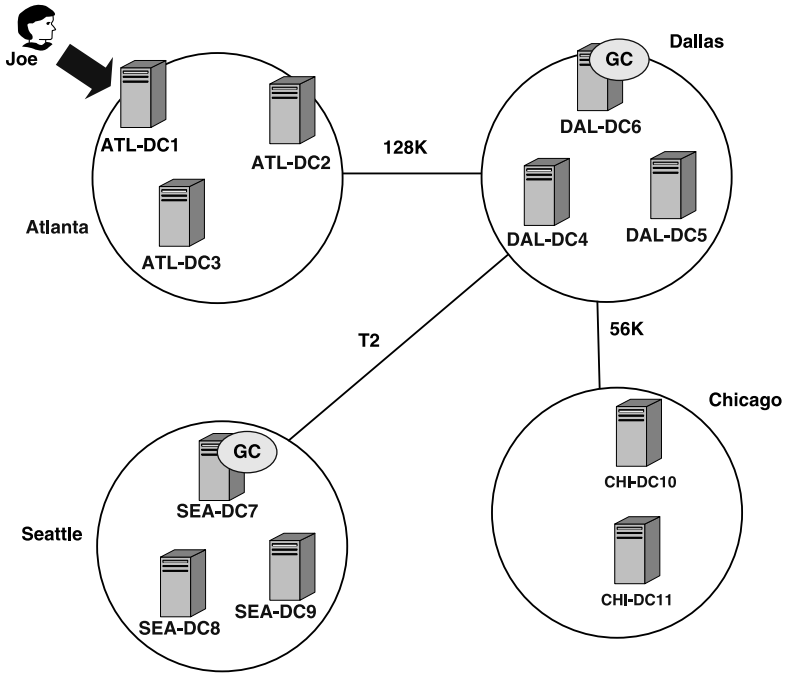

Figure 1.2 Domain configuration for scenarios used in this section.

2. User "Joe" logs on and is authenticated by DC ATL-DC1. ATL-DC1 contacts a GC server, DAL-DC6 in the Dallas site.

3. The three attributes, *msDS-Cached-Membership*, *msDS-Cached-Membership-Time-Stamp*, and *msDS-Site-Affinity*, are all populated.

   - *msDS-Cached-Membership* contains the Security Identifiers (SIDs) of all the global and universal groups that the user is a member of, obtained from the GC, DAL-DC6.
   - *msDS-Cached-Membership-Time-Stamp* contains the time that the group membership attribute was updated. (In this case, it's the time the attribute was initially populated.)
   - *msDS-Site-Affinity* contains the Atlanta site. This attribute contains GUIDs of all the sites that the user has logged on to.
   - The *msDS-Site-Affinity* attribute value is populated to the other DCs in the site.

Table 1.1 lists the attribute values populated for Joe after the initial logon (note that this is a conceptual illustration only; the group memberships are actually a binary blob). This table shows Joe's user object as viewed from all three DCs in the Atlanta site.

Table 1.1 Attributes After Initial Login with Universal Group Membership Caching Enabled and Authentication by ATL-DC1

User Object Joe viewed from DC:	*msDS-Cached-Membership*	*msDS-Cached Membership-Time-Stamp*	*msDS-Site-Affinity*
ATL-DC1	Domain users gg; ITStaff UG group; Atlanta Site admins gg (Binary blob)	06:10:2003:13:27	<Guid>
ATL-DC2			<Guid>
ATL-DC3			<Guid>

Figure 1.3 and 1.4 illustrate ADSIedit showing the cache of a user who has logged in for the first time to a site with caching enabled. Figure 1.3 shows the *msDS-Cached-Membership* and *msDS-Cached-Membership-Time-Stamp* attributes, and Figure 1.4 shows *the msDS-Site-Affinity* attribute. Note that while ADSIedit shows the hex representation, LDP simply shows this as <binary blob>, as shown in Figure 1.5.

Caching Behavior After Initial Logon in a Site

After the cache attributes are populated by the DC that authenticates Joe, and the *Site Affinity* attribute is replicated to the other DCs in the Atlanta site in the example, subsequent attempts by Joe to log on will be handled as described in the upcoming case scenarios. (Refer to Figure 1.2.) Prior to describing these three basic authentication scenarios, however, we need to define some terminology. We will present more detail on these parameters in the "Managing Cache Parameters" section.

- **Site Stickiness**: This value defines the maximum length of time a user's Group Membership Cache will be refreshed. If the time stamp is less than one-half of the stickiness setting, the attributes are refreshed. If it is between 50% and 100% of the stickiness setting (default = 180 days), the cache is refreshed when the user logs on or changes his or her password. If the time stamp is greater than the stickiness setting, the cache is cleared and the user must contact a GC to log on.

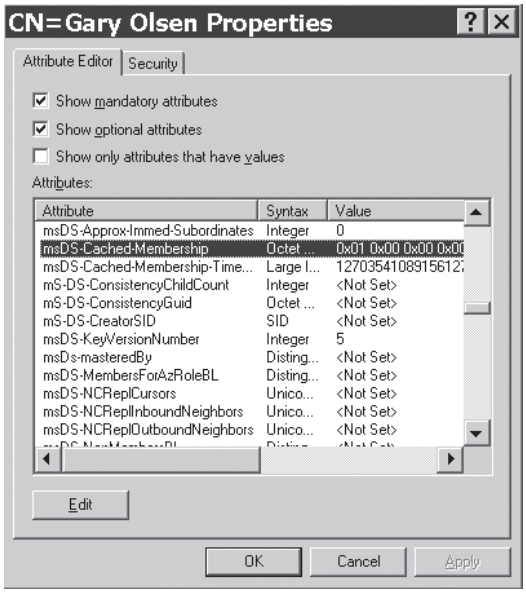

Figure 1.3 User's populated *msDS-Cached-Membership* and *msDS-Cached-Membership-Time-Stamp* attributes via the ADSIedit.msc tool.

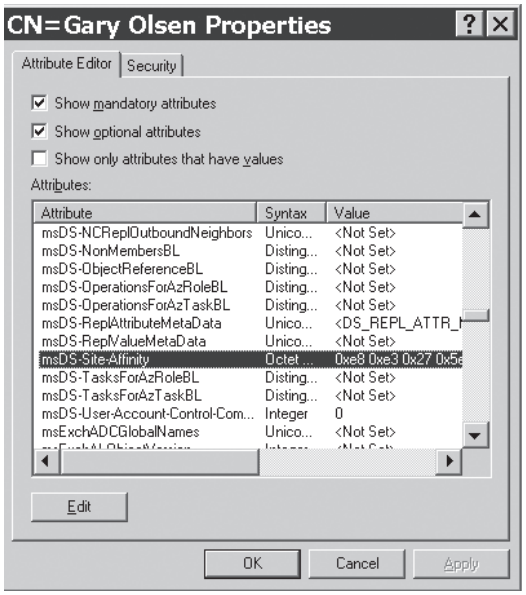

Figure 1.4 User's populated *msDS-Site-Affinity* attributes via the ADSIedit.msc tool.

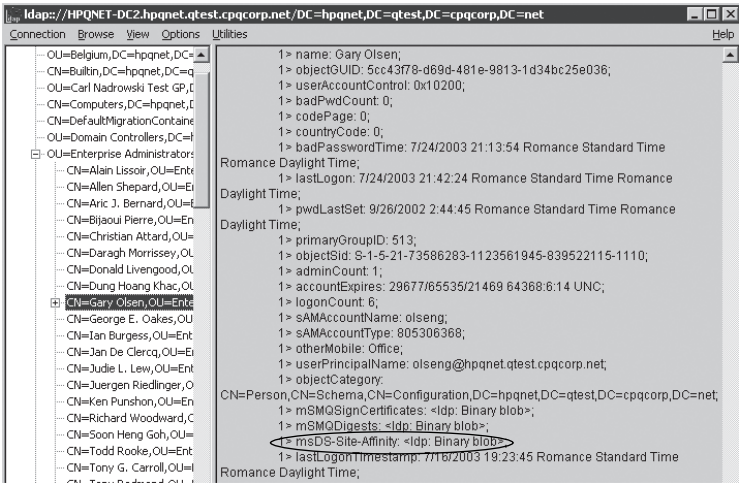

Figure 1.5 The LDP tool shows the *Site Affinity* attribute as a <binary blob>.

- **Staleness Threshold**: Specifies the lifetime of cached memberships. The Staleness Threshold is compared to the *msDS-Cached-Membership-Time-Stamp*. If the time stamp, compared with the current time when a user logon attempts to use the cached memberships for the security token, is older than the staleness value, the user is forced to contact a GC for group membership, which will repopulate the *msDS-Cached-Membership* value and reset the time stamp.
- **Refresh Interval:** This parameter specifies the time between cache refreshes. This is the period of time when a DC will issue a cache refresh to update group memberships for users in its site.
- **Refresh Limit:** This parameter defines the maximum number of users on a single DC whose cached attributes can be refreshed in each refresh cycle. The default is 500, but there is no limit. Refer to the Managing Cache Parameters section for more on this important value.
- **Falling Behind Check**: This check determines whether the refresh process is keeping up. A search is made for the oldest *msDS-Cached-Membership-Time-Stamp* value. If that time stamp is older than the staleness interval, error event ID 1670 is logged, stating that the refresh task has fallen behind.

Now that the terminology is defined, we can consider the following example scenarios.

Example 1

Joe logs in and is authenticated by ATL-DC1 again. A check is made for the Staleness Threshold. If the Staleness Threshold is not exceeded (one week by default), ATL-DC1 populates Joe's token with the membership list contained in the *msDS-Cached-Membership* attribute without contacting the GC. If the Staleness Threshold is exceeded, and cache refresh has not purged the values on the caching attributes, the DC contacts a GC for group membership in the authentication process.

Example 2

Joe logs in and is authenticated by ATL-DC2 or ATL-DC3. Joe's *msDS-Cached-Membership* attribute and associated time stamp are populated during normal cache refresh by the DCs. On cache refresh, the DC examines the site GUIDs stored in Joe's *msDS-Site-Affinity* attribute. If one of those is the GUID of the site the DC is in, the DC contacts the GC (DAL-GC6) and populates the *msDS-Cached-Membership* attribute. If this refresh is completed before the user is authenticated by ATL-DC2 or DC3 and the user logs in before expiration of the time stamp, the user gets his group membership from the DC without contacting the GC. Note that the *msDS-Cached-Membership-Time-Stamp* is set when the *msDS-Cached-Membership* attribute is populated initially or during a refresh, so this could be different on each DC.

Figure 1.6 shows the ADSIedit view of the state of a user's cached attributes on a DC that did not authenticate the user. In this case, the DC has had the *Site Affinity* attribute replicated, but the cache refresh has not taken place—thus the *msDS-Cached-Membership* and *msDS-Cached-Membership-Time-Stamp* are <Not Set>. Figure 1.7 shows that the *msDS-Site-Affinity* attribute is set (GUID of the site). Figure 1.8 shows the LDP view of this same situation. Note that LDP shows only the attributes that are populated, so only *msDS-Site-Affinity* is set, and the other two don't show up at all.

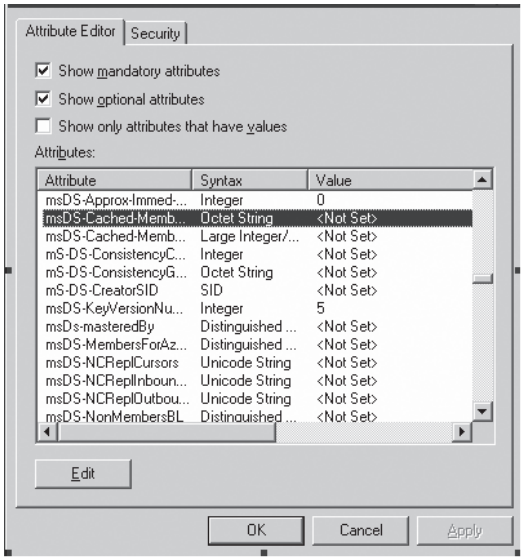

Figure 1.6 View of DC where user was not authenticated, showing *msDS-Cached-Membership* and *msDS-Cached-Membership-Time-Stamp* attributes not populated. Cache refresh has not happened.

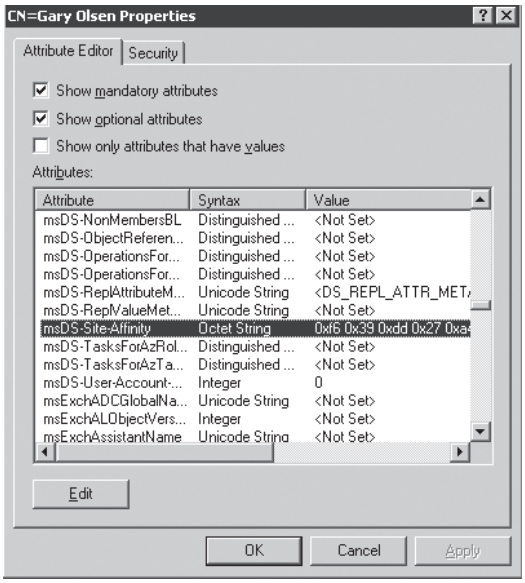

Figure 1.7 View of DC where user was not authenticated, showing *msDS-Site-Affinity* attribute populated. Cache refresh has not happened.

Figure 1.8 Using LDP.exe to view DC where user was not authenticated, showing *msDS-Cached-Membership* and *msDS-Cached-Membership-Time-Stamp* attributes not populated and the *msDS-Site-Affinity* attribute populated. Cache refresh has not happened.

Example 3

Cache refresh occurs when the other DCs in the site identify that the user's *msDS-Site-Affinity* attribute contains their site's GUID and populates *the msDS-Cached-Membership* attribute. After cache refresh is completed, the attribute values for Joe would be as shown in Table 1.2.

Table 1.2 User Joe's Attribute Values After Cache Refresh

Joe's User Object Viewed from DC:	*MsDS-Cached-Membership*	*MsDS-Cached-Membership-Time-Stamp*	*MsDS-Site-Affinity*
ATL-DC1	Domain users gg; ITStaff UG group; Atlanta Site admins gg	06:10:2003:13:27	ATL <Guid>
ATL-DC2	Domain users gg; ITStaff UG group; Atlanta Site admins gg	06:10:2003:14:23	ATL <Guid>
ATL-DC3	Domain users gg; ITStaff UG group; Atlanta Site admins gg	06:10:2003:14:44	ATL <Guid>

Logging into Remote Sites

Again referring to Figure 1.2, suppose Joe is a mobile user and he travels to the Chicago site to work for two weeks. When he logs on and is authenticated by CHI-DC10, the DC determines that Chicago's site is not in Joe's *msDS-Site-Affinity* attribute, so it contacts a GC to get Joe's membership and populates the cache attributes, including adding a new time stamp and adding Chicago's site GUID to the *msDS-Site-Affinity* attribute. On the next refresh, CHI-DC11 will populate Joe's attributes as well.

In this configuration, sites Dallas and Seattle are core sites with many users, good bandwidth, and GC servers at each site. Therefore, Universal Group Membership Caching is not enabled at those sites. If Joe travels to Seattle and logs in, the authenticating DC contacts a GC for Joe's group membership as it normally would.

Time Passes . . .

Joe completes his work in Chicago and returns to his home office in Atlanta. After a period of time, if Joe does not log in or change his password from Chicago, his attributes are purged from the Chicago DCs, including the site affinity. We will discuss these timeouts in the upcoming "Refreshing the Cache" section. If Joe goes back to Chicago after this time, his attributes will be refreshed as they were the first time he logged in. This guarantees that Joe's group membership is up-to-date.

warning

After the group membership attribute is populated, the DC will *not* contact a GC again for the user's membership until the next cache refresh. It will always use the cached membership in the *msDS-Cached-Membership* attribute. This can be a problem, especially in troubleshooting. If the Administrator changes the user's group membership (that is, he adds Joe to a new universal group) and has Joe log on, the new group might not be part of Joe's token. The administrator can force a cache refresh to update the membership, as described in the "Refreshing the Cache" section of this chapter.

Refreshing the Cache

Refreshing the cache is critical. If the cache isn't refreshed in a timely fashion, the user will not have the correct group membership because the security token is built from the cached attribute.

The refresh process is not trivial, and the DC performs a number of actions to keep the cache up-to-date. This refresh is completed independently every eight hours, or on reboot, on each DC for all users.

Stale cached attributes are eventually purged. For example, user Joe in our example traveled to Chicago from his home site in Atlanta. While he is in Chicago, his site affinity time stamp will be refreshed regularly while he is logged on. After he leaves, assuming the site stickiness setting is left at the default, Joe's cached attributes will be refreshed for another 90 days, even if he doesn't log on. When Joe returns to Atlanta, his cache for Chicago will be refreshed for another 90 days. Let's say Joe returns to Chicago and logs in 120 days after he left. Because he is beyond one-half the stickiness setting and less than the total stickiness setting, the site affinity time stamp will be refreshed when he logs on. If Joe returns after 180 days since leaving Chicago, he will have to contact a GC to get his cache populated because it will have been purged. These checks are to ensure that mobile users who go from site to site are not logging on with stale membership caches.

note

If the *msDS-Cached-Membership* values are purged, the worst thing that can happen is that the user will have to contact a GC when he or she logs in again, and if one is not available, cached credentials will be used. If cached credentials fail, login will be denied.

Managing the Cache

The cache update mechanisms and checks introduced in the previous sections are somewhat confusing and hard to keep track of, but they are configurable. In this section, we will explain how to configure the parameters and manage the cache.

Setting the Registry

You control cache refresh parameters through Registry settings. It is important that you understand what you are doing before changing these values as they can have a negative impact on WAN traffic and user logon performance.

- **Cached Membership Site Stickiness:** This parameter is configurable through the Registry at:

```
HKLM\CurrentControlSet\Services\NTDS\Parameters
Value: "Cached Membership Site Stickiness (minutes)"
Data Type: REG_DWORD
Default: 180 Days
```

- **Staleness Threshold:** This value is configurable through the following Registry parameter:

```
HKLM\CurrentControlSet\Services\NTDS\Parameters
Value: "Cached Membership Staleness (minutes)"
DataType: REG_DWORD
Default: 1 week
```

- **Cached Membership Refresh Interval:** This is manually configurable through the following Registry parameter:

```
HKLM\CurrentControlSet\Services\NTDS\Parameters
Value "Cached Membership Refresh Interval (minutes)
DataType: REG_DWORD
Default: 8 hours
```

- **Cached Membership Refresh Limit:** This parameter is configurable through the Registry at:

```
HKLM\CurrentControlSet\Services\NTDS\Parameters
Value "Cached Membership Refresh Limit"
DataType: REG_DWORD
Default: 500
```

- **Diagnostic Logging:** This parameter is similar to other NTDS Diagnostic logging parameters such as GC, Replication Events, Name Resolution, and so on, which were available in Windows 2000. The value's default is 0, meaning minimal logging. The range of values is 0–5, with 5 being very verbose. Microsoft recommends setting this value to 5 for troubleshooting. It will cause more verbose events to be recorded in the Application log. It is set in the Registry at:

```
HKLM\CurrentControlSet\Services\NTDS\Parameters
Value: 20 Group Caching
DataType: REG_DWORD
ValueData: 0-5
Default: 0
```

If you change these parameters from the default, you should be aware of the following:

- The longer the user's cache remains active, the more resources will be required by each DC in the site for refreshing the cache.
- Reducing timeouts for the cache forces the user to be authenticated by a GC more frequently.

The most common administrative chore associated with Universal Group Membership Caching is forcing a cache refresh to clean things up and start over. You can accomplish this task in four ways: purging the cached attributes for a single user, refreshing the cache for all users on a single DC, and deleting all cached attributes for all users at all DCs in a site, forcing a refresh for the entire site.

Deleting the Cache for a Single User

The Administrator might want to do this if a user's group membership has changed and it is desirable for the effects to take place immediately rather than waiting for the cache to refresh. The Administrator can use ADSIedit to accomplish this, as outlined in the following steps:

1. Open ADSIedit from Windows 2003 Support Tools.
2. Expand the domain container, expand the Users folder, and drill down to the user object to be modified (for example, "Joe").
3. Right-click on the user object and select Properties. Browse the list of attributes to find the *msDS-Cached-Membership* attribute. In the Edit Value field, delete the existing value, or change it to zero (0).
4. Repeat step 3 for the *msDS-Cached-Membership-Time-Stamp* attribute.

The next time the user logs in, because these attributes are not populated, the user is forced to contact the GC to get group membership, which will in turn populate the user's cache with the new group membership and a new time stamp.

Refreshing the Cache on a Single DC

If some users' group memberships are out of date, it could be because their authenticating DC cache refresh failed. The Administrator can force a cache reset on a single DC by setting the *updateCachedMemberships* value to 1 on the rootDSE. The Administrator can do this using the LDP tool, as outlined in the following steps:

1. Open the LDP.exe tool from the Windows Support Tools for Windows Server 2003. Connect to the server and bind as an Enterprise Admin, or a user that has the Refresh Group Cache for Logons right on the NTDS Settings Object for the DC.
2. In the LDP tool, go to Browse – Modify and enter **updateCachedMemberships** in the Attribute field and **1** in the Value field, and then press Enter. The result should look like the example in Figure 1.9.

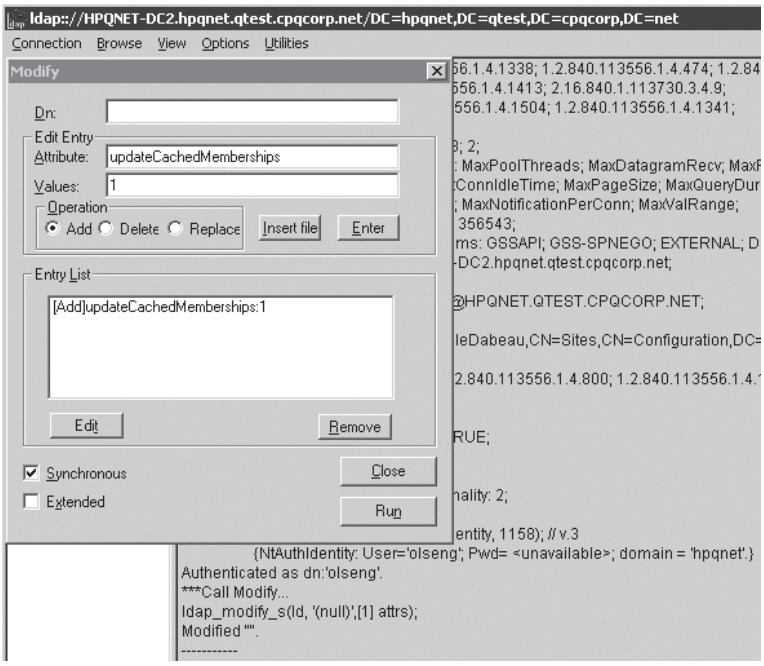

Figure 1.9 Setting the *updateCachedMemberships* attribute.

3. Select the Run button and the right-hand pane of the LDP tool should report "Modified" (also displayed in Figure 1.9).

note

Setting the *updateCachedMemberships* attribute triggers the DC to refresh the cache without rebooting the server.

Another way to refresh the cache on a single DC is to reboot it. This triggers the refresh to occur about 15 minutes after reboot.

Deleting the Cache for an Entire Site

You can purge the Group Membership Cache for all users in a site by setting the Site Stickiness Setting to zero (0), as noted previously in the "Setting the Registry" section. Doing this on one DC in the site purges all the cached attributes on that DC, including the *msDS-Site-Affinity* attribute. The site affinity attribute reset is then replicated to all other DCs in the site. At the next cache refresh, the DCs determine that they don't have the site affinity attribute, and reset the *msDS-Cached-Membership* and the *msDS-Cached-Membership-Time-Stamp* attributes for all users.

Conclusion

At the time of this writing, the Universal Group Membership Caching feature is so new that there is really no data in terms of recommendations based on experience. However, Universal Group Membership Caching has some negative impact on users and on the domain. Because the cache is a static value that is refreshed or updated periodically, this can cause delays in the user being affected by group membership changes. Therefore, if you're thinking about implementing this feature at a site, take note of the following:

- The location of the nearest GC (is user performance acceptable?)
- The bandwidth of the inbound and outbound links to the site
- The number of users at the site (more users puts a heavier load on the DC)
- Whether an Administrator is available to maintain this (this is difficult to anticipate, but be aware that there will be some administrative overhead)
- Any increased hardware cost associated with upgrading to a GC

If there are hundreds of users at a site, it might be prudent to locate a GC at that site, even if it is connected to the rest of the network via slow links. HP designed its network so that GCs are placed more frequently in sites with slow WAN links, thus shielding the users from those links. If there are only a few users in a small sales office, the use of Universal Group Membership Caching will give the users the performance of a local GC without the overhead and expense of having one at that site. The difference, of course, is the membership update latency.

Global Catalog (GC) Improvements

In addition to the Universal Group Membership Caching feature, several other significant improvements were made to GC server functionality, including improved replication performance of the partial attribute set (PAS), improvements in GC demotion performance, and improvements in the way GCs advertise themselves. Perhaps the biggest single improvement, not only in the arena of GC performance, but in all of Active Directory, is the addition of the Install from Media (IFM) feature.

Partial Attribute Set (PAS) Replication

GCs provide quick access to commonly used objects and attributes for any domain in the forest, reducing WAN traffic and improving search performance from the user's perspective. By definition, GCs contain all the objects of all domains in the forest, all the attributes of the objects in the domain for which the GC is a DC, and some of the attributes of the objects in the other domains in the forest. The objects and attributes for the other domains are stored in a read-only context and comprise the PAS, which is defined in the schema. Attributes in the PAS have the property *isMemberOfPartialAttributeSet* set to TRUE, which causes them to be replicated to the GC server.

If an Administrator desires to add additional attributes to the PAS so that searches on certain attributes occur more quickly, he or she could edit the appropriate object via the Active Directory Schema Manager, and change the attribute from the Optional list in the object properties to the Mandatory list. This action sets *isMemberOfPartialAttributeSet* to TRUE.

For instance, on the PrintQueue object, *printColor* is an optional attribute. An Administrator who is a member of the Schema Admins group desires to add *printColor* as a mandatory attribute so that attribute will be replicated to all GCs. Thus, a user visiting a site from another domain can locate a color printer faster because the local GC has the *printColor* attribute associated with the printers.

To set this, the Administrator would open the Active Directory Schema Manager snap-in, right-click on the printQueue object on the left pane, go to Properties, and add the *printColor* attribute from the Optional list to the Mandatory list, as illustrated in Figure 1.10.

In Windows 2000, an operation as simple as this would cause all GCs in the forest to perform a full synchronization of the read-only directory partitions, causing a bandwidth spike for each GC roughly equivalent to promoting a DC to a GC, and causing a temporary disruption of service.

Windows Server 2003 changes this behavior by replicating only the changed attributes. This is a significant improvement in performance, making the attribute change almost insignificant in most cases.

GC Partition Occupancy

In Exchange 2000 and Windows 2000, there was an issue with Exchange failures caused during a GC promotion. Exchange 2000 and later versions rely on the GC for the Global Address List (GAL). When a GC is promoted, it does not require a reboot at the end of the process. When it is rebooted, the GC advertises itself as a GC and is used by Exchange for directory lookups. If it is rebooted before the read-only partitions are fully replicated, some MAPI clients might use the GC for the GAL, causing possible mail failure for those clients.

Windows 2000 SP3 includes a workaround that allowed the Administrator to add the *Global Catalog Partition Occupancy* value to 6. This is located in the Registry at `HKLM\system\currentcontrolset\services\ntds\parameters`. This parameter does not allow the new GC to advertise itself until replication is complete for all naming contexts (NCs). (See Microsoft KB 304403 "Exchange Considerations for Promoting a Domain Controller to a Global Catalog Server" for more details.)

Windows Server 2003 implements a new value in this same Registry location to set this behavior by default. The value is *Global Catalog Promotion Complete,* and the data for this value is set to 1 by default. This prevents the GC from advertising itself until the replication of all NCs to the GC is complete. At that point, DSAccess adds that GC to its GCList for use by Exchange clients.

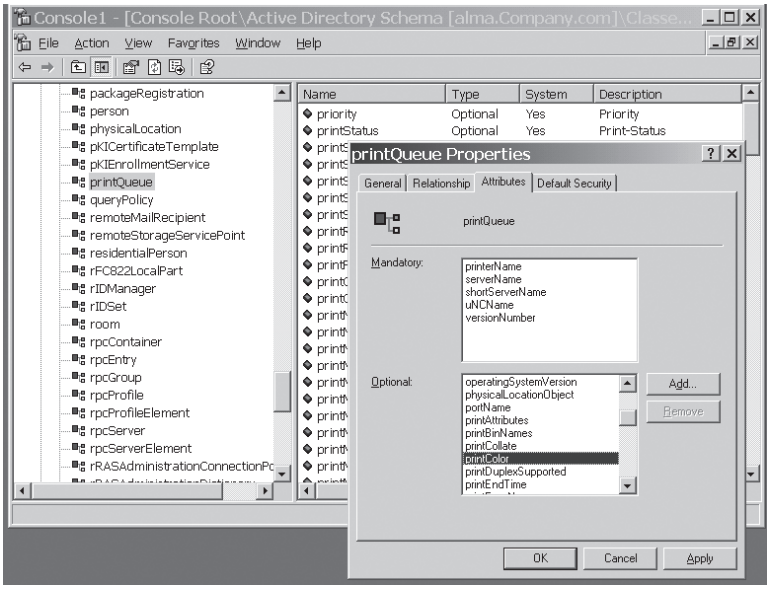

Figure 1.10 Changing an optional attribute to a mandatory attribute in the Active Directory Schema Manager.

Install from Media (IFM)

The Install from Media (IFM) feature is perhaps the single most significant improvement in the Windows Server 2003 GC features, and one of the top improvements in all of Active Directory. You enable this feature in DCPromo to permit replication from a restored backup of a DC or GC as the source, rather than replicating from a live DC or GC over the WAN. Thus, you can back up the system state of an existing DC/GC and then restore it to media that will be local to a server that will be promoted to become the new DC/GC. The media can be tape, CD, DVD, hard disk or other media that will be local to the new server (of course, the media must have the capacity to store the restored system state files). DCPromo can then be executed with the `/ADV` switch from a command line:

```
DCPromo /ADV
```

DCPromo produces an additional dialog box, shown in Figure 1.11, with the option to specify a path to the restored backup media. DCPromo will then use this media as the source to replicate the AD without touching a source DC. At the end of DCPromo, a connection is made to a live DC to source changes that occurred since the media was created. The "DCPromo" section later in this chapter details how the IFM feature works, and discusses the use of unattended answer files.

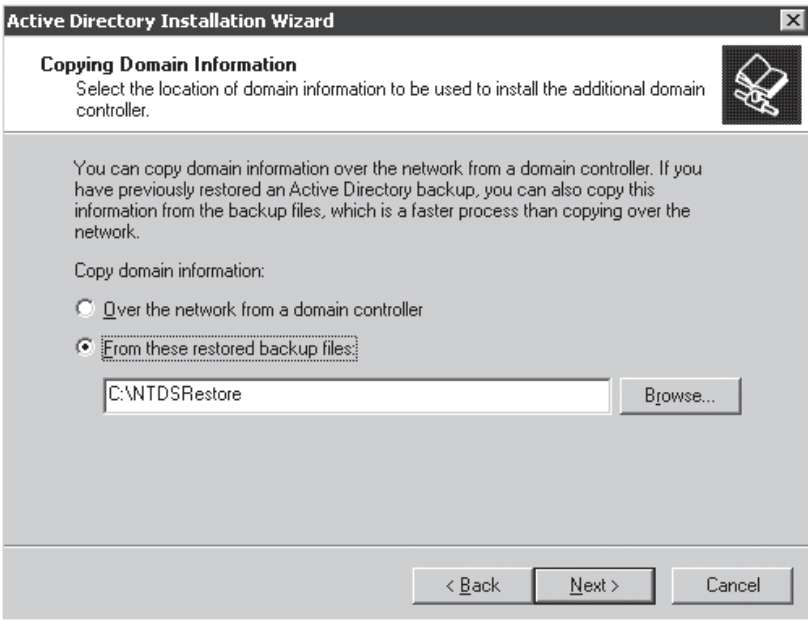

Figure 1.11 The new DCPromo dialog box permits Administrators to specify a path to restored backup files to be used as a source for replicating AD, rather than replicating over the network.

Although it is highly beneficial to build DCs using this feature, it's even more beneficial to build GC servers in this way because of the size of the GC compared to a DC.

A good example is Hewlett-Packard's AD deployment. The system state backup of a GC in HP's environment is about 10 to 12GB, but can be defragmented to about 7.5GB. In Windows 2000, HP experienced a number of GC-related problems that required GCs to be rebuilt. Because of the size of the AD and available bandwidth on the network, rebuilding a GC took from three to five days, depending on the location. This seriously impacted the users because the GC is used for Exchange's GAL, and loss of a GC at a site required the users to find another GC for GAL operations, impacting performance and putting an additional load on other GCs. In some cases, HP actually had backup GCs in some sites to mitigate the downtime required if a GC had to be rebuilt.

Initial testing of Windows 2003's IFM feature proved that a GC could be rebuilt in about 20 minutes from media. HP viewed this as a critical feature to making the AD environment more resilient and significantly reducing downtime. IFM was a key reason why HP migrated to Windows Server 2003. In fact, HP had migrated the Americas domain to Windows Server 2003 native in November 2002, shortly after the release of Release Candidate 3 of Windows Server 2003. Thus, HP was running its production environment on beta software. That speaks volumes for the importance of IFM as well as for the stability of Windows Server 2003. Today, in actual practice, HP can rebuild a GC using IFM in about 20 minutes.

Removing GC Role

One of the contributing factors in the lengthy process of rebuilding a GC in Windows 2000 was the time required to demote the GC. The operation is quite simple. In the Sites and Services snap-in, you right-click on the NTDS Settings object of a DC and select Properties. On the Properties page, there is an option for Global Catalog. If this box is checked, clearing it initiates a demotion process that removes the read-only partitions from that DC.

In Windows 2000, the GC removal process was limited to about 500 objects every time the Knowledge Consistency Checker (KCC) ran. By default, this was every 15 minutes. Thus, an AD with 4,000 users, 6,000 computers, and 500 groups (total of 10,500 objects) would require 21 iterations of the KCC to clean up one GC and make it a DC. Using the default 15-minute interval, this would require 5 hours and 15 minutes to complete this change.

Windows Server 2003 changed the way this demotion is handled. Instead of replicating a certain number of objects per each KCC cycle, the operation continues removing objects until all objects are removed. Replicating these objects is a low priority among replication tasks, so if another replication request comes in, it takes priority. Thus, if you remove the GC role during low utilization periods, the process continues, possibly uninterrupted. Otherwise, it uses available bandwidth until the job is complete. This results in a more efficient way to remove the GC. Combined with the IFM feature, removing a GC role and adding the role to another DC is much faster and more efficient.

Domain Controller Rename

While supporting Windows 2000 Administrators during the past several years, I ran into a number of situations in which the name of the DC had to be changed. The problem with Windows 2000, of course, is that the only way to change the computer name of a DC is to demote it, change the name, and then repromote it. This is a complex process to perform a simple task. Of course, the answer in NT was worse—you had to reinstall the DC.

Windows Server 2003 provides a very viable feature called Domain Controller Rename (not to be confused with Domain Rename). This functionality requires the domain to be in Windows Server 2003 domain functional level and can be performed either via the GUI (Graphical User Interface) or by using the Netdom option. You'll learn both methods in the following sections. The Netdom method gives you more options, such as renaming the NetBIOS or the DNS (Domain Name System) name, whereas the GUI method renames both.

note

> The Netdom method of renaming a DC requires the domain that the DC is a member of to be in Windows Server 2003 functional mode (all DCs are Windows Server 2003), whereas the GUI method works in Windows 2000 Native or Windows Server 2003 functional levels but only on Windows Server 2003 machines.

Using the GUI Method

This method is fairly simple. Use the same process you would use to rename any Windows 2000 Professional, Windows 2000 Server, Windows Server 2003 member server, or XP workstation. Right-click on My Computer, select Properties, and select the Change button in the Computer Name tab. A pop-up message appears, as shown in Figure 1.12. Just click OK and the familiar Computer Name Changes dialog box appears that allows you to change the name, as shown in Figure 1.13. You are prompted for credentials and, if all goes well, you are notified of the need for a reboot. As you can see, this process is pretty much the same as the process for renaming any other computer. This process modifies Registry values and cleans up DNS (mostly). If you want to see what is going on under the hood, or if you want to change the NetBIOS or the DNS name (but not both) read the "Using the Netdom Method" section coming up next.

Using the Netdom Method

Windows Server 2003 added a couple of new options to the Netdom command-line utility: Computername and RenameComputer. The RenameComputer option isn't made to work for a DC. Although it does seem to work without error, it doesn't do all the things it needs to for a DC rename, so don't use it. Using the Netdom Computername command requires several steps, and the process gives you a good view of what is going on under the hood. The steps to rename a DC with the existing name of DC1 to ATL-DC1 are as follows:

note

Before proceeding, you must set the domain functional mode to Windows Server 2003 (all DCs in the domain must be Windows Server 2003).

1. From a command prompt, execute the following command:

   ```
   Netdom Computername <currentcomputername> /add:<newcomputername>
   ```

2. Where `<CurrentComputername>` is the current name for the DC (the simple NetBIOS name), and `<newcomputername>` is the new name it is to be changed to, in Fully Qualified Domain Name (FQDN) format.

3. The Netdom command for the example is

   ```
   Netdom Computername DC1 /add: ATL-DC1.company.com
   ```

4. Note that `<existingcomputername>` is always the current name of the computer.

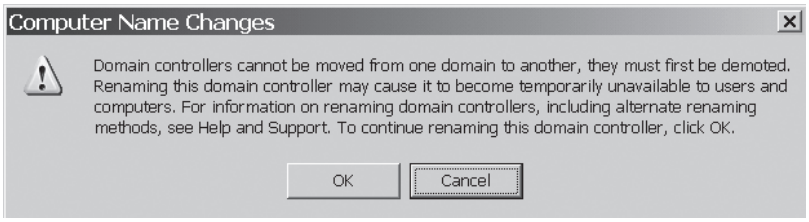

Figure 1.12 Warning message when attempting to rename a DC from within the UI (user interface).

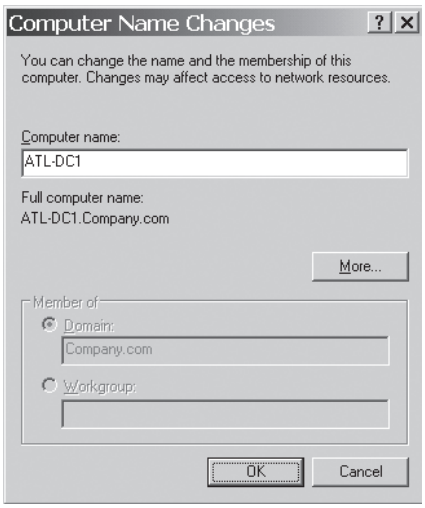

Figure 1.13 Familiar dialog box to rename a computer—now available to rename a DC.

warning

The Netdom option called computername should not be confused with an actual computername. You should enter the word **Computername** after the Netdom command, followed by the existing computername, as in this example.

5. Verify the success of the addition of the new name with the Netdom `/enumerate` command. For this example, the following command would be used. Here we see that DC1.corp.net and ATL-DC1.corp.net are listed—the old name and the new name we just defined.

```
C:> Netdom Computername DC1 /enumerate
  DC1.company.com
  ATL-DC1.company.com
```

6. Now define the new computername as the primary with the Netdom `/makeprimary` command:

```
Netdom Computername DC1 /makeprimary:ATL-DC1.company.com
```

warning

Remember that the `<existingComputername>` is always the NetBIOS version of the existing computername, and the `<newComputername>` used for arguments such as `/makeprimary`, `/add`, `/enumerate`, and `/remove` is always the FQDN version of the name—just as in the examples here.

7. At this point, the `/enumerate` command will not work correctly until the computer is rebooted and the new computername is recognized. Verify the results by executing the following command:

```
Netdom Computername DC1 /enumerate
  DC1.company.com
  DC1.company.com
```

8. You will see the old name, DC1.Corp.net, listed twice, unlike the previous `/enumerate` command, in which the new and old name were listed. This is expected behavior and will be corrected after rebooting the computer, but don't reboot yet.

9. Just for observation, open Regedt32, go to `HKLM\System\CCS\Services\TCPIP\Parameters`, and observe the *NV HostName* value. *NV HostName* should now reflect the new name, ATL-DC1.

10. Reboot the computer and log in as a Domain Administrator.

11. Open a command window and use Netdom to enumerate the computernames. Note that we have used the new name of the DC, ATL-DC1 as the first computername parameter:

    ```
    Netdom computername ATL-DC1 /enumerate
    ```

    The results show DC1 and ATL-DC1 listed, with ATL-DC1 listed as the primary name.

12. Delete the old computername, DC1, by executing the following command:

    ```
    Netdom computername ATL-DC1 /remove:DC1.company.com
    ```

13. Check to make sure the new name is recognized:

    - From a command prompt, issue the command `Hostname` to return the new name.
    - Go to My computer, select Properties, and then click the Computer Name tab. The new name should be displayed.
    - Open the DNS management snap-in on a DNS server and walk through the SRV records. There should be no records with the old name—only records with the new name.

DNS Caveat

Although both methods of DC rename do a good job of cleaning up DNS, they both fail in changing the name of delegation records with the old computername. This can be a big problem if you let DCPromo configure DNS on the forest root DC (the first one in the forest) because it automatically delegates the _msdcs zone to that first DC. If you have child domains and use this delegation, a DC from each domain will have a delegation record to this zone. If you rename any of these DCs, the delegation record will not be updated (as of the RTM release of Windows Server 2003). This is demonstrated in Figure 1.14. This is a screen shot of the DNS snap-in of a DC, previously named ALMA.Company.com, renamed to ATL-DC1.company.com. Note that the _msdcs zone is delegated, and there is one delegation record to ALMA.Company.com.

Left in this state, all queries for Cname records and GC records will fail because the referral will go to a nonexistent server. You can easily change this by right-clicking on the record in the right pane, selecting Properties, selecting EDIT, and then entering the correct name and the IP address. Do this for each delegation record that points to a renamed DC.

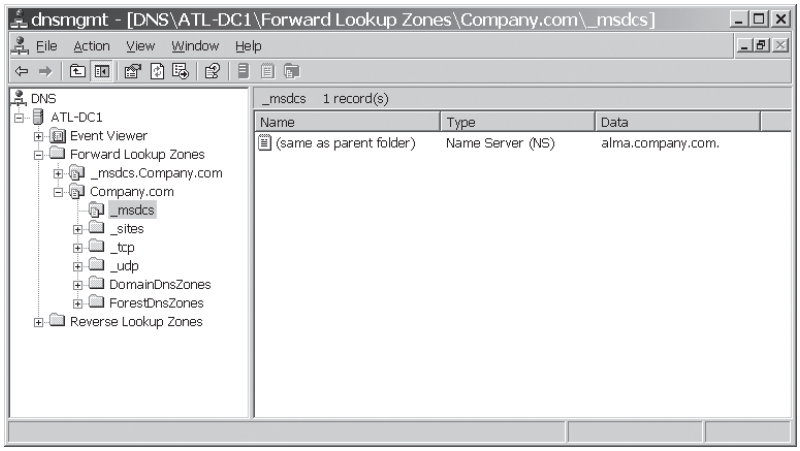

Figure 1.14 DCRename function does not clean up DNS delegation records for the renamed DC.

Application Partitions

One of the criticisms of AD has been that its scalability is limited by the fact that you have only three naming contexts: configuration, schema, and domain. Windows 2003 introduced application partitions. An application partition is a user-defined naming context that can be hosted by any set of DCs from any domain. Thus, there is no domain boundary in such a partition. You create the partition in the NTDSUtil tool. When you create such a partition, you also create a DNS forward lookup zone and place SRV records of the DCs that host the partition.

Think of a partition as a truck and data, such as the zone information, as cargo on the truck. When you replicate the partition, the zone information goes with it. You can add other data to the partition and replicate it to the DCs in the replica set. Application partitions have the following characteristics:

- User-created and user-managed.
- Can contain DCs across domains, but within a single forest.
- Cannot contain security principals.
- Can be queried using Lightweight Directory Access Protocol (LDAP).
- Can be replicated using normal AD replication.
- Can be located using DNS SRV records.

- Impacts the site topology in terms of replication traffic. These partitions replicate in addition to the configuration, schema, and any domain partitions defined by default.
- Observes site topology and schedule.
- Can't include GCs as replicas.
- Can be created directly by applications.
- Hosted only by Windows Server 2003 DCs.

How Application Partitions Work

Figure 1.15 shows how an application partition, Payroll.company.com, was created in the Company.com forest. The forest also contains EU.company.com and NA.company.com child domains. The application partition contains a DC from each of the three domains. Thus, DC1, DC3, and DC5 host the Payroll NC as well as the schema, configuration, and respective domain NCs. This partition is represented in the figure as a triangle, which usually denotes a domain. This is appropriate because this partition is a NC just like a domain is, although with limited capabilities. As noted previously in this section, executing Repadmin /showreps on one of the DCs lists the application partition's replication information just as any other NC:

```
Kansas City\NA-DC5
DC Options: (none)
Site Options: (none)
DC object GUID: 5b557f71-d9f1-4ad2-9252-185eefa117eb
DC invocationID: ac23b6a7-9734-4388-b644-80ba020aab13

==== INBOUND NEIGHBORS ==============================

<snip>

     DC=Payroll,DC=company,DC=com
       Seattle\Company-DC1 via RPC
         DC object GUID: df3cf60f-5b62-469a-a957-1782b52a00e8
         Last attempt @ 2003-11-07 19:47:56 was successful.
       Oslo\EU-DC3 via RPC
         DC object GUID: 144b8bc1-3321-4fcf-9b27-40c3f2cc0346
         Last attempt @ 2003-11-07 21:32:52 was successful.
```

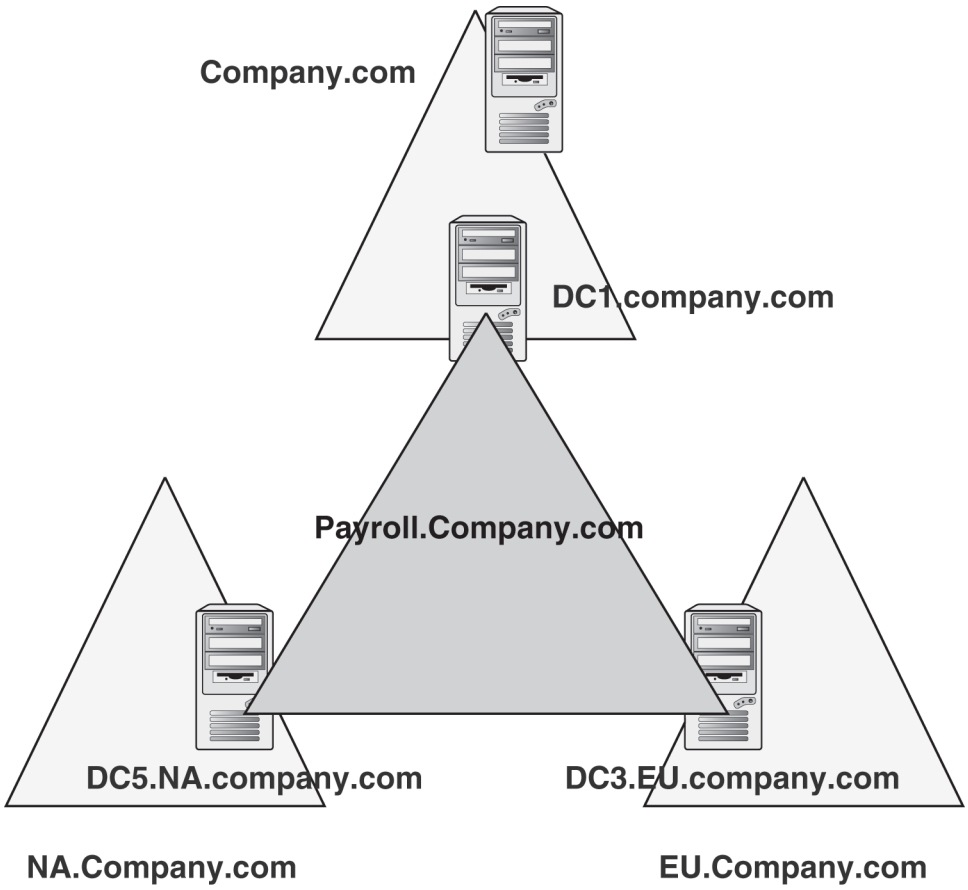

Figure 1.15 User-defined application partitions can use any DC from any domain in the forest as a replica.

Default Application Partitions: ForestDnsZones and DomainDnsZones

Microsoft implemented two application partitions in Windows 2003 domains that are created and configured by default when a domain is created. They are called ForestDnsZones and DomainDnsZones, and you can see them in the DNS snap-in as forward lookup zones. All DCs (including GCs) in the forest are replicas for the ForestDnsZones partition, and all DCs in a domain are replicas for the DomainDnsZones partition. There is a separate DomainDnsZones partition for each domain in the forest. We describe these partitions in detail in the "DNS" section in Chapter 6, "The Physical Design and Developing the Pilot." Microsoft says it's inappropriate to replicate data where it isn't needed. In an Active Directory Integrated (ADI) zone, the

zone data is held in AD and is replicated to all DCs, whether they are name servers or not. With the application partitions of ForestDnsZones and DomainDnsZones, DNS zone information is replicated only to DCs that are also DNS servers. ForestDnsZones contain only SRV records of DCs who are also DNS servers in the forest, and DomainDnsZones contain only SRV records of DCs who are also DNS servers in the domain. Each domain has one of these zones.

Because it really isn't recommended to use these default partitions for other purposes, let's look at an example of how you can use a custom application partition.

User-Defined Application Partition Example

The CFO of a company wants to develop a new in-house application to manage the compensation plan of the company's employees. Most of the data that the application will use is semi-static (pay rate, tax ID, health benefit selections, health benefit costs, and so on) and typically changes only on an annual basis. The human resources organizations that will leverage the application are split into two distinct groups. The NACompensation group manages North American employees and is located in New York. The APCompensation group manages Asia-Pacific employees and is located in Tokyo. The CFO mandates to the developers that unlike most information in the AD, the data managed by this application should be considered sensitive and should be replicated only between New York and Tokyo (both hub sites), but nowhere else.

The in-house developers decide that the data needs to be replicated between both the New York and Tokyo sites and should also be updated from both locations for the purposes of day-to-day human resources functions, data analysis, and application fault tolerance. As such, the developers determine that the type of data, the data's user affinity, the requirement for replicated data, and the need for multi-master updates fit well with the features and capabilities of AD. The developers decide to use an AD application partition, which is created on a DC in New York, followed by the addition of a Tokyo DC to the replica set. After the DCs have a replica of the application partition, they register an A record in DNS for the application partition name as well as an _ldap SRV record for the site in which the server resides.

The benefits of this architecture include the following:

- The application server can obtain data via the LDAP from the AD.
- The application can use DNS to locate a local LDAP server hosting the application partition.
- The AD will allow data to be written to the AD partition in either the New York or Tokyo DC.
- Data written to either DC will replicate to the DC in the other geography, but not to any other DCs.
- If the DC in New York fails, the application can fail over to the DC in Tokyo (the reverse is true as well).

The down side to this is that these partitions will work only on DCs, and it has long been recognized that installing applications on DCs is *not* a best practice. Another option is another Microsoft product called Active Directory Application Mode (ADAM). ADAM allows LDAP querying and replication to nondomain controllers but has some restrictions. We discuss ADAM in Chapter 5, "Active Directory Logical Design."

Creating an Application Partition

In the following example, we will create an application partition called MyAppPart in a forest that has three domains: Company.com, NA.company.com, and EU.com.

1. From a command prompt, run **ntdsutil.exe**. Go to the Domain Management menu, then to the Connection menu, and connect (bind) to a DC in the domain. In this case, we'll assume we connected to DC1.
2. Create an application partition named MyAppPart. At the domain management command prompt, type

   ```
   create nc dc=myAppPart,dc=na,dc=company,dc=com dc1.na.company.com
   ```

3. Make sure the MyAppPart folder exists under the NA.Company.com forward lookup zone and that there is an SRV record in it for DC1.
4. To add DC2 from the Company.com domain, at the Domain Management menu, type

   ```
   add nc replica dc=myAppPart,dc=company,dc=com dc2.na.company.com
   ```

5. Again note that there are SRV records for this new DC in the MyAppPart zone in DNS. You also can list all the DCs hosting the partition by going in NTDSUtil to Domain Management and entering this command:

   ```
   list nc replica dc=myAppPart,DC=company,dc=com
   ```

6. All the DCs added as replicas should be listed.

Finally, when a DC is demoted, DCPromo recognizes whether that DC is the last replica of an application partition. You will see a screen listing the application partitions and asking for confirmation that you want to continue and delete these partitions.

AD Replication

Microsoft made a number of improvements to AD replication. This section provides a quick summary of these improvements, but you'll find a more detailed description, including examples of implementation, in the "Replication Topology" section of Chapter 5.

Lingering Objects

Lingering objects have been causing problems in Windows 2000 deployments for some time, yet at conferences, none of the attendees recognize this issue. Simply stated, lingering objects are objects that are reanimated after their tombstonelifetime expires and is purged. This is caused when a DC or GC comes back online after having been offline for more than the tombstonelifetime period, 60 days by default. Objects that were deleted, tombstoned, and purged while the DC/GC was offline are replicated back into the environment when the DC/GC comes back online. This causes security problems because the user object of a dismissed employee could be reanimated, cause replication to break, and otherwise clutter the AD. The real problem is when a GC comes back online and reanimates read-only contexts of the objects, which were difficult or impossible to delete before Windows 2000 SP3.

Windows 2003 and Windows 2000 SP3 and later provide functionality to prevent these objects from replicating through the forest and permit deletion of the read-only objects via a Registry key. "Tight" behavior refers to a condition in which replication from a DC/GC trying to reanimate purged objects is shut down until it is repaired. "Loose" behavior allows the lingering objects to be reanimated. Loose behavior is the default in Windows 2000 and is the default after upgrading from Windows 2000 to Windows Server 2003. We discuss this thoroughly in Chapter 5.

ISTG Performance and Practical Limit to Number of Sites

Even when Windows 2000 was still in beta, Microsoft was promising improvements in the performance of the Intersite Topology Generator (ISTG). The ISTG uses the spanning tree algorithm to generate the connection objects and the AD replication topology. This is the foundation of AD replication. The efficiency of this topology determines replication latency and the time the AD takes to perform its calculations. The performance of the KCC was inefficient enough to impose a practical limit on the number of about 200 sites in a topology in Windows 2000. Windows Server 2003 has a completely rewritten spanning tree algorithm that makes dramatic improvements in the ISTG's performance, raising the site limit to about 3,000, according to Microsoft testing as of this writing. It also has improved general AD performance and reduced latency in many cases.

Chapter 5 provides a complete description of this issue.

Bridgehead Load Balancing

Another limit in Windows 2000 was the number of sites that could be connected to a single hub site in a single domain. That is, if you have a strict hub and spoke topology and a single domain, you are limited to about 100 to 150 sites connected to a hub site. This is due to the fact that the KCC will only pick one Bridgehead Server (BHS) per domain per site, and large numbers of sites put a load on that single DC in the hub site sufficient to impact the DC's performance.

The solution in Windows 2000 was to create all the connection objects manually, making connections from the site BHS to multiple DCs in the hub site. Unfortunately, this had to be managed manually as well. Windows Server 2003 provides the Active Directory Load Balancing Tool (ADLB), which is a GUI-based tool you can use to create and manage these connections. Chapter 5 provides details on this feature as well.

Improved Data Compression

Windows Server 2003 has improved the algorithm that decompresses inter-site replicated data. This improves replication performance and reduces latency. Windows Server 2003 also permits you to turn off inter-site compression of data, trading the increased bandwidth for local DC performance. (See Chapter 5 for additional details.)

Other

Chapter 5 also covers a number of improvements in Windows 2003 Server in the area of AD replication. The chapter describes and analyzes practical examples and case studies of companies that have had success and failure in deploying AD in Windows 2000 and 2003.

FRS and DFS

Chapter 5 also covers the new features and fixes made in Windows Server 2003, and provides a good description of new troubleshooting and diagnostic tools. For the purpose of this chapter, a brief summary is in order.

- FRS performs serialized version vector joins: Windows 2000 used parallel version vector joins when a DC joined the domain to source FRS content from other DCs in the domain. This caused all FRS content from all DCs to be pulled to the new DC at the same time, causing a high-bandwidth utilization on the network, and the sourcing DCs were unavailable for a period of time. Windows 2003 provides a serial vvjoin whereby one DC is sourced for the new DC, and then the others are sourced, one at a time, for changes, resulting in better performance and higher availability of the DCs.

- Changes to the automatic non-authoritative restore functionality: When Windows 2000 FRS encountered certain serious errors, such as Journal Wrap, it would automatically perform a nonauthoritative restore from a good DC to the one with the error, forcing a full sync of FRS content. This caused the sourcing DC to be unavailable for a period of time. Windows 2003 turns this behavior off by default, logging an event to inform the Administrator that a nonauthoritative restore should be performed at the Administrator's earliest convenience. You can turn the default back to automatic via a Registry key.
- NTFS Journal size: FRS uses the NTFS Journal to identify changes to files in the SYSVOL tree and then issue change orders to replicate those files. When the NTFS Journal was filled, it would overwrite the entries at the start, causing FRS to get lost, break replication, and require a nonauthoritative restore to get going again. Windows 2003 raised the Journal size from 32MB in Windows 2000 to 128MB.
- FRS detects and suppresses excessive replication: Windows 2003 makes FRS more tolerant in the event of an application that scans the SYSVOL tree, such as antivirus or disk defragmenter programs. These programs can cause large numbers of files to be copied to the staging directories. FRS now identifies situations where files are being replicated repeatedly in short periods of time and suppresses the replication, accompanied by an event.
- FRS does not stop replicating if the staging area is filled: Windows 2003 still has a 660MB limit on the staging area before replication is disabled, but it implements a cleanup operation. When 90% of capacity of the staging directory is filled, FRS deletes the oldest files until the directory is at 60% capacity, thus never reaching the limit.
- Allows compressed data replication: Windows 2003 FRS allows replication of compressed data, which is not possible in Windows 2000.
- Tools: Ultrasound and Sonar
- Multiple Distributed File System (DFS) roots on a single server: Windows 2000 allowed only a member server to host a single DFS root. Windows 2003 permits multiple DFS roots on a single server.
- DFS link targets can point to targets across different domains.
- The DFS MMC (Microsoft Management Console) snap-in that is included in Windows 2003 and the Windows 2003 Admin Pack includes the capability to customize a replica set's FRS replication topology.
- DFS referral list can be configured to prioritize DFS servers based on site cost, which is much more efficient than in Windows 2000.

Time Services

Kerberos authentication relies heavily on accurate time synchronization between all computers in the forest. By default, the time skew between any two computers in the forest must be five minutes or less. Windows 2003 uses the Network Time Protocol (NTP) for a much more efficient means of synchronizing computers in the network than Simple Network Time Protocol (SNTP) used in Windows 2000, with the capability to synchronize computers within milliseconds. Windows Server 2003 also provides a new version of the Win32tm.exe utility to configure time services. Chapter 5 describes in detail how time services work in conjunction with Kerberos for security, as well as time-service configuration and troubleshooting tips.

Domain Rename

If Windows Server 2003 were a retail business with a big electric sign out front advertising three exciting reasons to patronize the business, Domain Rename would likely be one of the three. With mergers, acquisitions, and divestitures becoming commonplace in the business community these days, the ability to rename a domain or forest is a very useful tool, especially because the alternative is to migrate users, groups, and computers from one domain/forest into another. In fact, HP is a good example of such a situation. Compaq had developed an extensive Windows 2000 infrastructure, participating as an early adopter in Microsoft's Rapid Deployment Program (RDP) and Joint Deployment Program (JDP). At the time of the merger with HP, Compaq was just completing a two-year migration of users from the old NT domains inherited from mergers with Digital and Tandem. HP, on the other hand, had a small NT environment and had not built a Windows 2000 structure at all. The decision was made to use Compaq's Windows 2000 infrastructure and to migrate the HP users into it. This made perfect sense, except for the name of the domain. The internal namespace for Compaq was CPQCORP.NET and was structured as shown in Figure 1.16. Thus, there became a business requirement to rename the domain. HP decided that the new domain should be HPQCORP.NET—the net change being one letter—C to H!

As of this writing, HP has not renamed this domain, but is planning on it. However, Microsoft has had a couple of customers successfully rename a domain, and Microsoft successfully renamed its corporate development domain as well. One of its customers mistakenly renamed a domain with Exchange 2003 (pre-SP1) and then had to rename it back to the original. Pretty impressive to do it twice, ending up with the same name and be successful. It also shows good resiliency.

This section provides a fairly high-level description of how Domain Rename works to give you an idea of what's involved. We give pointers to Microsoft whitepapers that provide the details on how to actually rename a domain. The remainder of the section reviews Domain Rename from a design perspective, discussing capabilities and limitations of Domain Rename, application compatibility issues, details on Exchange compatibility, benefits and risks analysis, and a few examples along the way.

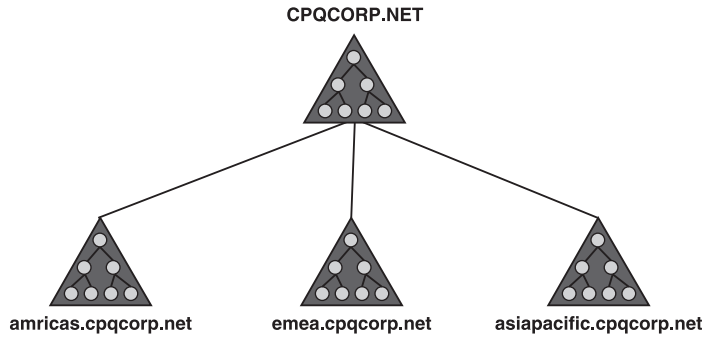

Figure 1.16 Compaq's Windows 2000 and 2003 domain structure.

How Domain Rename Works

The Domain Rename procedure is well documented in two whitepapers, *Understanding How Domain Rename Works,* and *Step-by-Step Guide to Implementing Domain Rename,* which you can download from `http://www.microsoft.com/windowsserver2003/downloads/domainrename.mspx`. In this section, we don't rehash the material in the whitepapers, but rather, summarize their contents and apply my experience to make recommendations. The whitepapers are the definitive guide to Domain Rename, so download, read, and use them. The step-by-step guide provides the actual procedure. I worked at Microsoft during the Windows 2003 beta documenting training materials for Domain Rename, and have monitored HP's Domain Rename efforts. In addition, I have worked with a few customers who were considering using it, so I've had a good bit of exposure to Domain Rename. Following is a high-level view of the procedure.

The forest must be at Windows 2003 functional level, and all domains must be at Windows Server 2003 functional level. This means every DC in the forest must be Windows Server 2003. Identify a member server in a domain (must have reliable access to the domain naming master) from which to run the Domain Rename commands. Get the Domain Rename tools, rendom.exe and gpfixup.exe, from the Windows Server 2003 CD in the \ValueAdd\MSFT\MGMT\Domren directory, from the Web at `http://www.microsoft.com/windowsserver2003/downloads/domainrename.mspx`, or from the Web site that contains the whitepapers noted previously. These tools are also available on the Windows Server 2003 CD, but the versions included on early Windows Server 2003 CD releases broke Exchange. It's best to get them from the Web site to get any new versions available. The *Step-by-Step Guide to Implementing Domain Rename* whitepaper details the process used to rename the domain. Following is an overview of that process:

1. Using the rendom.exe tool, generate an XML (Extensible Markup Language) listing of the naming contexts in the forest.

2. Edit the XML, changing the old domain name to the new, and create a new DNS forward lookup zone for the new domain.
3. Run **rendom.exe** and generate a list of the DCs in the forest.
4. Run **rendom.exe** and upload the instructions for the Domain Rename, including the information in the XML just created, to the domain-naming operations master. This populates attributes *msDS-UpdateScript* and *msDS-DNSRootAlias*. The *msDS-UpdateScript* attribute actually contains the VB Script that performs the Domain Rename. In addition, as a side effect, the *Service Principal Name* attribute of the DC accounts is updated as well. You can view the script using the LDP tool as an attribute on the partitions container (CN=partitions,cn=configuration,DC=company,DC=com).
5. Make sure all DCs can be contacted, and then execute the script on each DC. You can retry the command to execute the script as many times as needed until all DCs execute it successfully. The rename process generates a "state" file every time you execute the command to run the script on each DC. The state file reports the state of each DC (that is, whether it has been updated). Rerunning the command to execute the scripts does not execute on DCs that have been successfully updated.
6. Run **GPfixup.exe** to fix Group Policy associations.
7. Clean up Certificate Authority (CA), domain-based DFS volumes per the Microsoft whitepapers.
8. Clean up other applications.
9. Rename all DCs in the forest so they have the new DNS suffix.

These steps have been included here to show you how complex this process really is, and that there really isn't an easy way back once you start. The key is step 5. You need to realistically determine how long it will take to contact every DC in every site in the entire forest. According to HP's AD team, end-to-end replication took a few hours, but they estimated that it could take several weeks to update all the DCs in the Domain Rename process.

Now let's examine the capabilities—what Domain Rename can and can't do—to determine whether this operation can meet your requirements.

Capabilities of Domain Rename

It is important to note that Domain Rename is not the same as "Prune and Graft." Prune and Graft operations usually imply a "merge" of two forests; that is, Company A buys Company B. Moving the B.com domain to become a child of A.com (B.A.com) is referred to as pruning and grafting. Domain rename is bounded by the forest.

note

> Domain Rename is *not* the same as Prune and Graft. Prune and Graft, or merging of domains across forests, is not possible in Windows 2003.

Now let's see what you can and can't do with Domain Rename. The following list identifies significant operations that Domain Rename can perform:

- Rename a single domain.

  Examples:

  B.A.com → X.A.com

  C.B.A.com → C.Z.A.com
- Rename a domain to change its parent.

  Examples:

  C.B.A.com → C.A.com

  C.B.A.com → C.com

  C.B.A.com → C.D.com
- Rename a domain to create a new domain tree within the same forest.

  Example:

  C.B.A.com → C.com
- Rename a forest root domain.

  Example:

  A.com → Z.com

  B.A.com → B.Z.com
- Rename the DNS name, the NetBIOS name of a domain, or both. Some enterprises have had a problem when upgrading from Windows NT to Windows 2000 because their domain name included a special character such as a hyphen (Master-MUD), which was not supported by DNS. So they created a split name in Windows 2000 with the NetBIOS name retaining the name with the illegal character, and the DNS name comprising a new version without the character. Domain Rename enables you to rename the NetBIOS name without renaming the DNS name.

Limitations of Domain Rename

Domain Rename has a number of limitations, some of which could prevent you from deploying Domain Rename, forcing you to use a migration solution instead. These limitations include the following:

- The forest root domain cannot be restructured for placement in another location in the domain tree, such as moving it to become a child domain of another domain.

  Example:

  B.A.com → A.B.com

  A.com → A.Z.com
- Applications may not support Domain Rename. See the upcoming "Application Compatibility" section of this chapter for more details.
- Renaming a domain impacts all child, grandchild, and so on, domains under it. You must rename all of them.
- All DCs must be able to be contacted during the rename process. If you can't contact them to execute the rename instructions, you have to shut them off and reinstall or manually demote them, and then promote them back into the new domain.
- Because the rename instruction is executed independently on each DC, if some DCs cannot be contacted, some DCs will become members of the new domain and some will still be members of the old domain. This is a problem because it essentially splits the DCs into two domains and as such it cannot replicate domain data between them. If there are changes such as group membership modifications, user password changes, user object modification, and so forth, they will not replicate to the other domain until the Domain Rename has completed successfully. This could result in a loss of service.
- All clients must have their domain suffix changed to the new domain. However, you can do this prior to the rename operation via Group Policy.
- Domains cannot be "merged" between forests (pruning and grafting).

Application Compatibility

The problem at this point in the evolution of the Domain Rename technology is that there just isn't much data on application compatibility. Even Microsoft, at the time of this writing, doesn't have a definitive list of Microsoft applications that support Domain Rename. Note that no application is completely "incompatible," as it can certainly be uninstalled and reinstalled, but obvious costs and risks are involved in doing that. It is important to include application compatibility in any cost/benefit analysis for Domain Rename. There are certainly other ways to

solve a problem than by renaming the domain and cleaning up afterwards. In the next section, we discuss the cost/benefit of Domain Rename in detail.

Although our experience with Domain Rename is limited at this point, we do know the following:

- Domain Rename is compatible with no issues for NETLOGON, LSASS, and Key Distribution Center (KDC).
- Domain Rename is compatible with workarounds for DFS, Group Policy, Unintentional Disjoint Namespace, Trusts, and Certificate Services. Refer to the Domain Rename whitepapers we noted earlier in this section for details.
- Domain Rename is incompatible with Exchange 2000 and Exchange 2003 (pre-SP1). See details in the upcoming "Exchange Recovery Options" section if you rename a domain with Exchange 2000 or 2003 pre-SP1 deployed (these options aren't pretty, but Microsoft says they work).
- Exchange 2003 SP1 supports Domain Rename. However, as of this writing, SP1 has not been released, so there are no cases to cite.
- Systems Management Server (SMS) supports renaming the DNS name, but not the NetBIOS name at this writing. Check with Microsoft for current information.
- Certificate Services has workarounds to support Domain Rename.
  - You can prepare and clean up Certificate Services to support Domain Rename. Details are in the Microsoft Domain Rename whitepapers previously noted.
  - You must move root CAs on DCs in renamed domains to member servers. See Microsoft KB 298138: "HOW TO: Move a Certification Authority to Another Server."
  - You can't rename a DC that is also a CA. Actually it's possible, but the process to make it work is prohibitive.

note

> The information in this list is accurate as of this writing. Because reviewing application compatibility is an ongoing process, you should contact Microsoft via its Web site or directly via its support center to determine the current status of these and other applications.

When I asked Microsoft for a list of applications it had compiled, the company told me that rather than providing the list (which will have changed by the time this book is printed), it preferred to give the following guidelines:

1. Check with the support provider for each application in question. There are simply too many products in the market to keep an exhaustive list updated, and the application developer is the one most deeply familiar with the dependencies of the product.
2. Perform extensive testing in a lab test environment.
3. The only way to be fully prepared is to test, test, and test. Each organization is different and has unknown variables that may or may not be impacted by a Domain Rename.
4. You should conduct the tests in a lab environment that represents the customer's production environment as closely as possible.

Exchange Recovery Options

It has already been noted that Exchange 2000 and Exchange 2003 (RTM or pre-SP1) do not support Domain Rename, and renaming a domain with these versions of Exchange deployed breaks the System Attendant. Microsoft recommends the following process for recovering Exchange if Domain Rename is deployed:

- **Option 1:** Remove and then reinstall Exchange. Treat this as a disaster recovery scenario:
  1. Uninstall Exchange 2000.
  2. Rename domains back to their original names.
  3. Reinstall Exchange 2000.
  4. Reassociate mailboxes. See KB 326278: "Mailbox Recovery for Microsoft Exchange 2000."
- **Option 2:** Build a new forest and migrate the desired objects. This might not be a viable option for large organizations.
- **Option 3:** Ensure that Exchange 2000 SP1 is installed, and then upgrade to Exchange 2003 with E2K SP1 installed.

Microsoft reported customers who successfully renamed the domain back to the original name as a recovery measure, but careful planning and understanding the issues will avoid this, as it is time-consuming and costly.

In terms of planning, it is important to determine whether Domain Rename is appropriate for your situation. Let's review the benefits and risks to help you determine whether Domain Rename really is the best solution.

warning

Rendom.exe versions prior to 6.0.4011.0 failed to detect Exchange 2000. Such versions erroneously allowed Domain Rename operations when Exchange was detected in the forest. Broken versions shipped on Microsoft.com in 2003 and on Windows 2003 installation media. Domain Rename breaks Pre-SP1 Exchange's System Attendant and implicitly Exchange 2000 name resolution. The fix for this was to rename modified domain names back to the original name. The updated rendom.exe is available from `http://www.microsoft.com/windowsserver2003/downloads/domainrename.mspx`. Exchange SP1 and later supports domain rename, but make sure you use the rendom.exe from this Web site.

Benefits and Risks

When considering Domain Rename as a solution for your situation, review the previous list of capabilities and limitations to determine whether Domain Rename fits your requirements. If it does, assess the impact on downtime, testing, and IT staff costs as well as how much time it will take. Remember that you must contact all DCs in the forest to complete the operation. Make sure all your applications will support Domain Rename as well. As noted previously, the best way to do this is to test your applications' compatibility in a Domain Rename operation in an accurate test environment.

Next, determine how a simple migration stacks up against this criteria and see which one (migration or Domain Rename) makes sense. Microsoft's experience at this point suggests the following:

- Simple environments accomplish Domain Rename fairly easily.
- Large organizations can benefit from Domain Rename as a cost-effective alternative to a costly migration operation. However, a complex environment introduces a lot of variables that must be examined for compatibility, such as applications, and conditions that don't support Domain Rename noted in this section.
- Organizations that need to repair the domain namespace, such as those who have a single label name, or a "dotted name" where a dot "." is part of the NetBIOS name, or a name with a nonstandard character that DNS doesn't support. The latter two could be carryovers from NT migration, and usually were accomplished by having NetBIOS names different from the DNS names because DNS has a problem with these formats. Domain rename is a way to clean this up.

- Microsoft had several customers who incorrectly deployed Domain Rename, breaking something (such as Exchange) in the process. They recovered by renaming the domain back to the original name. This is impressive as it shows the resilience of Domain Rename.

In HP's case, although Domain Rename is a big undertaking, it's not nearly as difficult as migrating the users, groups, and computers to a new domain. Remember, it took Compaq two years to migrate the Compaq objects—it would have taken much longer with the HP users and they would have had to start all over again. In HP's case, the decision was basically whether to take another two or three years to remigrate everyone, or spend the time and money to test and implement the Domain Rename.

One customer I spoke to at a conference worked for a subsidiary of a large aerospace company that was changing its company name. The company was currently at Windows 2000 and considered upgrading to Windows Server 2003 so it could simply rename the domain. The company has a single domain, 15 domain controllers in 8 sites, and about 1,000 users. I gave him an explanation of how it works, and helped him create a Domain Rename lab that I created for the conference. After three hours trying the lab (three DCs in the test forest), he decided migration with Active Directory Migration Tool (ADMT) would be easier and more reliable.

Domain rename is a one-way street. Recovery of the original domain depends on your backups, whether you have any DCs left, and how the rest of the environment can absorb the impact. Remember that there are other alternatives to solve your problem; if you use Domain Rename, make sure it is the best solution available, and then thoroughly test it.

Additional Information

Appendix B, "ProLaint Product Details," contains a Domain Rename flowchart that will be helpful in assessing the impact of implementing Domain Rename, as well as providing a good overview for the design of the process. In addition, a training exercise for Domain Rename is available on the book's Web site at `http://www.phptr.com/title/0131467581`. This exercise can easily be done using VMWare or Microsoft's Virtual PC software and will give you a good idea of how Domain Rename is accomplished.

DCPromo

DCPromo improvements in Windows 2003 consist primarily of the new Install from Media (IFM) feature noted previously in this chapter, and some improvements in the DCPromo answer file for unattended promotions. The next section provides a step-by-step procedure on how to use the IFM feature, followed by a section on unattended DCPromo operation. The latter provides details on creating a DCPromo unattended answer file and using it with the IFM restored backup files to create an unattended DCPromo using IFM. These two exercises require a DC and a member server in a domain.

Install from Media (IFM)

Previously, we described this feature from a benefits standpoint in the "Global Catalog Improvements" section of this chapter. This section provides a step-by-step description of how to perform IFM to restore a DC or a GC. The exercise backs up a DC's system state (AD), copies the backup file (.bkf) to a share on a member server, and restores the .bkf to a local directory on the member server.

note

IFM can restore only a replica DC. It cannot create the first DC in the domain because it uses the backup of an existing DC to source from.

1. Log on to a DC as a domain Administrator.
2. Create a directory to hold the backup files, such as C:\backup.
3. Using Windows 2000 Backup, back up the System State (which contains the AD). Save the backup to C:\backup.
4. Log on to the member server that is to become a DC.
5. Create a directory to contain the restore files, such as C:\NTDSrestore, on the member server and share it as NTDSrestore. Grant proper permissions to allow writing to that directory/share for the account you are using.
6. On the DC, map a drive to \\<server>\NTDSrestore (the share created in step 5), where <server> is the name of the member server where the share was created.
7. On the DC, open the Windows 2000 Backup Utility, and use the Restore Wizard to restore the .bkf file created in step 3 to the \NTDSrestore share. Make sure that you

   - Select the System State as the file to be restored.
   - Select Advanced Options, and then specify the location; otherwise, the file will be restored to the original location.

8. On the member server, execute the following command from a command prompt:

```
C:> Dcpromo /adv
```

9. This launches the familiar Active Directory Installation Wizard, but produces an additional screen, shown previously in Figure 1.11 in the "Global Catalog Improvements" section of this chapter. In this dialog box, you can select the From These Restored Backup Files option and enter the path to where you restored the backup: **C:\NTDSRestore**, in this case.

note

The restored files must exist on local media to the member server—tape, CD, DVD, disk, and so on. This operation will not work from a network share.

DCPromo proceeds as usual. The difference is that it will be sourcing from the media, not from a live DC. At the end, it will sync with a live DC to replicate changes between the state of the media and the current state of the AD.

tip

If you test this with a copy of your production data in a lab environment, try running a normal DCPromo operation and time it. Use IFM to promote another DC and compare the time.

Unattended DCPromo

There are only a few new commands to the DCPromo unattended answer file syntax for Windows Server 2003, but the combination of an unattended answer file and the IFM feature presents some intriguing possibilities. For instance, one company had a remote office in Alabama with no IT staff. The company sent the receptionist to training so that she could perform basic tasks that required on-site attention. The company developed a simple .bat file and an unattended answer file to run from it, and sent that on a floppy disk (or even e-mail), along with the restored media and some instructions about where to put things, and she could run the .bat file and promote the DC without having to answer any questions in the process.

In this section, we will demonstrate how you can create an unattended answer file for DCPromo on a member server. Then, you run DCPromo from a command line with the `/adv` switch to allow the AD to be installed from the restored backup rather than from a live DC, with the `/answer`: switch to read the answer file. This exercise assumes you have a DC and a member server in a domain.

1. Follow steps 1 through 7 from the "Install from Media (IFM)" section.
2. Using Notepad or your favorite editor, create a file named Unattend.txt that includes the following:

```
[Unattended]
Unattendmode=fullunattended
[DCINSTALL]
UserName=<enter domain admin acct>
Password=<enter pwd>
UserDomain=<enter domain of the user acct>
DatabasePath=c:\windows\ntds
LogPath=c:\windows\ntds
SYSVOLPath=c:\windows\sysvol
SafeModeAdminPassword
CriticalReplicationOnly
SiteName=<Enter pre-defined site name here>
ReplicaOrNewDomain=Replica
ReplicaDomainDNSName=<name of the domain you are building a replica
for>
ReplicateFromMedia=yes
ReplicationSourcePath=c:\NTDSrestore
RebootOnSuccess=yes
```

   For example, to promote a member server to DC in the company.com domain, locating the DC in the Seattle site and putting the SYSVOL path on the D:\ drive using the account of the Administrator Norm Johnson (NormJ), the answer file would look like the following (assume we save it as DCPromoAnswer.txt):

```
[Unattended]
Unattendmode=fullunattended
[DCINSTALL]
UserName=NormJ
Password=myPassword2003
UserDomain=Company.com
```

```
DatabasePath=c:\windows\ntds
LogPath=c:\windows\ntds
SYSVOLPath=D:\windows\sysvol
SafeModeAdminPassword=Safepwd
CriticalReplicationOnly
SiteName=Seattle
ReplicaOrNewDomain=Replica
ReplicaDomainDNSName=Company.com
ReplicateFromMedia=yes
ReplicationSourcePath=c:\NTDSrestore
RebootOnSuccess=yes
```

3. From a command line (or entered in a .bat file), execute the following command:

```
C:> Dcpromo /adv /answer:dcpromoAnswer.txt
```

If you successfully provide all the answers to DCPromo's questions, you will not be prompted. DCPromo completes and reboots the server. Obviously, you might have typos or other problems, so it's best to test this out before using it in production. A couple of important things to note:

- The siteName option must point to a valid, existing AD site.
- You can leave commands in the file with no answer without error.
- You can use a number of additional commands in these DCPromo answer files. (See Microsoft KB 223757 "Unattended Promotion and Demotion of Windows 2000 Domain Controllers.")
- When running DCPromo in unattended mode, you will not see the dialog box that prompts you for the path to the media, shown earlier in Figure 1.11.
- If you enter the actual password in the answer file, when DCPromo runs, it strips the password out. If you want to run the answer file again, you must re-enter the password each time (or leave it blank and let it prompt you).

warning

The backup media is good only for the time specified by tombstonelifetime (default 60 days). Do not use media older than this, because it will cause lingering objects in the AD. (See Microsoft KB 216993 "Backup of the Active Directory has 60-day Useful Life.")

Linked Value Replication (LVR)

Windows NT performs replication on an object level, whereas Windows 2000 improves on this and performs replication on an attribute level. This is a significant improvement: When you changed the password on a user account, NT 4 replicated the whole object and Windows 2000 replicated the password attribute and not the rest of the object. However, this still had some limitations when replicating multi-valued attributes.

Multi-valued attributes are attributes of AD objects that contain multiple values. Functionally, you can describe the Jet database as a table with rows, columns, and values in the cells. Table 1.3 depicts how a global group object would be stored in Jet. The cells labeled *Group Name*, *Type*, *Members*, and *Scope* are attributes. The entries Domain Users, Security, and Scope are attributes. Note that the group members are all lumped together in a single attribute, instead of each name being an attribute itself. In this case, *Members* is a multi-valued attribute. The problem comes when you remove or modify a member of the group, or add another member. Because Windows 2000 replicates the entire attribute, all values—thus all the members of the group—are replicated, too. In the case of large groups of thousands of members, this can have a negative impact on replication performance due to the increase in network bandwidth required. In addition, there is a limitation in the capability of the database write operation to make a change in large attributes. Because of this, Microsoft does not support groups that contain more than 5,000 members. Deployments like HP's with tens of thousands of users solved this with nested groups.

Table 1.3 Functional Layout of a Global Group Object in the Jet Database

Group Name	*Type*	*Members*	*Scope*
Domain Users	*Security*	Joe Johnson, Jim Shoos, Melinda Urbanawiz, Shanna Witbeck, Colleen Olsen, Lisa Olsen, Catherine Maycock, Steve Andersen, Richard Woodward, Carl Mongrue, Mike Stewart	*Global*

In Windows 2003, replication is performed on attribute values. Thus, for large groups, if the membership is modified, only the value (member) that is added, modified, or deleted is replicated. Thus, a single value can be replicated rather than the entire group. This has eliminated the infamous 5,000-member limit on groups.

Group Policy

Microsoft seems to be following a philosophy to expose options for just about anything through Group Policy. Windows NT had 79 System Policies, Windows 2000 had about 700 Group Policy settings, and Windows 2003 has added more than 200 new settings, many of which deal with new features such as Time Services (NTP), Wireless Network Policy, and Software Restriction policies.

In addition, some significant new tools were produced that make managing and troubleshooting Group Policy much easier. First, we cover the new tools, and then we cover some of the new policies.

Three new tools are significant to troubleshooting and managing Group Policy: Gpupdate.exe, DCgpofix.exe, and the Group Policy Management Console.

GPupdate.exe

This tool replaces the old `Secedit /refreshpolicy`. Remember this because `secedit` no longer refreshes policies. GPupdate.exe includes a number of switches:

- **`/Target` <computer | user>:** Specifies whether to update computer or user policy. No switch refreshes both.
- **`/Force`:** Reapplies all policy settings, not just the ones that changed.
- **`/logoff`:** Forces a logoff if client-side settings have been changed that require a logoff/logon (not background-processed).
- **`/boot`:** Forces a reboot if client-side settings have been changed that require a reboot (not background-processed).
- **`/?`:** Help.

DCgpofix.exe

Customers using Windows 2000 frequently called for support when trying to restore the Default Domain Policy or Default Domain Controllers Policy to the original default settings. Microsoft developed a tool to address this problem. DCgpofix.exe has been enhanced and is now a built-in tool that restores the Default Domain Policy, the Default Domain Controllers Policy, or both using the following switch:

```
/Target: Domain
/Target: DC
/Target: Both
```

In addition, the `/?` switch produces a help file.

warning

If you reset these policies to the default, you will lose all manual settings you have made, including Encrypted File System (EFS) settings.

Group Policy Management Console (GPMC)

The GPMC tool is a welcome sight for Administrators of environments with multiple domains and many policies. It has a number of great features, including the following:

- Manages all policies from all domains in one GUI-based tool.
- Employs Resultant Set of Policy (RSoP). The Planning option reports "what if" scenarios such as what would be the effects of modifying settings on the Default Domain policy, or adding a new policy. The Logging option allows modification of the GPO settings.
- Allows you to dump the settings to a text file, which is very handy for customer support folks trying to diagnose a problem over the phone.

GPMC is available as a free download from Microsoft's Download Center on their Web site, although you technically need to have a Windows Server 2003 license. It runs on Windows 2000 or Windows 2003 domains. However, the machine that GPMC runs on must be a Windows Server 2003 server or a Windows XP client. We include more details and examples in Chapter 5.

New Client-Side Extension Policies

Windows Server 2003 policy includes three new Client Side Extensions (CSEs)—Wireless Network Policy, Software Restriction Policy (SAFER), and QOS (Quality of Service) Packet Scheduler—in addition to the CSEs that were in Windows 2000. The QOS Packet Scheduler was part of several updates to QOS in Windows Server 2003. This feature is used in QOS for bandwidth throttling. Software Restriction Policies are a new security feature described in the "Security" section of this chapter. Wireless Network Policy added settings in the policy to define wireless settings such as preferred networks (by Service Set Identifier—SSID) and types of networks to access. Figure 1.17 shows the Wireless Network Policy Properties page available via the Group Policy Editor.

note

By default, no Wireless Network Policies are defined. The Wireless Network section of the Computer Configuration of Group Policy is blank. Right-click on the Wireless Network icon and select Create Wireless Network Policy. Note that you can create only one Wireless Network Policy.

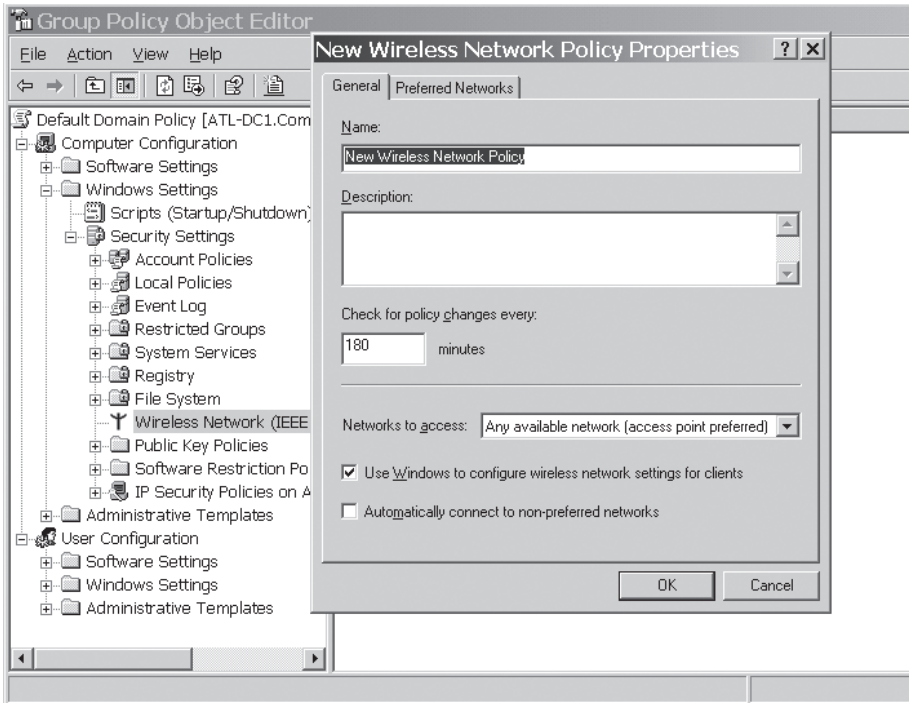

Figure 1.17 Wireless Networks Property Policy page as it appears in the Group Policy Editor.

Changes in User Rights

One of the most glaring changes that trips up a lot of Administrators new to Windows Server 2003 is the old Logon Locally right as used in Windows 2000. This right does not exist as such in Windows Server 2003. It now has four settings:

- **Allow Logon Locally**: Users and groups in this list are allowed specifically to log on locally. The default groups that were set in Windows 2000 (such as that in the Default Domain Policy) are configured here by default.
- **Deny Logon Locally**: Users and groups in this list are specifically denied privilege to log on to the machine.
- **Allow Logon Through Terminal Services**: Users and groups listed are permitted specific logon rights via Terminal Services.
- **Deny Logon Through Terminal Services**: Users and groups listed are specifically denied logon rights via Terminal Services.

New DNS/Net Logon Policies

Net Logon settings, exposed only in the Registry, are easily configured now in Group Policy. Fourteen settings are located in `Computer Configuration\Administrative Templates\System\Net logon`. Some interesting ones include the following:

- **Site Name**: You can specify a site name so that all computers to which this policy applies will be members of this site instead of letting the DC Locator process find one. This setting is potentially dangerous because it's easy to set, but it's useful only if you have all the computers defined in OUs that represent sites. You could also force Site Affinity this way. However, if you forget about it and the site configuration or the OU structure changes, or if you move computers to another OU, you might have trouble figuring out why the DC Locator is behaving the way it is and finding the wrong site. Of course, it can't find the right site because you defined the site manually.
- **Expected Dial-Up Delay on Logon**: You can specify additional time for a dial-up client to wait for a DC's response at logon. This could help make the logon process more tolerant if users are having connection issues due to timeouts.
- **Log File Debug Output Level**: This is a great improvement. In Windows 2000, you had to enable Net Logon logging (to the Netlogon.log) by modifying a Registry key. This policy setting not only turns it on, but also has several degrees of verbosity that you can set. However, the verbosity values are strange. The default verbosity value is *536936447*. Nonzero values turn it on, and the higher the value, the more verbose the logging will be.
- **Contact PDC on Logon Failure**: This setting lets you turn off the functionality of a DC querying the PDC for password change if a user's password is judged by the DC to be incorrect. This was designed to take care of the problem of a password change not being replicated to a DC that authenticates the user after changing passwords. If you have a large geographic domain, and users in remote sites must go over a slow WAN link to get to the PDC, this would save some time and traffic, although it would result in logon failure until replication caught up.

New Windows 2003 DC Locator DNS Records

Located in `Computer\Administrative Templates\System\Net logon\Domain Controller Locator Records`, this is another dangerous set of policies that allow you to change the behavior of the DC Locator. Here are some interesting ones:

- **Dynamic Registration of the SRV DNS Records**: You can turn off dynamic registration of DC DNS SRV records. Each SRV record has a TTL (Time to Live) attribute that is set to 60 minutes by default in Windows 2000. This means that every hour, all DCs update all their SRV records, which can cause a duplication of DNS records. To avoid this, Administrators can turn off Dynamic Registration when all the DCs register their records in DNS.

- **Automated Site Coverage**: You can disable Auto Site Coverage so that affected DCs will not provide site coverage for sites other than their own. This is perhaps a good thing under peculiar circumstances in which you don't want a very busy DC to cover more than its site.

Several other settings allow you to manually constrain the DC Locator. Make sure you have good reason for tweaking these settings, and that you thoroughly test and document them before implementing.

New Client-Side DNS Settings

Several new settings control the client DNS functions, located in `Computer Configuration\Administrative Templates\System\Network\DNS Client`. In Windows 2000, there was only a single setting: Primary DNS Suffix. Now there are 13 settings. Some of the interesting (and frightening) ones are as follows:

- **Primary DNS Suffix**: This oldie but goodie is useful in operations such as Domain Rename that pushes DNS suffix changes to clients in advance of the rename.
- **DNS Servers**: Defines a list of DNS servers to which a client will send name resolution requests. The scary thing is that this supersedes the DNS list in the local TCP/IP, as well as Dynamic Host Configuration Protocol (DHCP) configured DNS servers, and it applies to all network interfaces of the computers that this policy applies to.
- **Dynamic Update**: Although there might be some isolated cases in which you don't want to register a network interface in DNS, this qualifies as yet another opportunity to shoot yourself in the foot. Let's quote from the Explain tab on the setting in the Group Policy Object Editor:

  > If you disable this setting, the computers to which this setting is applied may not use dynamic DNS registration for any of their network connections, regardless of theconfiguration for individual network connections.

  Note that this disables dynamic registration on *all* interfaces of affected computers.
- **`Computer Configuration\Administrative Templates\ System\ Logon\Always wait for the network at computer startup and logon:`** Enabling this setting ensures that Folder Redirection, Software Installation, roaming user profile settings, or perhaps other applications that require network access complete in just one logon. Without this setting enabled, these features might require two logons because the user can be logged on before the network settings are available.

Computer Configuration\Administrative Templates\ System\ Group Policy

- **Allow Cross Forest User Policy and Roaming User Profiles:** Enabling this setting allows a user's policy and roaming user profile to apply when logging in from a computer in another forest that is trusted to the user's home forest via a two-way trust.

note

As of this writing, the application of Group Policy between Windows 2003 forests or between a Windows 2003 forest and an MIT Kerberos Realm does not work for a trust created for forest-wide authentication. In this model, the authenticated users group from one forest is added to the authenticated users of the other, so security is wide open between the forests. (Refer to Microsoft KB 827182 "Group Policy Settings are not applied when you log on to a server by using an account from an MIT Kerberos realm.")

- **Group Policy Refresh Interval for Computers**: By default, workstations refresh policy every 90 minutes with a random offset of 10 to 30 minutes. This policy allows easy reconfiguration. One customer I worked with experienced a strange issue in which every time the policy was refreshed, the computer would hang for 15 to 20 seconds and then work again. Sometimes this is expected (that was the case here). Windows 2000 made this very difficult to modify.
- **Turn Off Background Refresh of Group Policy**: The customer just noted could have enabled this setting, which turns off the refresh while the computer is in use.

System Restore and Remote Assistance

Windows XP and Windows 2003 Remote Assistance

Enabling the Remote Assistance policy allows expert help by permitting access to the computer. Note that the user will still be prompted to allow access to the computer when the expert contacts it. You can find this policy in `Computer Configuration\Administrative Templates\System\Remote Assistance`.

Windows XP System Restore

`Computer Configuration\`The System Restore policy enables or disables the user's ability to restore an XP client to a previous known good state. You can find this policy in `Administrative Templates\System\System Restore`.

Windows Installer System Restore

`Computer Configuration\` Windows Installer creates a system restore checkpoint for each application during installation. You can disable this with this policy setting, found in `Administrative Templates\System\System Restore`.

Error-Reporting Policies

The error-reporting policies enable you to control the error-reporting feature Microsoft first built into Windows XP. A typical error is shown in Figure 1.18. When an error is encountered, this pop-up window appears, prompting the user to indicate whether the information should be sent to Microsoft (assuming a current connection to the Internet). Microsoft uses these reports to gather data on problems to aid in resolution. It's a way of gathering data from a large number of users who would likely not report the problem, and I've actually seen cases in which Microsoft has used the data to resolve problems. In addition, if there is a possible solution, the user will be notified. I've seen cases in which a Microsoft Office application, such as Word, caused an error report and after I sent it, Microsoft notified me that the error could be solved by upgrading to a service pack and gave me the link to it.

Microsoft is not trying to steal information and it offers the option of reviewing the report before it is sent. I highly recommend you always send this report. However, Group Policy also provides a number of options for error reporting, including the following.

Basic Error Reporting

Basic error reporting provides two options:

- **Display Error Notification**: Determines whether the user is offered the option of sending the report to Microsoft.
- **Report Errors Properties**: A number of options in regard to sending the reports to Microsoft. If the Display Error Notification policy is enabled and these properties are disabled, the user will be notified of a problem, but will not be able to report it to Microsoft.

You can find this policy at `Computer\Administrative Templates\System\Error Reporting`.

Advanced Error Reporting

Only one advanced error-reporting option is available. This policy setting contains more granular configuration of error reporting, such as enabling or disabling reporting of operating system errors, unplanned shutdown events, and application errors. You can find this policy at `Computer\Administrative Templates\System\Error Reporting\Advanced Error Reporting`.

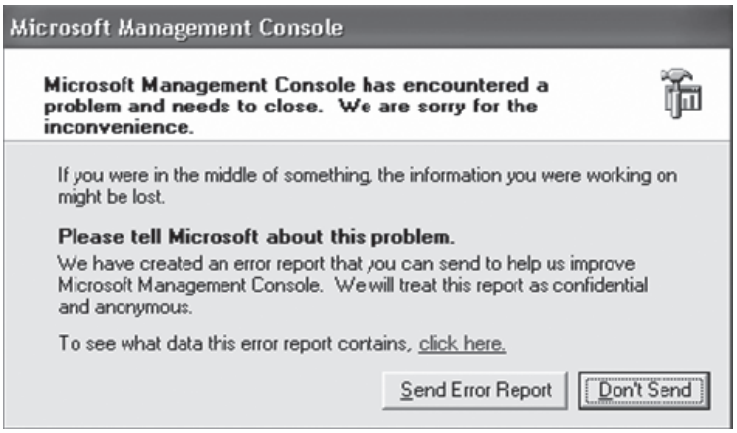

Figure 1.18 Typical error produced by the error-reporting mechanism in Windows XP and Windows Server 2003.

Miscellaneous Policies

- **This is one of those dangerous policies that tend to generate support calls.** The Configure Windows NTP Client setting is particularly dangerous in that it allows you to specify an NTP Server. As noted in the "Time Services" section of Chapter 6, the Windows Time Service structure is configured properly by default and works fine without manual intervention. I've seen one customer already who thought he wanted to configure this by specifying an NTP server. It completely broke Time Services, which, of course, broke authentication. We ultimately had to restore everything to defaults to get it to work again. I recommend just leaving it alone. That is, if it's not broken, don't fix it! You can find this policy in `Administrative Templates\System\Windows Time Service`.
- **Remote Procedure Call:** This policy, which you can find in `Administrative Templates\System\Remote Procedure Call`, contains options that will help with troubleshooting (first two bullets) and make RPC more or less tolerant of errors (last two bullets):
  - RPC (Remote Procedure Call) troubleshooting state information
  - Propagation of extended error information
  - Ignore delegation failure
  - Minimum Idle Connection Timeout for RPC/HTTP connections (relating to the new Exchange and Outlook feature)

New Uses of WMI Filters

Windows Server 2003 provides new capabilities for customizing queries using WMI (Windows Management Instrumentation), including the WMIC interface and the WMI filters in Group Policy. We cover WMI in detail in Chapter 10, "System Administration," in the "Windows Management Instrumentation" section. The WMI filter in Group Policy basically allows a WMI query (filter) to be applied from within a policy. For instance, the following WMI filter checks to see whether the physical memory of the computer is greater than 128MB:

```
Select TotalPhysicalMemory from win32_ComputerSystem where
TotalPhysicalMemory >= 134217728
```

Using the Group Policy Management Console (GPMC), which we describe briefly in this chapter and in more detail in Chapter 10, we can define the WMI filter and associate it with a Group Policy Object (GPO). The procedure to do this is as follows:

1. In the GPMC, expand Forest, Domains, <domain name>. Under the domain name, you'll see a list of all GPOs followed by OU folders, a Group Policy Object folder, and finally the WMI Filters folder.
2. Right-click on the WMI Filters folder, and select New. The New WMI Filter dialog box is displayed.
3. Click the Add button to display the WMI Query dialog box. In the Query field, enter a valid WMI filter, such as the example shown in Figure 1.19.
4. Click OK and return to the New WMI Filter dialog box. You can add other WMI filters at this point, but it's best to do one at a time.
5. Click the Save button. If the WMI filter you added is correct (syntax, and so on), it will be added to the list.
6. Close the New WMI Filter dialog box and you'll see the filter added under the WMI Filter folder in the left pane of the GPMC.
7. Click on the filter you just created and in the left pane, shown in Figure 1.20, you'll see the WMI filter description at the top and a list of *GPOs that use this WMI filter* at the bottom.

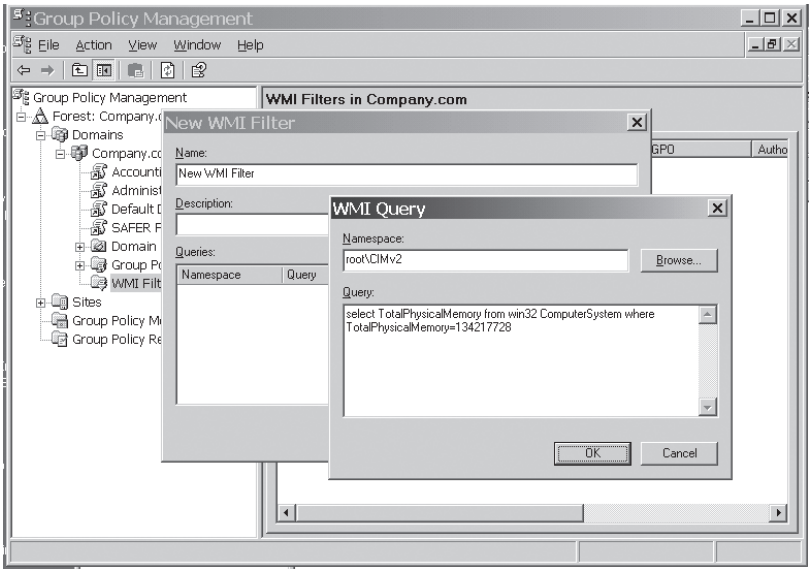

Figure 1.19 Dialog box for creating WMI filter in GPMC snap-in.

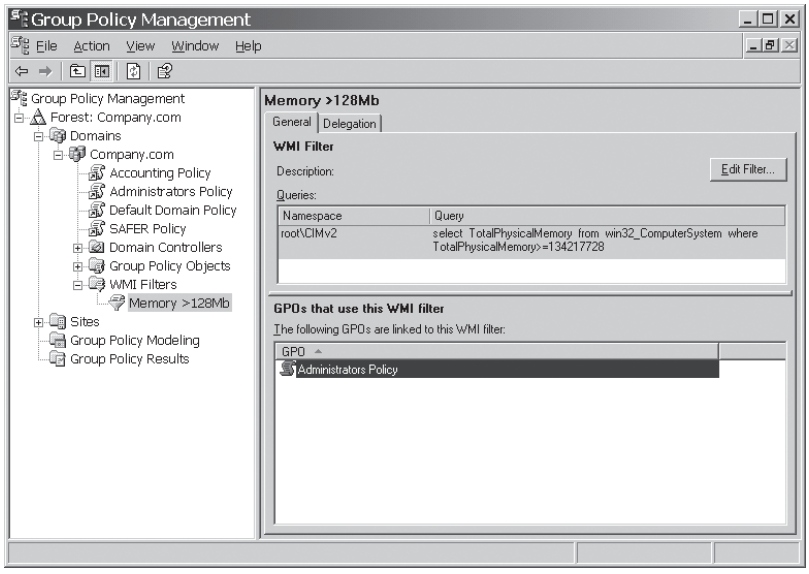

Figure 1.20 Linking a WMI filter to an existing GPO in the GPMC snap-in.

8. Click in the white area of the *GPOs that use this WMI filter* section and select Add. A list of GPOs is displayed. Select a GPO to have the filter apply to and click OK. To add the filter to additional GPOs, repeat this procedure. You can't add more than one GPO at a time.

We have created a WMI filter on the Accounting Policy (in the example in the figures) so that this policy applies only to computers that have 128MB of memory or more. This is similar to the security filtering in Windows 2000, where you could apply or deny a policy only to certain users or groups.

Of course, there is a price to be paid for this. Just like ACL (Access Control List) filtering in Windows 2000, you pay a logon performance price for WMI filtering. In addition, keep in mind that the Windows 2000 clients don't support the WMI queries, so they will always apply the policy, and the filter won't apply to them.

Tools

Microsoft made a number of enhancements to tools used for troubleshooting and monitoring in Windows 2003. The bad news is that most of them won't run on a Windows 2000 machine. The good news is that they will run on a Windows XP client or a Windows 2003 server in a Windows 2000 domain. One of the best troubleshooting tools I've found so far is a Windows XP client. In fact, I've required more than one customer to buy a copy of XP and load it on a client machine so that we could do proper troubleshooting. Here is a summary of some significantly improved tools in Windows Server 2003 (at least the ones I like). Descriptions and examples of how these tools are used are contained in various chapters in this book.

- **Windows XP client**: This is a very useful tool, as it can execute most of the new Windows 2003 version of old tools as well as new ones. I have often required a customer to purchase XP and put it on a laptop so that we could run the enhanced version of the tools in a Windows 2000 domain or forest.
- **GPresult**: One of the most valuable tools ever for diagnosing Group Policy problems on the client, the Windows Server 2003 version of this tool is invaluable, as it produces RSoP information and provides all security and user rights details. Where Windows 2000's version simply told which GPO it got the security from, Windows 2003's version displays all the settings, such as password length. This will reduce your time in diagnosing Group Policy problems considerably.
- **W32tm:** This is a new version of the time configuration utility with new options and syntax. (See Chapter 6 for more details.)
- **GPMC (Group Policy Management Console)**: This new tool allows management of all GPOs from one tool, and allows Administrators to use RSoP in planning and deployment phases. (See Chapter 5 for more details.)

- **ADLB (Active Directory Load Balancing Tool)**: This tool allows management of manual connections and enables you to add additional BHSs to a hub site for load balancing. (See Chapters 5 and 10 for more information.)
- **Sonar**: This is a monitoring tool for FRS. (See Chapter 6.)
- **Ultrasound**: This is an advanced monitoring tool for FRS. (See Chapter 6.)
- **Repadmin**: This new version has a `/replsum` option that displays end-to-end replication information to show outstanding change requests on each DC in the enterprise. It also contains a switch to remove lingering objects, and it will run on an XP client in a Windows 2003 domain. (See Chapter 5.)
- **Dcgpofix (in Windows Support Tools)**: This was previously a secret tool available only from Microsoft PSS. It restores the default Domain Policy, or Default Domain Controllers policy, or both. This is useful if you have hosed either of these and want to restore them to their default condition. It will lose all changes you've made, including EFS settings.
- **NTDSUtil—Change Directory Service Repair Mode (DSRM) Password**: This allows the DSRM Administrator password to be reset online—without rebooting.
- **GPupdate**: This replaces the Secedit `/refreshpolicy` command and is available native to Windows XP and Windows Server 2003.
- **WMIC**: This is an interface in XP and Windows Server 2003 to provide commands to the WMI APIs (Application Programming Interfaces). (See Chapter 10 and elsewhere in this chapter for more information.)
- **Active Directory Users and Computers snap-in**: This policy enables you to select multiple objects (users, groups, and so on); it also allows you to save LDAP queries in a saved queries folder, and permits drag and drop.
- **Remote Desktop**: This replaces the old Terminal Services Administration Mode and Terminal Services Client. Built into XP and Windows Server 2003, Remote Desktop permits copy and cut/paste between a remote console and a local machine. (See Chapters 10 and 15, and section 1.2.3 "Remote Desktop Client and Resource Redirection" in this chapter.)
- **Account Lockout**: This is a lockoutstatus.exe and a DLL (Dynamic Link Library) that produces an additional Acct Info tab in the User object. It permits better management and monitoring of account lockout. (See Chapter 5 for more information.)
- **dsadd, dsmod, dsrm, dsget and dsquery**: These built-in command-line tools permit adding and modifying objects in the AD. Entering the command followed by `/?` produces a brief online help for each command. (for example, dsadd `/?`). Windows Server 2003 only.
  - **dsadd**: Adds a computer, contact, group, OU, quota, or user to the AD.
  - **dsmod**: Modifies a computer, contact, group, OU, server, user, quota, or partition in the AD.

- **dsrm**: Removes (deletes) objects from the AD.
- **dsget**: Displays properties of an object such as computer, contact, subnet, group, OU, server, site, user, quota, and partitions.
- **Dsquery:** Searches the AD for computers, contacts, subnets, groups, OUs, sites, servers, users, quotas, and partitions.

Scalability

Previously in this chapter, I've noted new features that provide enhanced scalability. These features are summarized here only to identify them with the fact that they do enhance scalability for the Windows architecture:

- **Increase of the maximum NTDS.DIT database to about one billion objects**: Like Windows 2000, this is a fairly flat curve in terms of performance. You can increase the number of objects without directly impacting performance. Performance will stay fairly consistent.
- **IFM:** This permits DCs and GCs to be configured or rebuilt in minutes compared to hours or days in Windows 2000. This makes the environment more resilient and easier to extend.
- **Improved replication**: A rewritten spanning tree algorithm made the ISTG much faster and reduced replication latency.
- **LVR:** This enhanced the replication performance of multi-valued attributes, eliminating the size of these attributes, such as the 5,000-member limit of global groups in Windows 2000.
- **Inter-site compression**: This can be disabled, reducing load on BHSs. In addition, the compression algorithm was improved to aid in decompression of data so that even if you opt to leave the data compression intact, there is still a performance increase.
- **ADLB**: Helps manage environments where remote sites with slow links require manual intervention to schedule replication and build connection objects.

Interoperability and Functional Levels

Microsoft introduced a new feature in Windows 2003, called functional levels, which provides interoperability with Windows 2000 and Windows NT DCs. Functional levels were implemented in Windows 2000, but they were simply known as "native mode" and "mixed mode," referring to whether there were downlevel (NT) DCs in the domain. Windows Server 2003 adds a degree of complexity: Not only does it add another OS that a DC can be installed with, but also introduces

the concept of a "forest native mode." Forest native mode means all DCs in all domains in the forest must be at Windows Server 2003, all domains must be switched to native mode (similar to how Windows 2000 domains had to be switched to native mode), and the forest must then be switched to native mode. This is very important because the features and improvements discussed in this chapter depend on the functional level of the domain or forest. To get all the Windows Server 2003 benefits, you must be in a native Windows Server 2003 forest.

Table 1.4 gives a good summary of each functional level, what DCs are allowed (by OS), and a comment about functionality. Chapter 3, "Migration Planning: Business and Technical," provides a thorough treatment of functional levels, including a detailed description of how they work and how to configure them, as well as how they affect the design and the migration.

Table 1.4 Functional Levels in Windows Server 2003 Domains and Forests

	DCs Allowed	Comments
Windows Server 2000 Mixed Mode	Windows 2000 DCs Windows NT DCs Windows 2003 DCs	Same functionality as in Windows 2000 Mixed mode.
Windows 2000 Native	Windows 2000 DCs Windows 2003 DCs	Windows 2000 Native functionality.
Windows 2003 Forest Native	Windows Server 2003 DCs or NT PDC/DC only; no Windows 2000 DCs in any domain in forest	Domain functional level for every domain must be set to "Windows Server 2003" level. Provides full Windows Server 2003 functionality.
Windows 2003 Domain Native	Windows 2003 DCs only; no Windows 2000 DCs in domain	Provides Windows Server 2003 functionality for domain-related functions.
Windows 2003 Interim	Windows Server 2003 and Windows NT 1.0 No Windows 2000 DCs	Used only during NT to 2003 migration, then switched to 2003 native mode.

Schema

In Windows 2000, Microsoft made no small effort to frighten everyone away from modifying the schema because it cannot be restored from an earlier date. If damaged, the schema could cause the entire infrastructure to fail. This is still true in Windows Server 2003. However, there has been an improvement on the front of object management in the schema. Classes and attributes in the schema can now be disabled (as was the case in Windows 2000), but they can also be marked as "defunct" or "redefined." What we need still is the ability to purge and delete, which hopefully is coming in a future update.

Dynamic TTL

In Windows 2000, objects existed until they were explicitly deleted. With Windows 2003, objects can be created with a Time To Live (TTL) stamp. They are then deleted when the time stamp expires unless they are refreshed. This should help keep the AD clean.

Networking

Although the flashy improvements in Windows Server 2003 are arguably in AD, there were some significant improvements and features added in the area of networking.

DNS

DNS added a number of changes and improvements. We discuss these in detail in Chapter 6, but a few are worth noting here for an overview:

- Stub zones are now configurable in Windows 2003 DNS.
- Conditional forwarding allows the Administrator to do custom forwarding of certain zones or domains to specific authoritative DNS servers by their IP addresses. This is a shortcut of sorts—similar to stub zones.
- _MSDCS zone is automatically delegated when DCPromo configures DNS on the first DC in the forest, helping to address the problem in multi-domain forests where the Cname records and GC SRV records are stored in the forest root domain and may be unavailable at times to DCs in child domains. We discuss this in detail in Chapter 6.
- ForestDNSZones and DomainDNSZones are two application partitions that are configured by default in Windows Server 2003 and are visible in DNS as forward lookup zones.

- When application partitions are created, a forward lookup zone is automatically created. We discussed application partitions in detail previously in this chapter.
- The DNS Event Viewer is now included as part of the DNS Management snap-in.

Another improvement in DNS has been the collective knowledge of how it works with AD. Take advantage of Microsoft's volumes of whitepapers, training, and other documents located in the DNS Center (also called DNS Center for Windows 2000), located at `http://www.microsoft.com/windows2000/technologies/communications/dns/default.asp`.

Home Networks

The popularity of consumers having networks in their home to connect multiple computers to the Internet has prompted Microsoft to develop features in the operating system to address their needs. This section describes some of those new features.

Network Bridge

The network bridge feature really has its benefit in the home network environment. It allows you to "bridge" multiple network adapters, such as between a public network and a private network, or between a wireless adapter, an Ethernet adapter, and a dial-up adapter. The network bridge allows these adapters to communicate with each other without requiring the user to set up complicated routing tables. You can enable the bridge easily by selecting the adapters to be in the bridge (hold the Ctrl key down as you select them in Network Connections), and then right-clicking and selecting Bridge Connections. It's really that simple. The bridge is created and shows up in the Network Connections under "Network Bridge," as shown in Figure 1.21. You also can use the New Connection Wizard to set up and administer the bridged network.

Device Driver Enhancements

This is another improvement for home networking, removing legacy drivers that are no longer used or supported and adding or improving drivers in the following areas:

- Local Area Network (LAN) network drivers, such as 10/100 Network Interface Cards (NICs), IEEE 802.11, and Home Phoneline Networking Alliance (HomePNA)
- Broadband, including cable modems, Asynchronous Digital Subscriber Line (ADSL), and Integrated Services Digital Network (ISDN)
- Modems, including driver-based and 56Kbps V.90 modems

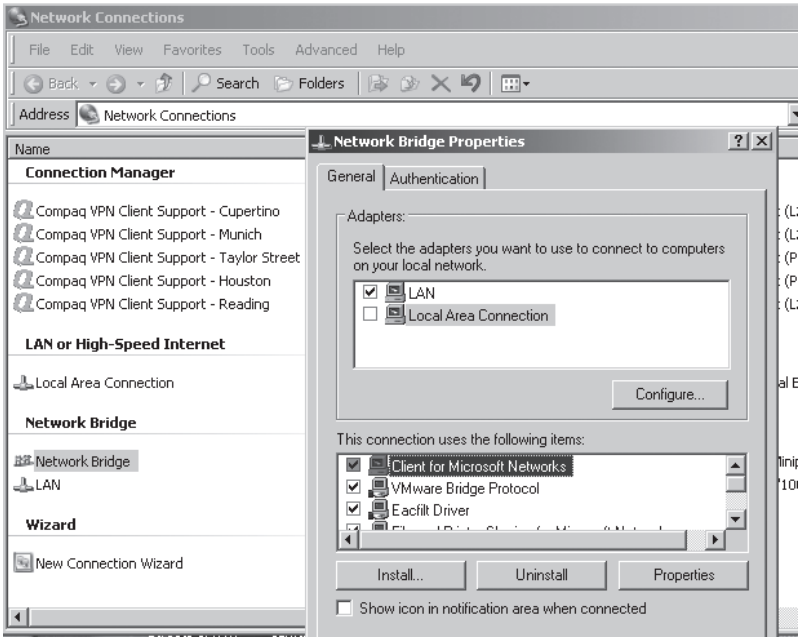

Figure 1.21 Network Bridge is listed in Network Connections window.

Remote Desktop Client and Resource Redirection

In answering customer support calls at HP, and in speaking engagements at conferences, I hear Administrators frequently comment that Windows 2000's Terminal Services (Administration mode) is the best tool Microsoft offers. It allows remote access to servers anywhere in the network. In HP's Qtest environment, we use Terminal Services (TS) to manage, install, configure, and repair servers all over the world. All we need is someone at the site to put in a CD or do physical tasks.

In Windows 2000, you must install the TS client manually by building floppy disks with the Terminal Services Server Manager or by running `%systemroot%\system32\clients\tsclient\win32\disks\disk1\setup.exe` from the server. However, only Windows Servers can host the client and allow remote logon. In Windows XP and Windows Server 2003, the TS Client is replaced by the Remote Desktop and is installed and enabled by default. Go to Start, Programs, Accessories, Communications, Remote Desktop Connection, and enter the name or IP address of the computer you want to connect to. You can use Remote Desktop to connect to Windows 2000 Server, Windows 2003 Server, or Windows XP client.

The Remote Desktop feature permits resource redirection, which means you can access your local disks, printers, and even speakers through the Remote Desktop session. This allows you to log on to a remote session and save files (such as log files) to your local disks—a tremendous benefit and improvement over Windows 2000 TS. In addition, Remote Desktop is available on Windows XP clients, allowing you to establish remote sessions to client computers.

Chapter 15, "Terminal Services for Windows Server 2003," contains more in-depth detail on Remote Desktop connections in Windows XP and Windows Server 2003.

Internet Authentication Service (IAS)

In addition to a new MMC snap-in, IAS has several new features. Two important features are noted here.

IAS and Radius Client

In Windows 2000, you can use an IAS server only as a Radius server, configured to perform access request authentication against the domain. In Windows Server 2003, you can configure an IAS server as a Radius Proxy that either authenticates the remote request or forwards the request to another Radius server.

IAS and Cross-Forest Authentication

You use cross-forest authentication to authenticate the user account when two AD forests are connected with a two-way, cross-forest trust. We discuss these trusts in the "Security" section of this chapter.

New Features in RRAS

IPSec over NAT

VPN clients behind a NAT (Network Address Translation) can now establish IPSec (IP Security) or Level 2 Tunnel Protocol (L2TP) tunnels to a Windows 2003 server. This was not available in Windows 2000. You can use this feature to make a connection to the company's internal network when one server is in the company's DeMilitarized Zone (DMZ), a branch office, or perhaps a client in a home network that shares a single IP address behind a NAT.

Broadcast Name Resolution

Also known as the NetBT Gateway, this feature provides TCP/IP name resolution for RAS (Remote Access Service) clients where no WINS (Windows Internet Naming Service) or DNS servers are available. This is similar to the WINS proxy and is shown in Figure 1.22. This is advantageous for small networks in which a DNS or WINS server is not in place.

When the RAS server receives a request to resolve a name to an IP address, and it does not have that information in its NetBIOS name cache, it performs a NetBIOS name query over the network on behalf of the RAS client. The computer who owns that name replies to the RAS server, who caches the name/address and sends the reply to the client. The client can then contact that computer resource. The NetBIOS name cache in the RAS server and the client has a ten-minute lifetime.

note

If the *NetBIOS over TCP/IP* option is disabled on the RAS server, Broadcast Name Resolution will fail.

Broadcast Name Resolution is enabled by default, but you can disable or reenable it by doing the following:

1. In the Routing and Remote Access snap-in, right-click on the appropriate server icon.
2. Select the IP tab.
3. Clear the *Enable broadcast name resolution* option (or check it to enable the service).

This option is controlled by the Registry key `EnableNetbtBcastFwd`:

```
HKEY_LOCAL_MACHINE\CurrentControlSet\Services\RemoteAccess\
Parameters\Ip\
Name: EnableNetbtBcastFwd
Type: REG_DWORD
Data: 0 (disabled)
      1 (enabled)
```

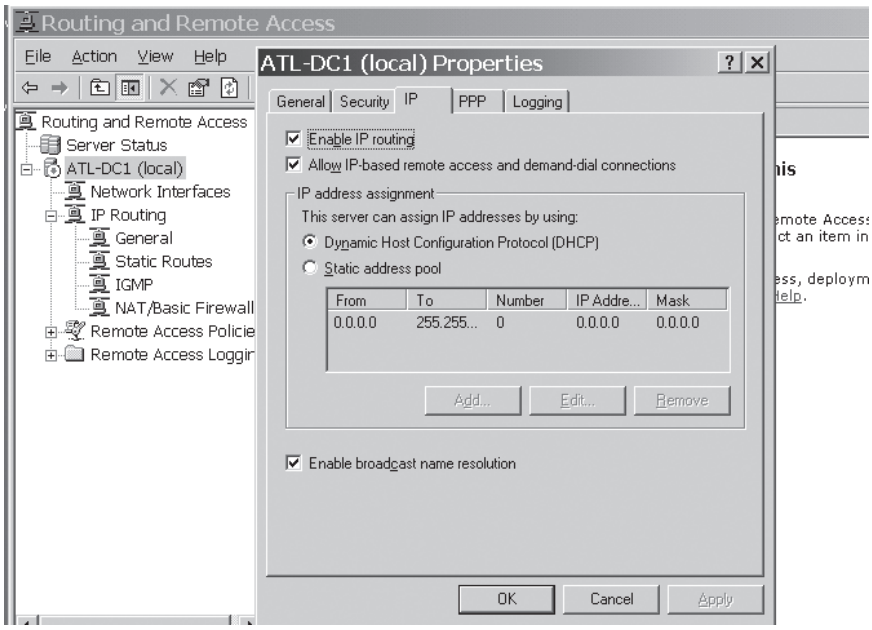

Figure 1.22 Configuring the NetBT Gateway in the RRAS snap-in.

RRAS Firewall and NAT Integration

This feature enables you to integrate a firewall with a RRAS NAT function. It provides the Administrator a built-in firewall in RRAS. I have used this firewall for my home network, and while it isn't as flashy as some third-party applications, and it doesn't have logging or reporting features, it works quite well.

Enable RAS Interface as a NAT Private Interface

This feature allows a user, as a RAS client, to access the Internet via a server that is used for both NAT access to the Internet and dial-in access to his or her corporate network. Previously, you could not use this server for both.

Protocols

Support for IPv6

Windows Server 2003 supports the IPv6 protocol, which may someday replace the existing IPv4 standard TCP/IP. IPv6 provides for 128-bit addresses, or more than 3.4×10^{38}.

Removal of Legacy Networking Protocols

In Microsoft's effort to increase security, Windows Server 2003 has eliminated inclusion of and support for some legacy protocols. These include

- Data Link Control (DLC).
- NetBIOS Extended User Interface (NetBEUI).
- Internet Packet eXchange/Sequenced Packet eXchange (IPX/SPX) and IPX dependent services have been removed from RRAS in all versions of Windows Server 2003, and thus cannot use IPX for routing, VPN, or RAS. In addition, IPX/SPX is not included in the 64-bit version of Windows Server 2003.
- Infrared Data Association (IrDA).
- Open Shortest Path First (OSPF).

Addition of New Protocols

HTTP.sys

HTTP.sys is a kernel mode driver that supports client-side and server-side APIs, although client-side APIs are disabled in the Windows Server 2003 implementation. Some server applications already take advantage of HTTP.sys, including IIS (Internet Information Services) v6.0, SQL's next release code named Yukon that are all usermode applications. HTTP.sys also supports the WebDAV (Web-based Distributed Authoring and Versioning) redirector described in the "Security" section of this chapter. You can find an excellent resource on this topic in the Internet Information Services (IIS) 6.0 Resource Guide available from the Microsoft web site's IIS 6.0 Resource Center at `http://support.microsoft.com/default.aspx?pr=iis60`.

Internet Group Management Protocol (IGMP) v3

Windows Server 2003 supports IGMP (Internet Group Management Protocol) v3 (currently an Internet draft). Some of IGMP v3's features include

- Support for source filtering, which allows multicast traffic only from a specific source address or addresses.
- New IGMP extensions.
- Prevention of denial of service (DoS) attack by a rogue server by configuring multicast routers to not forward multicast traffic outside of specified networks.

Other Enhancements and Changes

Other networking enhancements include improvements for wireless LAN security, TCP/UDP (User Datagram Protocol) port ownership, removal of support for some modems and network adapters, and improvements in IPSec monitoring.

Secure Wireless LANS

Windows Server 2003 provides security and performance improvements for wireless LANs, such as automatic key management and user authentication and authorization prior to LAN access. It will also provide access control for Ethernet networks when wired Ethernet is used in public locations.

TCP/UDP Port Ownership

A new NETSTAT option displays the process that owns the (TCP/UDP) port. An Administrator can use this feature for configuring secure servers, security audits, and performance improvements.

Modems and Network Adapters No Longer Supported

Microsoft has dropped support for a number of modems and network adapters in Windows Server 2003, largely due to the respective vendors not supporting them anymore. They fall into five classes:

- Home phone line network adapters
- 10Mb-only Ethernet adapters
- End of Life 10/100 PCI (Peripheral Component Interconnect) adapters
- End of Life Wireless PCMCIA (Personal Computer Memory Card International Association) adapters
- Other adapters that vendors do not support in Windows Server 2003

Details on the specific adapters that are not supported are available in Microsoft KB article 320892 "List of unsupported modems and network adapters in Windows Server 2003."

IPSec Monitoring Improvements

Windows Server 2003 includes the IPSECMON MMC that replaces the ipsecmon.exe monitor program found in Windows 2000. IPSECMON MMC includes all the features of ipsecmon.exe, but also includes RSoP data and contains the logging mode and planning mode, just like any other policy. With RSoP, the IPSec policies can be analyzed for application of the policy, and settings applied by using the logging mode. The planning mode allows "what if" scenarios—enabling you to configure a policy and then get a report on what effect it had.

Security

There probably hasn't been a time in the modern era of computing where there was more widespread concern for security. Only a few years ago, security was one of those dark corners of the industry for hard-core types, but largely ignored by most of us. Today, securing computing resources ripples from the largest corporation to the most computer-illiterate home consumer, trying to prevent sophisticated hackers not only from infecting their computers with destructive viruses, but also from stealing information or using their computer as an ad hoc server.

note

Jan de Clercq of HP's Security Office provided much of the technical information and technical review of this section. Jan is one of HP's leading security experts. I highly recommend his new book, *Windows Server 2003 Security Infrastructures*, Digital Press, 2004.

Windows 2000 made some huge strides in securing the computing environment with features such as Kerberos for authentication and authorization, IP Security (IPSec) for remote access security, and the Encrypted File System (EFS) for protecting sensitive data. Windows NT and 2000 provided Certificate Services, which gave Administrators the ability to install their own CA and secure the certificates in their enterprise inexpensively, enticing smaller companies to implement Public Key Infrastructures (PKI).

Security issues always provide room for improvement, and Windows Server 2003 and Windows XP have taken a step forward in a number of areas. Think of XP as a Windows 2003 client—they both have very similar and compatible features. Significant security feature improvements in Windows XP and Windows Server 2003 include

- Software restriction policies
- Internet connection firewall

- Personal firewall
- EFS enhancements
- IPSec enhancements
- Credential Manager
- PKI improvements
  - Cross-certification trust model
  - Editable certificate templates (v2)
  - Key recovery
- Enhanced Security Management—Effective Permissions tab
- User Autoenrollment
- Kerberos and forest trusts

The following sections provide a brief description of each issue. Refer to Jan de Clercq's books listed in the "References" section at the end of the chapter for more information.

Software Restriction Policies

Also known as SAFER, these policies are Microsoft's first attempt to provide stronger cryptographic compliance with Federal Information Processing Standards (FIPS) 140-1. FIPS determines whether products meet the standard and evaluates overall security provided by a cryptosystem. The kernel mode driver, FIPS.SYS, supports EFS (efs.sys), IPSec (ipsec.sys), and other crypto functions. Although this is only a level 1 compliance intended for general PC use, it is nonetheless a start. The idea with SAFER policies is to provide the Administrator the ability to grant granular client-level security policies to restrict the access to read, write, or execute software on computers.

Because the "How To" basics of configuring these policies are reasonably well documented in the Microsoft KB 324036 article "How to use Software Restriction Policies in Windows Server 2003," I'll use the space here to describe some important concepts and implementation recommendations.

Software Restriction Policies is implemented through GPO settings located in the GPO tree `Computer Configuration\Windows Settings\Security Settings\Software Restriction Policies`, and in `User Configuration\Windows Settings\Security Settings\Software Restriction Policies`. When you drill down to this location on a new policy, a warning appears in the right pane of the GPO Editor, "No Software Restriction Policies Defined," as shown in Figure 1.23. Right-click on the Software Restriction Policies in the left pane and click

New Software Restriction Policies. You will then see two folders titled Security Levels and Additional Rules. Security Levels defines a high-level access policy, whereas Additional Rules allows you to define specific rules, including the following:

- **Path rules**: Allow or deny access to files specified by a directory path on the local PC. Also available is a Registry path. Using the Registry path to identify files to allow or deny access to is more reliable. For instance, if you deny access to `C:/applications/abc.exe` to prevent users from running the ABC program, but the user installs it on `D:\myapps\abc.exe`, the policy would fail. The Registry path for any given application will be the same no matter where the files are stored on disk. This can also be used to disallow access to a known file name that is installed by a virus.
- **Hash rules**: The Administrator can generate a hash of a file and set up a hash rule to compare the original hash to a hash of the existing file. If they differ, access is denied. This rule is intended to allow users to run or prevent them from running certain versions of a program. This could also be useful in preventing users from accessing files infected with a virus. The hash is not affected by file rename or relocation.
- **Certificate rules**: These rules can be used to allow or disallow access to a digitally signed program. A comparison is made via user certificate to determine access level.
- **Internet Zone rule**: As of this writing, this rule is pretty useless in my opinion. Its intent is to let you create a rule to prevent users from installing software from sites in a given Internet Zone defined in Internet Explorer (IE). Unfortunately, it applies only to downloading .MSI files—not a big threat since there are so many other file types: .exe, .zip, etc..

Security Levels

Think of Security Levels as the locked condition on the front door to a building. There are two options (look in the Security Levels folder in the GPO): Disallowed and Unrestricted. The Disallowed option locks the front door and requires all who enter to provide proper credentials. No public access. Employees with ID badges only. The Unrestricted option leaves the front door unlocked, allowing anyone to enter the building. Inside the building are special rooms (programs and files on the PC) that also have locks on the door. These locks are the four classes of rules described in the previous section. If the front door is Disallowed, then you don't get access unless you can prove you have access to a room in the building. The guard escorts you to that room (or rooms), but you can't go anywhere else. This obviously is a high-security condition. If the front door is set to Unrestricted, anyone can walk in and roam through the building and enter any room that is not locked. However, if a room is locked (that is, a rule has been created to restrict access to a program), you have to be granted specific access to get in. I recommend, as does Microsoft's KB 310791 article "Description of the Software Restriction Policies in Windows XP," you use the Unrestricted option because it simply locks out certain files and programs. Unrestricted is the default. Setting the Disallowed mode, unless it is in a very controlled environment, will generate a lot of help desk calls if it's not well planned and tested.

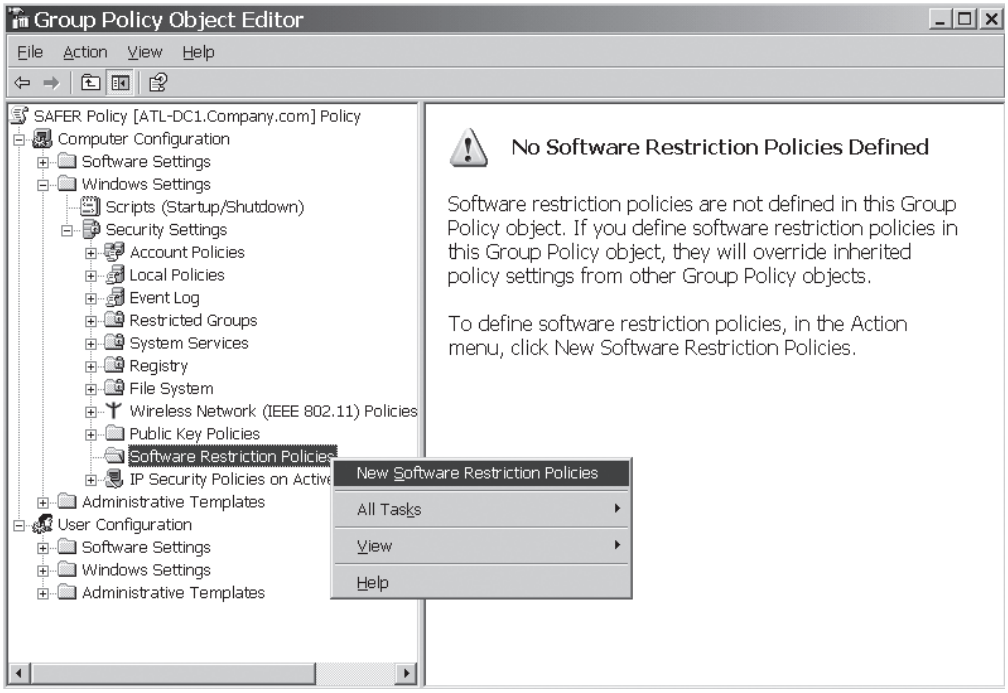

Figure 1.23 Defining Software Restriction Policies in the Group Policy Editor.

Additional Rules

These are the keys to the rooms—the exceptions to the lockout. Note that four Registry path policies are created by default in the Additional Rules folder. Figure 1.24 shows the default Registry path rules, as viewed in the GPMC. They are all defined as Unrestricted and are applied to

```
%systemroot%
%systemroot%\*.exe
%systemroot%\system32\*.exe
%programfilesdir%\
```

The purpose of these policies is to permit access to critical directory space when you've set the Security Level to Disallowed. These policies allow access to areas like c:\> Program files.

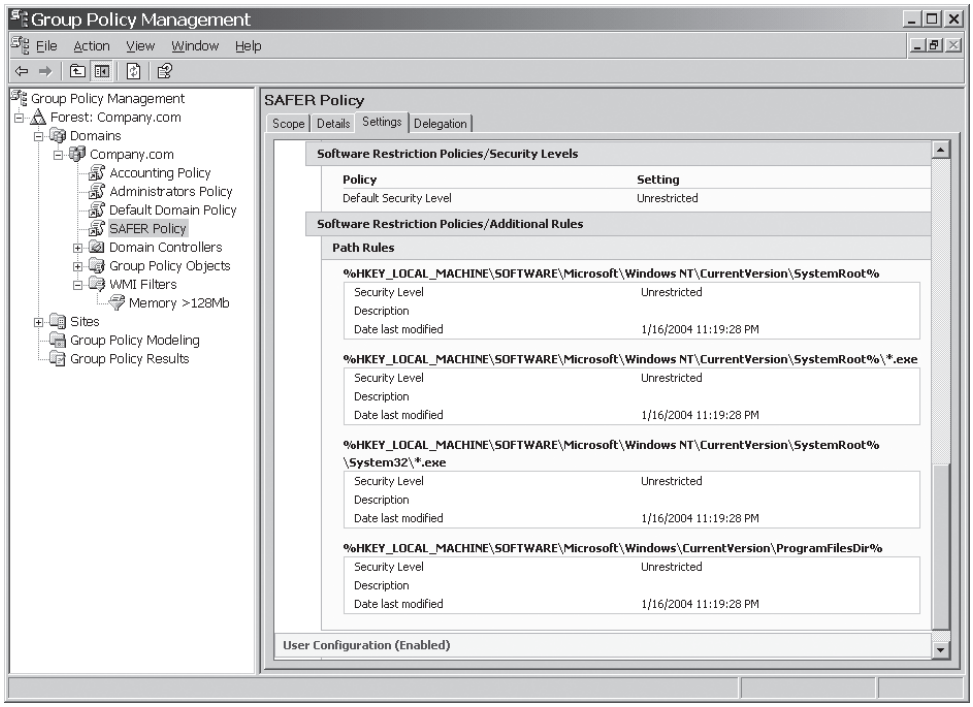

Figure 1.24 The additional rules defined by default in Software Restriction Policies are shown using the GPMC snap-in.

Implementation

So far these rules seem fairly straightforward. If you want to put the admin pack on every PC, but restrict users from running it, you set the SAFER policy to Unrestricted, and then create a path policy to restrict domain users from executing it. Pretty simple until you get multiple rules defined in multiple policies and perhaps in a combination of User and Computer Configuration settings.

note

The description of functionality described here is based on my experience and testing and asking questions of Microsoft PSS. I have not seen any Microsoft documentation to confirm or deny these assertions. Your mileage may vary, so be sure you test these policies before implementing.

If multiple Software Restriction Policies are defined, things get complicated. Several design rules apply to the application of these policies. Understanding them is critical to obtaining the desired results.

- Closest match applies the rule. That is, if Rule 1 allows access to c:*.exe and Rule 2 denies access to C:\payrollapp.exe, then the user would be unable to run payrollapp.exe because Rule 2 is a closer match to the program being run.
- Most restrictive rules apply. Apply most-restrictive path rules further down in the directory tree than less-restrictive ones.
- Rules are processed in the following order:
  1. Security Level (highest priority)
  2. Hash
  3. Certificate
  4. Path
  5. Internet Zone rule
  6. Default rules
- Software Restriction Policies accumulate from multiple GPOs and are evaluated together. They do not obey normal SDOU (Site Domain Organizational Unit) processing of GPOs, and they can be applied at the Computer and User Configuration sections of each GPO.
- Security Levels determine how the Additional Rules are defined. If the Security Level is Unrestricted, the Additional Rules should be designed to restrict access. If the Security Level is Disallowed, the Additional Rules should be designed to allow access.
- Software restriction policies—especially when setting general categories (such as entire directories) to be disallowed—can have unexpected results. For instance, disallowing access to *.vbs to prevent virus deposited files from running can also restrict access to logon scripts and other valid scripts.
- Always thoroughly test these policies.
- Define the policies in individual GPOs. This makes them easier to manage and troubleshoot.
- Avoid applying Software Restriction Policies from multiple locations (domains and OUs) if possible. Applying them from fewer locations makes the results easier to predict and troubleshoot.

- To restrict Administrators from a Software Restriction Policy defined in User Configuration, restrict the appropriate Administrators' accounts and/or groups by removing the Apply Group Policy directly on the GPO.
- To restrict Administrators from a Software Restriction Policy defined in Computer Configuration, you can remove the Apply Group Policy directly from the appropriate user or group, or you can go to `Computer Configuration\Windows Settings\Security Settings\Software Restriction Policies\Enforcement`. In the Enforcement Properties page, under the Apply Software Restriction policies to the following users, select All Users Except Local Administrators.
- Make certain you have not inadvertently restricted access to valid files such as logon scripts and directories and files that should be scanned by antivirus programs. Testing should expose these types of flaws.

Evaluation of these policies at this time seems like a bit of black magic. As of this writing, Microsoft had not seen a lot of customers using these policies, but in the work I've done with the policies, I've found they are difficult to design and it's difficult to get the desired results if you have a lot of them. Again, adding complexity to these policies results in a high probability of failure, so you must test them thoroughly. You can apply them to User Config and Computer Config on multiple levels (domain and OU), and then have them all sorted by rule precedence, closest match, and most restrictive. Keep these GPOs in as few domains and OUs as possible, keep the rules simple and well planned, and avoid applying them to both User Config and Computer Config in the GPO.

Sample Rules

Let's look at a couple of examples of rules. Suppose we have six policies defined in a Domain SAFER Policy defined at the domain:

- Security Level: Unrestricted
- Hash Rule 1: hash of Payroll.exe (v4.2) set to Unrestricted
- Hash Rule 2: hash of Payroll.exe (v4.1) set to Disallowed
- Certificate Rule 1: Applications Certificate set to Unrestricted
- Path Rule 1: C:\RestrictedApps set to Disallowed
- Path Rule 2: C:\RestrictedApps*.exe set to Unrestricted

Using these rules, a user attempting to run the following programs would have these results:

- User runs the program: C:\restrictedApps\Payroll.exe (v4.1).

  Security Level set to Unrestricted allows access.

  Path Rule 1 allows access.

  Hash Rule 1 disallows access.

Certificate Rule 1 applies because this app was digitally signed by the certificate owned by the Applications group and is set to Unrestricted.

Result: Hash Rule 2 applies because it is most specific (denies access to the exact file name, whereas the others refer to directories or groups of files). Access is granted.

- User runs the program: C:\restrictedApps\backupfiles.exe.

  Security Level set to Unrestricted allows access.

  Path Rule 1 disallows access to C:\restrictedApps.

  Path Rule 2 allows access to C:\restrictedApps*.exe.

  Result: Because Path Rule 1 denies access to the directory, Path Rule 2 never applies. Access is denied. Remember that the most-restrictive path rules should be applied deeper in the tree than the least-restrictive ones.

Suppose the same rules are defined, but the Security Level is set to Disallowed. Path Rule 1 and Hash Rule 2 are not needed because the Security Level is set to Disallowed. With that setting, all you need to do is define rules to allow access to programs that you want the user to access.

Microsoft has attempted to help with some exceptions to these rules. In the Software Restriction Policies folder in the GPO, you'll see three policies besides the two folders. These policies are

- **Enforcement:** Allows you to exclude .dlls from restricted programs, and to exclude local admins.
- **Designated File Types:** By default, policies only apply to normal executable types, such as .exe, .dll, and .vbs. This option allows you to add other types to it, such as .bat, .cmd, .msi, and others, including a blank so you can add one that isn't in the list.
- **Trusted Publishers**: Defines users who can choose trusted publishers (for certificates).

In summary of Software Restriction Policies, I'd say to use them very carefully and only after extensive testing so you understand what the ramifications are. They have some value in the security configuration of the enterprise, but they are complex and can create a lot of problems if you aren't careful.

Some good resources for SAFER policies include the Microsoft whitepaper, and Jeremy Moskowitz's *Group Policy, Profiles and IntelliMirror for Windows 2003, Windows XP, and Windows 2000*, Sybex, March 2004.

Internet Connection (Personal) Firewall

Also referred to as the Personal Firewall, this feature is provided with Windows XP as built-in protection for each computer. It isn't the most secure firewall, but it's probably sufficient for casual home users. If firewalls are implemented elsewhere, the personal firewall in XP is usually required to not be enabled. To enable the Internet Connection Firewall, open a network connection's Properties page, go to the Advanced tab, and select the Protect My Computer and Network by Limiting or Preventing Access to this Computer from the Internet check box. Click the Settings button to access the Advanced Settings pane, shown in Figure 1.25 to configure advanced settings, including

- **Services:** Allows you to select services such as FTP, Telnet, HTTP, and Remote Desktop that the users of this computer are allowed to access. Default is no access. So if you turn this firewall on and can't get to a Web site, the reason is probably wasn't enabled.
- **Security Logging**: Allows configuring options to log dropped packets and successful connections (or not), to specify a log file name and location (default is `%systemroot%\pfirewall.log`), and to set a log file size limit.
- **ICMP**: Permits configuring of ICMP traffic, such as Allow Incoming Echo Request, Allow Outgoing Search Request, Allow Incoming Router Request, and other options. By default, all are disabled.

note

Windows XP Service Pack 2 enables the Internet Connection Firewall by default.

EFS (Encrypting File System) Enhancements

Although Windows 2000 EFS brought a level of security to local files, it had a number of deficiencies:

- Stealing a local account password is easy using common hacker tools in standalone mode.
- Encrypted files stored on file servers are decrypted on the server and then transported in clear text across the network to the user's workstation. Because EFS needs access to the user's private key, which is held in the profile, the server must be "trusted for delegation" and have access to the user's local profile.
- Inability to share EFS encrypted files.

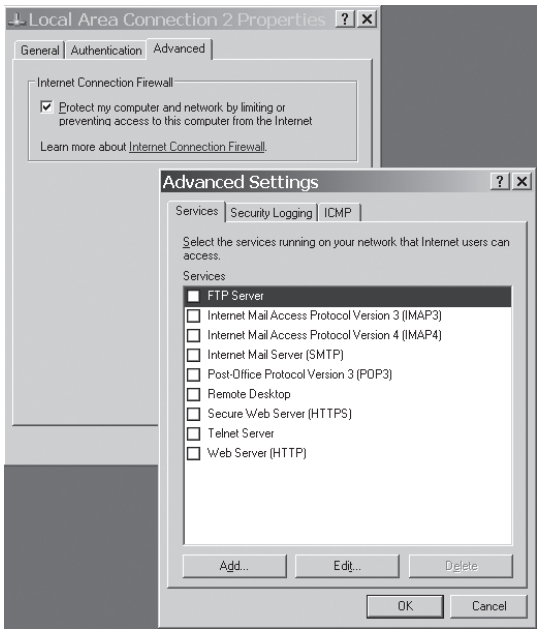

Figure 1.25 Internet Connection Firewall Advanced Settings enable you to define what the user *is allowed* to do.

EFS has been enhanced in Windows XP and Windows Server 2003 to support sharing of encrypted files. Windows 2000, Windows Server 2003, and Windows XP all store the EFS metadata in the NTFS. Thus, encrypted files on NTFS volumes in Windows 2000 and Windows Server 2003. Sharing an encrypted file is enabled by opening the file's Properties page, clicking the Advanced button, and selecting the Details button in the Advanced Attributes area. You can add users who you want to share the file with to the list.

Note that EFS file sharing is set at the file level—not the folder level——and inheritance of EFS file-sharing metadata is not supported.

Another option for sharing encrypted files in Windows Server 2003 is to use Web folders. Windows Server 2003's support for the transport of EFS metadata using the WebDAV protocol, an extension of HTTP 1.1, makes Web folders an excellent alternative to sharing files on file servers, and is recommended by Microsoft. WebDAV is supported on IE v5.0 on the client and IIS v5.0 and later on the server. For more information about WebDAV, see the WebDAV Resources FAQs at `http://www.webdav.org/other/faq.html`.

To configure a Web folder on the server running IIS, go to the properties sheet of the encrypted file (or any file), go to the Web Sharing tab, and select the Share this Folder option. Table 1.5 provides a side-by-side comparison of features between the Web folders and File Shares.

Table 1.5 Comparison of EFS Features Between File Shares and Web Folders

Remote EFS Operations On . . .	. . . File Shares	. . . Web Folders
Where does EFS encryption/decryption occur?	Files are encrypted and decrypted on the file server.	Files are encrypted and decrypted on the user machine.
Are the files secured during transmission over the network?	Files are sent in the clear over the network connection.	Files remain encrypted while being sent over the network connection.
What technology is or can be used to secure the transmission of the files over the network?	Requires IPSec to secure the file transfer between file server and user machine.	Does not require IPSec to secure the file transfer; relies on the WebDAV EFS extensions to securely transmit the file.
Must the file server be "trusted for delegation?"	Requires file server to be "trusted for delegation."	Does not require file server to be "trusted for delegation."
Does the solution require a copy of the user profile on the file server?	Requires availability of user profile on the file server (local or roaming profile).	Does not require availability of user profile on the file server.
Where does the EFS file-sharing authorization process take place for users?	EFS checks for other user certificates on the file server and/or in the AD.	EFS checks for other user certificates on the local machine and/or in the AD.

(Table reprinted by permission from Jan de Clercq, Hewlett-Packard Company.)

Credential Manager

Single Sign On (SSO) capability has been a frequent request from users and Administrators who find it challenging to keep track of all the credentials needed to log on to the domain, or use Microsoft Passport credentials and Smart Card credentials. Managers who must keep getting validated to perform administrative tasks on various computers also have requested this capability. Windows Server 2003 introduces Credential Manager, a client-based SSO solution that uses an intelligent credential-caching mechanism that keeps credentials in a store on the client, requiring the user to provide a single username/password to open the store. Credentials

in the store can consist of a user account and password; a user account, certificate, and private key (which can be stored on a smart card); or Microsoft Passport credentials.

The credential store is part of the user profile and supports roaming and can optionally be disabled by the Administrator via the GPO. This is done by opening a GPO and going to `Windows Settings\Security Settings\Local Policies\Security Options` and enabling the option `Network Access: do not allow storage of credentials or .NET Passports for network authentication`.

This feature is available in Windows Server 2003, Windows XP Pro, and Windows XP Home operating systems.

Public Key Infrastructure (PKI) Improvements

Microsoft provided an out-of-the-box, user-defined and administered CA in the Windows NT 4.0 Option pack. This might not have been the most effective and robust product on the market, but it was cheap (and was included at no extra cost with Windows 2000). This allowed small- and medium-size companies, who couldn't afford the high cost of third-party companies such as VeriSign and Entrust, to fortify their security systems. However, according to one of HP's security experts, consultant Jan de Clercq, "ease of use and security don't easily coexist." But considering the improvements made in Windows Server 2003 security features, especially in the PKI arena, Jan stated that "Given the low cost and the advanced PKI features of Windows.NET, it may be a product that will bring PKI to the masses." Let's see what features are now available to make this possible.

Cross Certification Trust Model

Windows 2000 used a hierarchical CA trust model, shown in Figure 1.26, whereby the parent (root) CA was linked to the child (subordinate) CA by trusts. Only one root CA can be in the forest, and the only way to configure it to trust certificates issued by other CAs in other forests was to reconfigure the entire CA structure using Certificate Trust Lists. This might be desirable in multiforest enterprises or to link extranets.

Note that the CA hierarchy is not specifically tied to the domain hierarchy. Rather, it might more appropriately be associated with geographies of the enterprise to provide accessible CAs to users and computers in each geography.

Windows Server 2003 introduces the Cross Certification trust model, which allows CAs in different organizations (forests) to trust each other and is in addition to the subordinate trust model. The Cross Certification model permits the Administrator some freedom in configuring the trust relationships with the extranet partners, by specifying whether it is a one-way or two-way trust, and by controlling the issuance and usage scope of the CA agreement via naming constraint, application, and issuance policy rules.

Cross certification provides the ability to create transitive cross forest trusts, as described in the "Creating a Cross Forest Trust" section in this chapter.

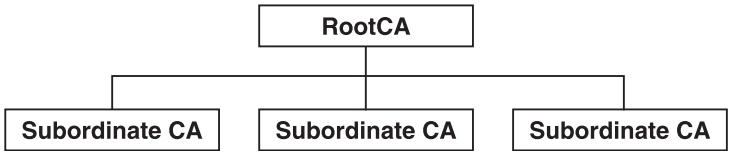

Figure 1.26 Hierarchical CA structure with the root (enterprise) CA at the top and subordinate (issuing) CAs below.

Editable Certificate Templates (v2)

Windows 2000 provided v1 certificate templates that were not editable. Windows 2003 provides v2 templates that are editable and include a mechanism to modify v1 templates and convert them to v2. The v2 templates facilitate cross certification and policy enforcement, which allows configuration of certificate trust relationships across forest boundaries. This could be trusts to CAs in other forests within the company or to a CA in a business partner's extranet. This also allows a CA to issue certificates to a CA in another forest and can link a policy to this cross certification. The policy can limit the issuance and usage scope of the CA cross-certification agreement.

A new Certificate Templates snap-in is included in Certificate Services for Windows Server 2003, as shown in Figure 1.27. Note that the icons for v1 and v2 templates are a different color, the *Minimum Supported* column identifies v2 templates as Windows 2003 and Auto Enrollment functionality is listed. This snap-in is used to modify Version 2 templates to perform the following functions:

- Create a new template, modify an existing template, or duplicate an existing template.
- Modify template properties such as certificate lifetime, renewal period, whether it is published in AD, Issuance Requirements such as re-enrollment, extensions, and other properties.
- Define accounts that can enroll and autoenroll a certificate template.
- Enable a template for Autoenrollment.
- Set which accounts can enroll and auto-enroll for a particular certificate template. This is done by right-clicking on a template, selecting its properties and modifying the template's Access Control List (ACL) in the "security" tab. Windows.NET ACLs contain an Access Control Entry (ACE) for "Enroll" and "AutoEnroll."

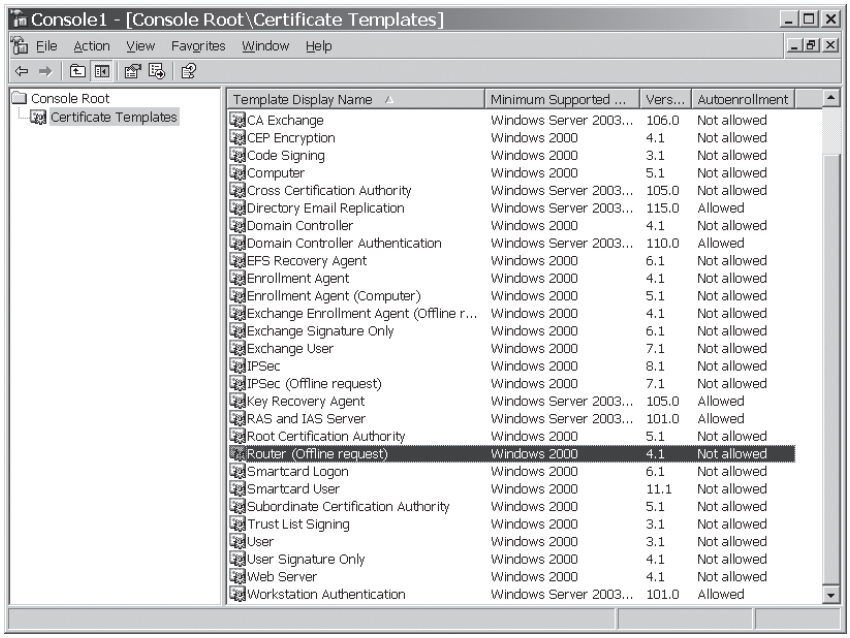

Figure 1.27 Certificate Template snap-in.

note

> You can copy and save a v1 template as a v2 template and then configure it as desired. You also can select the Duplicate Template option when you right-click on the template in the Certificate Templates snap-in, which will make a duplicate and give you a head start on creating a similar one.

Private Key Recovery

Windows Server 2003 PKI made significant changes to the key recovery process, providing centralized key recovery services. An important component to this process is the Key Recovery Agent—usually one or more Administrators who have the ability to recover encrypted files, by means of a special public key stored in an EFS Recovery Agent certificate. The key recovery data is stored in the CA database. A user's archived private key is encrypted using a symmetric key, which is encrypted using a Key Recovery Agent's public key. These keys are stored in the CA database in the `RawArchivedKey` and `KeyRecoveryHashes` columns, respectively.

To recover a user's private key, an Administrator must be added to the Recovery Agent list. The preparation for the Key Recovery process includes the following steps:

1. Issue the EFS Recovery Agent certificate to an Administrator.
2. Create a GPO called EFS Recovery, for example.
3. In the GPO, go to `Computer Configuration\Windows Settings\Security Settings\Public Key Policies\Encrypting File System`. Define an EFS Policy. Select the Add a Recovery Agent option and add a user account that has the EFS recovery certificate (from step 2).
4. In the Certificate Authority snap-in, in the RootCA properties, select the Recovery Agents tab. Select the Archive the Key option and select the Add button. Select the Key Recovery Agent certificate in the next dialog box.

note

Multiple Administrators can be designated as Recovery Agents so that both public keys are required to recover private keys for an added measure of security.

This four-step process will recover a private key in Windows Server 2003:

1. The Administrator who has the Recovery Agent rights (key) must know the User Principal Name (UPN) or serial number of the certificate whose private key he wants to recover.
2. From a command prompt, run the command:

   ```
   certutil –getkey <serial number or UPN> <outputfile>
   ```

   This exports the recovery data from the CA database.
3. From a command prompt, run the command:

   ```
   certutil –recoverkey <outputfile> <pkcs12file>
   ```

   This will convert the output file in a PKCS12 format and could store it on a floppy disk.

4. The PKCS12 file is provided to the appropriate user, who can then import it in his certificate store.

Enhanced Security Management—Effective Permissions Tab

The Effective Permissions tab displays the cumulative permissions calculated from group membership and any inherited permissions, and displays the result applied to a given user. This is part of the Advanced Settings on the ACL properties page. For instance, right-click on an NTFS folder, go to Properties, select the Security tab, and then select the Advanced button. In the Effective Permissions tab, shown in Figure 1.28, you see a cumulative summary of all permissions applied to that user on that object.

User Autoenrollment

Windows 2000 Certificate Services provided a valuable feature in the autoenrollment and renewal of machine certificates. This allowed DCs to obtain and renew certificates. Windows Server 2003 now provides autoenrollment of users. You can do this in a Certificate Template by going to the Properties of the template, and then on the Security tab, giving Autoenroll the READ privilege. You also can enable it via Group Policy, as shown in Figure 1.29. In `Computer Configuration\Windows Settings\Security Settings\Public Key Policies`, check the Enroll Certificates Automatically option.

Kerberos and Forest Trusts

Windows 2000 was unable to create a transitive trust between forests, allowing only a one-way NTLM type trust that had to be created between individual domains in different forests—much like what we had to do in Windows NT to build trusts between all domains. Interestingly enough, this actually has a side benefit—because this type of trust can't be used by Kerberos, it solves a migration problem known as the "Pile On" issue, described in Chapter 3. Other than that, an NTLM trust isn't a good thing because it requires the Administrator to manage a lot of trusts between forests.

Windows Server 2003 includes schema modifications made to the trusted domain object (TDO), which allows Kerberos to authenticate across the domain. This provides transitive trusts to be built across forests. That is, we create a trust between two forests, (one way or two way) and no other trust is needed. The Administrator does have some control over the trust, in making it a one-way trust or a two-way trust, and determining the default level of authentication that is allowed between forests (described in more detail in the next section).

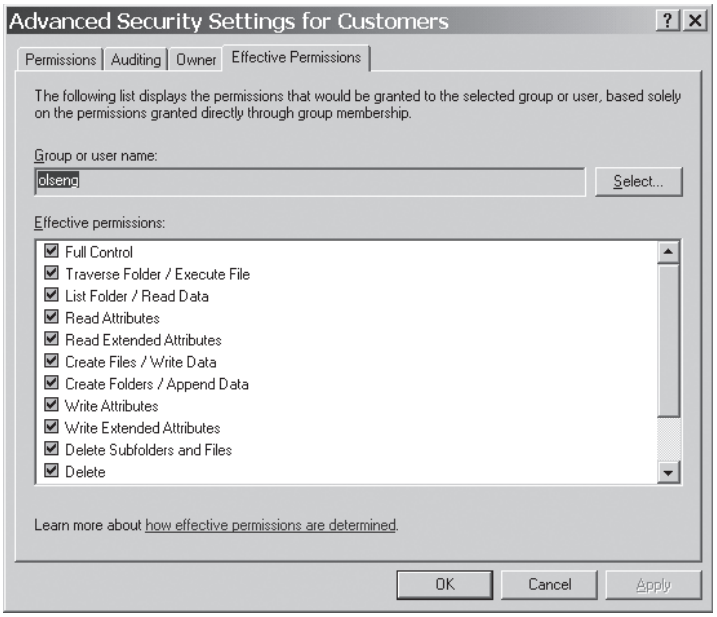

Figure 1.28 Effective Permissions tab.

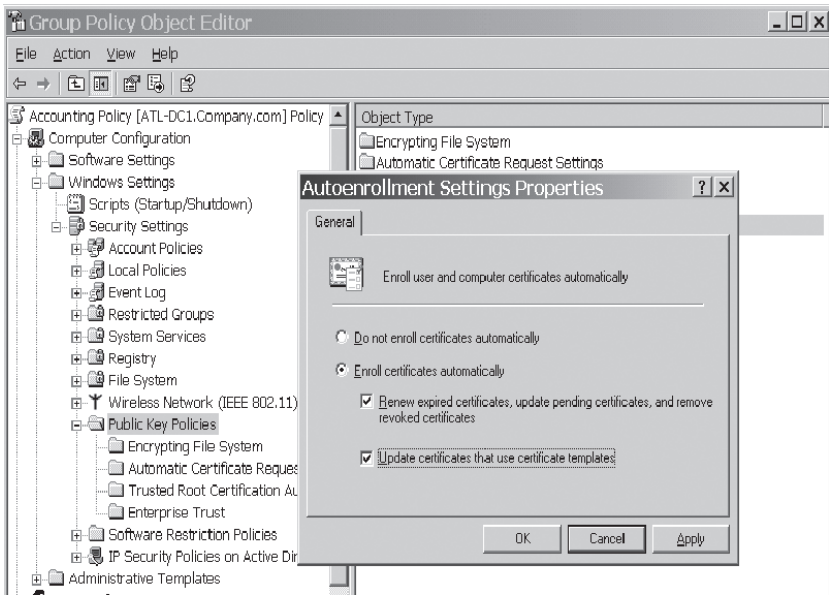

Figure 1.29 Using Group Policy to enable autoenrollment of users.

Creating a Cross Forest Trust

The preparatory steps to creating a cross forest trust include establishing DNS name resolution between the two forests, synchronizing system time between all computers in both forests (easier than it sounds), and setting the forest functional level to Windows 2003 (native). Windows Server 2003 provides three different ways to establish DNS name resolution between the two forests:

- Create a stub zone in the forest root of each forest, for the other forest.
- Create a secondary zone in the forest root of each forest, for the other forest.
- Set up a conditional forwarder on the DNS server that is authoritative for each root zone for the other forest.

Test name resolution by pinging the root domain name of each forest from the other forest. In addition, you must ensure that the system time of all computers in both forests is within the allowable Kerberos time skew (default is five minutes). If the time skew between the two DCs contacted to build the trust is more than the allowable time skew, the trust might be created successfully, but authentication will fail.

note

There is no default time synchronization between two forests. Because Kerberos authentication, by default, requires the system time of all computers to be within five minutes of each other, ensure the time between forests is synchronized. It is recommended that you configure the PDC emulator of the root domain in each forest to synchronize with the same external time source. Time synchronization is configured by default for a forest, but not between forests. More information is available in the "Time Services" section of Chapter 6.

Setting the forest functional level is described in Chapter 3. To do this, all DCs in every domain must be Windows 2003 servers, and the domain functionality must be set to Windows Server 2003 (referred to as native, though the word "native" is not used in the UI). After the domains are all raised to Windows 2003, the forest must be raised to Windows 2003 native—similar to the way Windows 2000 domains were raised to native mode, but Windows 2000 didn't have a forest native mode.

After the forest is at Windows 2003 level, Time Services are synchronized, and DNS name resolution is established between the two forests, the trust can be created. You create the trust just as you did in Windows 2000 by going to the Active Directory Domains and Trusts snap-in, choosing Domain Properties, selecting the Trusts tab, and clicking New Trust. This engages the New Trust Wizard. Some of the options specified in the wizard are described here, along with screen shots of the dialog boxes used in the wizard.

- **Trust Type**: Can create an External Trust (downlevel, nontransitive trust) between domains, or a Forest Trust (transitive, Kerberos) between forests. The dialog box is shown in Figure 1.30.

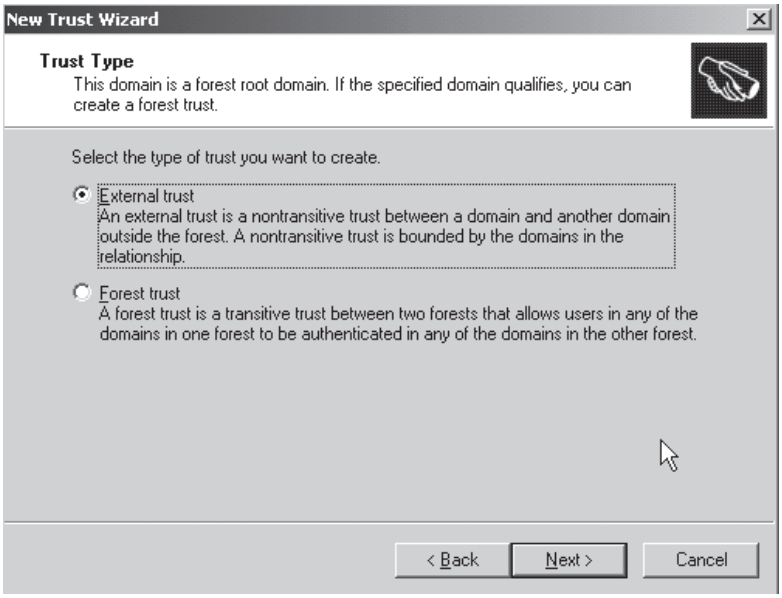

Figure 1.30 The New Trust Wizard allows you to create an External (NTLM type) Trust or a Forest Trust (Kerberos transitive trust).

- **Trust Direction—Incoming**: Users in the source domain can be authenticated in the target domain or forest (source domain is the trusted domain). The dialog box is shown in Figure 1.31.
- **Trust Direction—Outgoing**: Users in the target domain or forest can be authenticated in the source domain or forest (source domain is the trusting domain).
- **Trust Direction—Two Way:** Outgoing and incoming trust. Users in source and target domains or forests can be authenticated in each domain.
- **Authentication Level—Forest-wide Authentication:** Users from the other forest are automatically authenticated in the local forest. Essentially this makes the authenticated users in the other forest included in the authenticated users in the local forest. This is appropriate for a company with a multiple forest deployment, and users from both forests are equally trusted for resources in each forest. They can be denied access by changing permissions just like a single forest implementation. This is defined for the ingoing and the outgoing trust separately. The dialog box is shown in Figure 1.32.

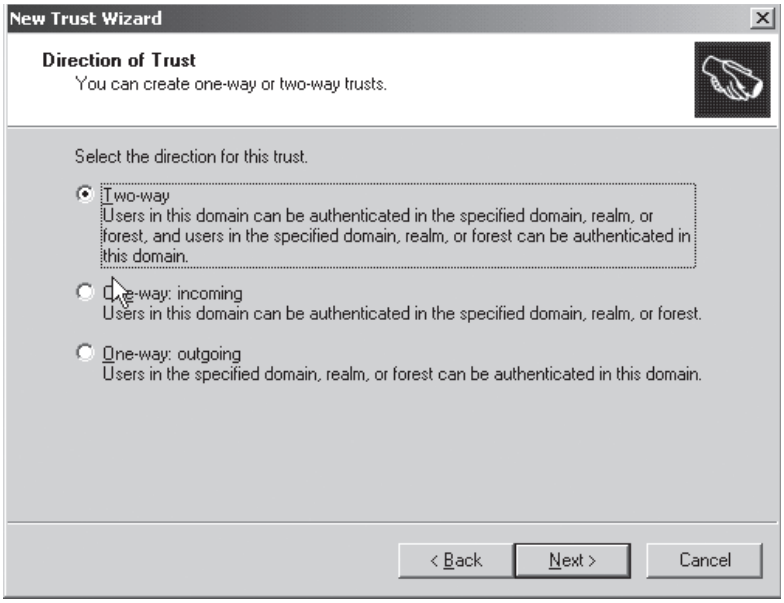

Figure 1.31 The Trust Direction dialog box lets you specify whether the trust is incoming (trusted), outgoing (trusting), or two–way.

- **Authentication Level—Selective Authentication**: This scope requires users to be specifically trusted to access any resource. This is appropriate for enterprises that have business partner extranets as separate forests and want to grant limited and specific access. This is defined for the incoming and the outgoing trust separately.

The wizard also gives you the ability to choose to verify the trust. I recommend that you do so. After the trust is created, you will be able to see users and groups of the other forest in the object picker to assign permissions to resources. For instance, in Figure 1.33, we have a share on a server in forest CorpB.Net, and in assigning permissions, the users and groups of forest CorpA.Net are available. Note that all you see in the locations option of the object picker is the remote forest with a triangle-like icon. Child domains in that forest, if any, are not exposed. However, entering user or group names (including the use of wild cards) in the object picker will find objects in any domain in the remote forest. For instance, if you entered **Admin** as the name in the object picker, it would return the Administrator account for all domains in the remote forest.

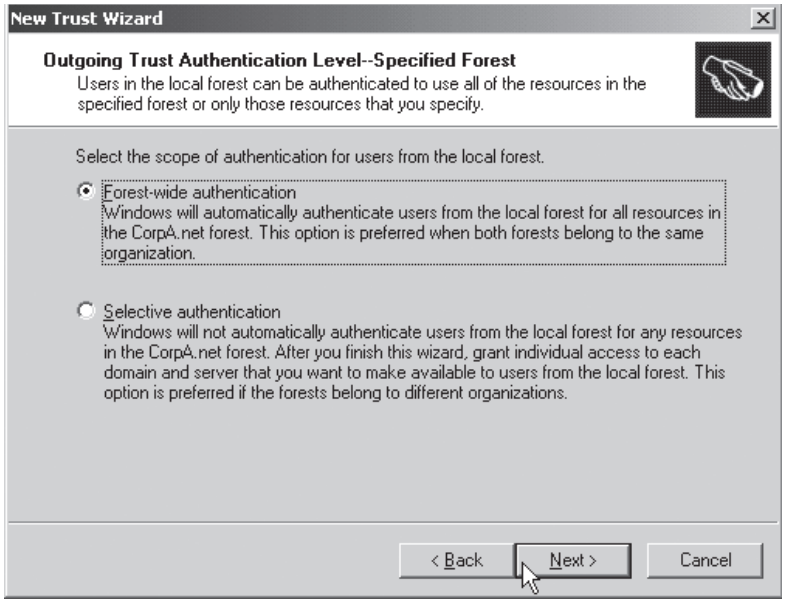

Figure 1.32 Authentication scope options allow you to specify the default authentication between two forests.

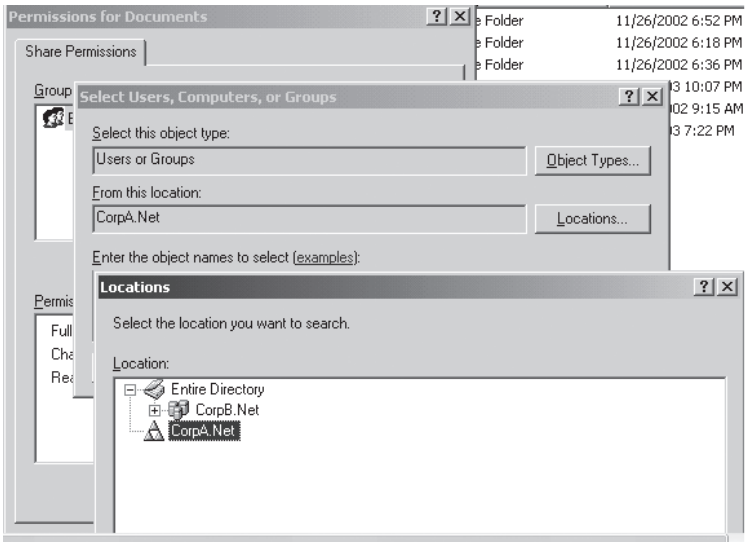

Figure 1.33 The object picker displays the trusted forest for assigning rights.

The cross forest trust will be a valuable benefit to enterprises with multiple forest deployments. Where Windows 2000's NTLM trusts have made multiple forests prohibitive to manage, Windows Server 2003's cross forest trusts could easily make the multiple forest configuration feasible.

Conclusion

This is by no means a comprehensive dissertation on security or even Windows 2003 security, but it does give a good overview of the many improvements provided in Windows Server 2003. Hopefully, after reading these issues, you will have some area you want more information on that might affect your Windows infrastructure and should be included in the migration plan. I have the good fortune of getting much of this information from HP's Jan de Clercq, a recognized security expert and author of two books regarding PKI and other security issues in the Windows environment. I highly recommend Jan's latest book, *Windows Server 2003 Security Infrastructures,* Digital Press, 2004.

Windows Management Instrumentation (WMI)

WMI was available in Windows 2000, but there was no user-level interface. Programs had to be created to take advantage of the data exposed by WMI APIs. Windows XP and Windows Server 2003 provide an interface called WMIC. WMI is a very powerful tool that has endless possibilities, and with WMIC, it is readily available. For a quick introduction to WMI and WMIC, go to a Windows XP client (perhaps your laptop or office desktop), go to a command prompt, and enter **WMIC**. You will get a message that WMIC is being installed and it might take a minute or two. You can now execute WMI commands that heretofore required a program to call the APIs. After WMIC installs, you'll get a `wmic:root\cli` prompt (referred to here simply as "the WMIC prompt"). Here are some introductory commands you can play with:

- **/?**: Online help (many screens of these; paused at each page).
- **OS**: Operating system info (name, build, and so on).
- **NIC**: Network Interface Controller info (adapters, type, version, and so on).
- **NTDOMAIN**: Domain information (domain name, site name, DC name and address).
- **PAGEFILE**: Pagefile data (current usage, location, peak usage).
- **IRQ**: IRQ assignments, including a status of each.
- **QFE**: Be careful! This lists all the hotfixes, security updates, and so on installed on the system, which is quite a lengthy list on most systems.

Beyond the basics, there are WMI service providers that permit you to gather information about the Windows infrastructure, such as AD and DNS. You can execute some fairly complex commands to extract information. My first introduction to WMI was by a colleague, Frank Blando, who authored Chapter 13, "High Availabiltiy in Windows Server 2003," and the "Windows Management Instrumentation (WMI)" section of Chapter 10 of this book. In his demo, Frank had concocted a very long WMI command and executed it. This command ran the Cluster Administrator, extracted information from it, formatted it in HTML format, saved the HTML file to the `wwwroot` directory, and then refreshed the data every five minutes. He could then use a browser to view the cluster data. Setting this up on an internal or even an external server would allow you to monitor cluster events without messing with the snap-in and event logs. It was quite impressive.

Realize that these APIs are the same ones called by any other tool, but now you have some easy ways of customizing them. Some of the available providers include

- Active Directory Provider: MicrosoftActiveDirectory
- Trust Health Provider: MicrosoftHealthMonitor
- DNS Provider: MicrosoftDNS
- Cluster: MSCluster

A more extensive description and code samples are provided in Chapter 10.

New Features in ProLiant Architecture and Tools

Since the mid-sixties, the de-facto standard for the rate of increase in processing power has been "Moore's Law." Today, next-generation processors are powering servers to a new era of exponential increase in performance. ProLiant Servers are stepping up the pace with new processor technology called x86 Extensions. The Opteron Processor from AMD (Advanced Micro Devices) and Xeon Processors with EM64T (Extended Memory 64-bit Technology) from Intel bring new processor power and technology to ProLiant.

In addition to these new processors for ProLiant, this section highlights some of the new features and technologies in ProLiant Servers, Server Options, and ProLiant Essentials Software that complement the features found in Window Server 2003.

New x86 Processors from AMD and Intel with 64 bit Extensions

Throughout its life, the X86 processor architecture has been extended many times. One of the most exciting new developments for Industry Standard Servers is the latest extension to the platform. The updated architecture is based on 64-bit extensions to the industry-standard x86

instruction set, allowing today's 32-bit applications to run natively on 64-bit extended processors such as the AMD Opteron and the Intel Xeon with Intel Extended Memory 64 Technology (EM64T). At the same time, new 64-bit applications are executed in 64-bit mode, which processes more data per clock cycle, allows greater access to memory breaking the four gigabyte memory barrier, and speeds numeric calculations. The end result is a platform that leverages the existing wealth of 32-bit applications while also providing a smooth migration path to 64-bit computing.

AMD Opteron 200 and 800 Series Processors

The 200 series processors are now available in the ProLiant DL145 and the 800 series are available in the ProLiant DL585. They offer a substantial performance increase with the Opteron processor architecture. Table 1.6 shows some of the key features and benefits of the Opteron.

Table 1.6 AMD Opteron Processor Features and Benefits

Feature	Benefit
Simultaneous 32- and 64-bit computing capabilities	Allows users to run 32-bit and/or 64-bit applications and operating systems as they desire without sacrificing performance
Support of up to three coherent HyperTransport links, providing up to 19.2GB/s peak bandwidth per processor	Provides substantial I/O bandwidth for current and future application needs
256TB of memory address space	Creates a significant performance benefit for applications in which large (or many) datasets are held in memory
Scales from one-way to eight-way across entire data or compute centers utilizing the same hardware and software infrastructure	Allows for maximum flexibility in IT infrastructure, helping contribute to bottom-line success
Integrated memory controller reduces latencies during memory access in an SMP server system	Yields fast computational processing for increased performance and productivity

Intel Xeon processors with 64-bit Extensions

Intel's new Xeon processors became available mid-2004; HP delivered one- and two-way ProLiant servers with the new Xeon processors in summer 2004, with four- and eight-way models expected in 2005. See the HP Web site (`http//:www.hp.com`) for details. Table 1.7 shows the new Intel Xeon processor features.

Table 1.7 Intel Xeon Processors with 64-bit Extensions Features

Performance Features	Xeon Extensions Add . . .
Increased frequency headroom; 3.6GHz/1MB cache	Additional Registers: 8 SSE and 8 general purpose
800MHz FSB—1.5x system bus speed versus 533MHz FSB	Double precision (64-bit) integer support
DDR2-400—Faster memory technology	Extended memory addressability (64-bit) pointers, registers
PCI Express 4x - 8x—Faster I/O	

Processor Performance Comparison

The performance measurements were conducted on the ProLiant DL145 with 2.2GHz Opteron processors against the ProLiant DL140 server using 3.2GHz Intel Xeon processors and the 32-bit version of Microsoft Windows Server 2003 using Ziff Davis Media Inc.'s WebBench 5.0 benchmark. The benchmark results showed:

- The 2P ProLiant DL145 is 57% faster than the ProLiant DL140.
- The 1P ProLiant DL145 is 44% faster than the ProLiant DL140.
- The ProLiant DL145 with dual processors achieved a 39% higher performance score than with a single processor. The ProLiant DL140 showed only 28% performance scalability. This represents a 39% processor scalability advantage for the ProLiant DL145 over the ProLiant DL140.

Windows Server 2003 for 64-Bit Extended Systems

Windows Server 2003 for 64-Bit Extended Systems provides high performance for both 32-bit and 64-bit applications on the same system. The underlying architecture is based on 64-bit extensions to the industry-standard x86 instruction set, allowing today's 32-bit applications to run natively on 64-bit extended processors such as the AMD Opteron and the Intel Xeon with

Intel Extended Memory 64 Technology (EM64T). At the same time, new 64-bit applications are executed in 64-bit mode, which processes more data per clock cycle, allows greater access to memory, and speeds numeric calculations. The end result is a platform that leverages the existing wealth of 32-bit applications while also providing a smooth migration path to 64-bit computing. Windows Server 2003 for 64-Bit Extended Systems is currently under development, with delivery scheduled for the second half of 2004. Prerelease versions of the operating system are available for evaluation from the Microsoft Web site.

HP is looking forward to the benchmarks for ProLiant with x86 extension processors running Windows Server 2003 for 64-Bit Extended Systems. In this testing, the performance capabilities and benefits of the x86 extensions will be fully realized. It should prove to be an exciting day in ProLiant history.

ProLiant Blade Servers

One of the most notable new products in the ProLiant Line is the blade server. HP pioneered blade technology a few years ago, so this technology is not brand-new. However, it is maturing and starting to catch on as blades are engineered for the enterprise and their compelling benefits are more widely recognized. The basic idea behind blade servers is consolidation. By allowing servers to share resources provided by an enclosure, the individual servers can be made much more compact. The architecture not only allows you to put a lot more processing power into a smaller space, but also its modular design simplifies deployment. In terms of hardware, two basic components are in a blade server solution: the blades and the enclosure that houses them. The enclosure connects the blades installed to the shared resources, and can be configured for I/O options according to your needs. The blades are complemented by automated deployment software that utilizes scripting or drive imaging to rapidly deploy a server. A server can be deployed in 10 minutes over a gigabit Ethernet connection using drive imaging. Blade servers save space, drastically reduce cabling, and simplify installation and the processes in deploying and managing servers. The ProLiant BL-p series offers multiple processors, the benefits of the blade architecture, and many of the enterprise class server features found in ProLiant 300 and 500 series rack mount servers. As an example, the BL20p blade delivers these enterprise-class capabilities:

- Dual-processor capability
- 8GB memory capacity
- Integrated Smart Array 5i Plus
- Dual 3.5-inch SCSI hot plug drive bays
- Dual Fiber Channel ports, optional for redundant SAN connections
- Three 10/100/1000T Ethernet ports
- One iLO (integrated Lights-Out) advanced management port
- Rack-centralized, external shared redundant hot-plug power

These capabilities make the BL20p blades ideal for hosting these applications:

- Web-hosting
- E-commerce
- Computational cluster
- Terminal Server Farm
- AV, streaming media
- Messaging front-end and mobility
- Small database
- Application server

Add in the benefit of storage consolidation by attaching and even booting blades from a SAN, and you can build a robust and rapidly adaptable infrastructure. For some really cool ideas on using blade servers in advanced architectures and configurations, check `http://www.phptr.com/title/0131467581` for technical papers on "Automated Provisioning with ProLiant Servers."

Systems Insight Manager (SIM)

Customer surveys show one of the most popular value-adding feature sets of the ProLiant server family is the enhanced management capabilities present across the platform. HP has brought together the HP OpenView and Insight Manager Development teams. Working together, they have created the next generation systems-management application called HP Systems Insight Manager (SIM). It combines the strengths of Insight Manager 7, HP Toptools, and HP Servicecontrol Manager into a single tool for managing ProLiant, Integrity, and HP 9000 systems running Windows, Linux, and HP-UX. The core SIM software delivers the essential capabilities required to manage all HP server platforms.

Following are some of the core capabilities and benefits of SIM:

- SIM is free and included with all ProLiant and Integrity Servers, or customers can download SIM from the HP Web site.
- SIM offers more stability and efficiency through management across hardware platforms and operating systems. SIM provides common configuration, performance, and management across all HP servers. It also enables management of clients, printers, storage, and other devices.
- Automated fault detection and reporting enables you to detect and respond to potential and actual failures before they result in unplanned server downtime. Staff can be notified via console, pager, or e-mail.

- Automated asset inventory and reports reduce time and effort associated with discovering, identifying, and collecting configuration data for managed systems. Simplified report generation improves preparation tasks.
- Automated software updates reduce time and effort associated with maintaining up-to-date system software baselines. Version control and software repository automates software download, gives available updates, and allows distribution of components to groups of systems.
- Command-line interface in all versions of SIM support command-line access to scripting capabilities.
- Security features are enhanced because all users have individual accounts and log on to SIM using their OS username and password. All HTTP communications are protected using 128-bit Secure Sockets Layer (SSL), and all commands to applications integrated using standard tool definition files are encrypted using Secure Shell (SSH).
- SIM can be extended with plug-ins to deliver enhanced device management for rapid deployment, performance management, partition management, workload management, HP clients, storage, power, and printer products.
- ProLiant Essentials Performance Management Pack now integrates seamlessly with HP SIM to provide hardware bottleneck analysis for ProLiant servers and now Modular Storage Array (MSA) series storage. Five complementary licenses are included in the SIM installation package
- Snapshot comparisons allow staff to compare configuration snapshots of up to four different servers or configuration snapshots of a single server over time. This assists IT staff in pinpointing configuration issues that might contribute to system instability.
- Insight Manager 7 migration utility is included to transfer Windows servers Insight Manager 7 settings to HP SIM with either an in-place upgrade on the same server or transfer of Insight Manager 7 data to SIM on another server.
- SIM uses the same management agents and instrumentation that Insight Manager 7 uses, so no agent updates are required on managed systems when upgrading or installing a SIM server.
- SIM enables consolidation of general IT resource management by managing all HP servers while still allowing OS and server platform specialists to focus on specific needs.
- Improves efficiency by monitoring progress of management tasks across groups of diverse systems from a central point of control.
- Reduces training expenses typically associated with learning multiple management tools.

AD Integration of iLO and RILOE II

The integrated Lights-Out (iLO) advanced and Remote Insight Lights-Out Edition II (RILOE II) standard feature set now includes Directory Services integration. The iLO and RILOE II are hardware-based management processor solutions that provide a "Virtual presence at the server," allowing full access and remote control of the server, independent of the state of the operating system or server hardware even when the server is hung or powered off "in a lights-out state."

What Directory Services integration means is you can now manage user access and privilege levels on the iLO and RILOE II boards through AD, centralizing and simplifying access management. Formerly this was done individually through a browser or in groups via the Lights-Out Configuration Utility. This new feature is free and provides Windows Administrators a familiar interface to manage the iLO and RILOE II across the AD.

ProLiant's iLO Now Provides "Terminal Services Pass Through" for Windows Remote Console Sessions

ProLiant servers with the iLO Advanced Features Pack enabled can leverage iLO's remote console function to provide Terminal Services pass through authentication to Windows Server.

Beginning with iLO firmware version 1.50, the iLO can leverage the OS functionality of Windows Terminal Services and a remote desktop connection to significantly increase the responsiveness of the graphical remote console. Terminal Services complements the technology within iLO by providing a software-based remote console when the Windows Server OS is functioning normally. In the event the Windows Server OS is not functioning normally, iLO can revert to the hardware-based console at any time. This gives administrators the performance of an OS-based, graphical remote console with the assurance that the hardware-based iLO remote console is available at all times.

One Button Disaster Recovery (ODBR) Tape Drives

OBDR can rapidly restore or replicate a server's operating system, software applications, and data using tape cartridge(s). This is possible because the tape drive emulates a CD-ROM that allows the server to "boot from tape." This standard feature is embedded in the firmware of HP tape drives.

ODBR is easy to set up and use. It is supported on ProLiant ML and DL servers with fifth- and sixth-generation Smart Array controllers; Ultra3 and Ultra320 SCSI controllers; and SDLT, DAT, and Ultrium Tape Drives. Some of ODBR's key benefits are

- Provides fast restores and server replication using only a tape cartridge.
- No additional media, such as CD-ROMs or floppies, are required.

- The HP OBDR functionality allows an IT Administrator to perform a complete system restore from a remote location via the HP RILOE on ProLiant Servers. The process includes the ability to invoke the OBDR mode during Power On Self Test (POST) using the <F8> function key from the remote client. The powerful combination of OBDR and RILOE gives the Administrators the ability to completely recover a failed server at a remote location without physically traveling to where the server resides.
- Independent software vendors supporting OBDR on Window Server 2003 are Yosemite TapeWare, VERITAS Backup Exec, Novastor NovaNet, and ULTRABAC.

References

This book's two Web sites contain information and downloads for ProLiant tools and utilities.

Authors' Web site: `http://WindowsOnProLiant.com`

Publisher's Web site: `http://www.phptr/title/013146758`

"Using Software Restriction Policies to Protect Against Unauthorized Software," Whitepaper at `http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/winxppro/maintain/RstrPlcy.asp`.

Balladelli, Micky and Jan de Clercq, *Mission Critical Active Directory*. Digital Press, 2000.

de Clercq, Jan, *Windows Server 2003 Security Infrastructures*. Digital Press, 2004.

Moskowitz, Jeremy, *Group Policy, Profiles and IntelliMirror for Windows 2003, Windows XP and Windows 2000*. Sybex, March 2004.

Olsen, Gary L., *Windows 2000: Active Directory Design and Deployment.* New Riders, 2000.