

Securing an Active Directory Network Infrastructure: best practices and security concepts for small- and medium-sized networks

By

**Brian Yoshino
Lewis University**

September 12, 2006

(This page has been intentionally left blank)

Table of Contents:

Introduction	4
Dynamic Host Configuration Protocol	5
DHCP Server Configuration Basics	6
DHCP Client-Server Interactions	6
Automatic Private IP Addressing	7
DHCP Server hardware recommendations	8
Common Types of Attacks	9
Best Practices	12
Windows Internet Name Service	21
WINS Basics	21
NetBIOS name resolution	23
WINS Server Replication	24
WINS Server hardware recommendations	26
Ways a WINS server can be attacked	27
Best Practices	28
Domain Name System	33
DNS Basics	33
DNS Server Hardware Recommendations	35
Common types of Attacks on DNS	36
Best Practices	38
Works Cited	45

Introduction

Active Directory, first introduced in Windows 2000, is Microsoft's implementation of a network directory service. Active Directory is the central location for network administration and allows for the delegation of administrative authority. Active Directory uses the concepts of forests, domains, and trees to provide security and scalability for any sized organization, from small businesses to global enterprises.

Like any network directory service, Active Directory requires a number of core infrastructure services to function. These services provide two primary purposes, IP configuration and name management. Dynamic Host Configuration Protocol, or DHCP, is a client-server based networking protocol that provides IP address configuration information to requesting DHCP clients. In theory, a network administrator could run a fully functioning Active Directory network by using static IP addresses that have been manually configured. While this may be feasible in a small network, the administrative overhead of accomplishing this on a medium- or large-scale network is nearly impossible. In addition, even on very small networks, the chance for errors in the configuration of IP address information is reason enough to move to DHCP.

To accomplish name resolution, Active Directory makes use of two technologies: the Domain Name System (DNS) and the Windows Internet Name Service (WINS). DNS is the foundation of Active Directory, while WINS is more of a remnant from the NT4 days. Unfortunately, unless you run a pure Windows 2000 or later network and use network-aware applications that were built for pure Windows 2000 networks, you will still need to have a WINS server on your network.

While necessary for the proper functioning of Active Directory, these core infrastructure services require careful installation and configuration. There are a number of security concepts and best practices that should be followed to ensure proper security and reliability. The goal of this paper will be to enumerate and describe these concepts and practices.

Dynamic Host Configuration Protocol

The Dynamic Host Configuration Protocol (DHCP) was designed to automate the way IP addressing and configuration information was distributed to clients. Before DHCP, administrators of TCP/IP networks would have to manually configure the hosts on their networks. While this may seem like an easy task in small environments, it easily became an administrative nightmare once the network grew to more than just a few clients. Not only was the task of manually configuring each machine a project in and of itself, but this method was also prone to errors.

DHCP, and its predecessor BOOTP, were invented to solve this problem. In the next section we will review some of the basic concepts surrounding DHCP. Note that this is not meant to be a complete review of Microsoft's implementation of DHCP, and the reader is expected to have general familiarity in supporting a DHCP installation. Nevertheless, due to the complex nature of DHCP, a short review is in order. Later we will discuss some of the known attacks common to a DHCP infrastructure, and finally we will review a list of best practices that can be used to secure DHCP.

The DHCP service architecture makes use of three components: DHCP clients, DHCP servers, and DHCP relay agents. A DHCP client can be any network-enabled device that can communicate with a DHCP server with the purpose of obtaining IP addressing information. All modern operating systems include DHCP clients as do various hardware devices. For instance, network-enabled printers have become more prevalent in recent times due to falling costs and the increased prevalence of small business networks. These printers, using either on-board network adapters or external printer server modules, can request IP addressing information from a DHCP server just like any other client.

DHCP Servers are Windows 2000 Server or Windows Server 2003 computers running the DHCP Server service. These servers have been configured by an administrator to distribute IP addressing information to requesting clients on the network. This configuration includes scope, reservations, and other options as designated by the administrator. Finally, DHCP Relay Agents are software or hardware devices that pass DHCP messages between clients and servers. As we will see in just a moment, some DHCP messages are sent as broadcasts. Therefore, if a DHCP client was on one subnet, and a DHCP server was on another subnet, since the router would not pass the broadcast, the client would never be able to lease an IP address. This is where DHCP relay agents come in; they relay these broadcast messages on behalf of the client and act as a proxy between the DHCP client and the DHCP server. Otherwise, an administrator would have to install a DHCP server on every subnet that had DHCP clients.

DHCP Server Configuration Basics

Before a DHCP server can begin to lease IP addressing information out to DHCP clients, it must be configured with certain information. For example, it needs to know the range of IP addresses that it can hand out. This and other information makes up what is known as a *scope*. The scope is a range of IP addresses and related TCP/IP configuration information that the server can lease out to DHCP clients on a specific subnet. Along with the IP address ranges, including any reservations and exclusions, the scope also includes information such as subnet masks and lease durations.

A DHCP reservation is an IP address that the server can be configured to set aside for the exclusive use of one DHCP client. For instance, say there is a printer on your network with a network card that you always want to have a certain IP address. You can configure the DHCP server so that whenever that DHCP client requests an IP address, the server will consistently assign it a specific IP address you choose. An exclusion on the other hand, is a particular address or range of addresses in the scope that you forbid the DHCP server from leasing out. You might do this if you have a device on the network that doesn't support DHCP. You can configure the device with a static IP and then configure the server not to hand out that particular IP address to anyone.

As mentioned before, besides the IP address itself, the DHCP server will also offer the DHCP client configuration information regarding the subnet mask, the default gateway, local DNS and WINS servers, as well as any other pertinent information the administrator deems necessary. In addition, the DHCP server will also inform the client as to how long the IP address is good for. This is called the *lease duration*, and by default it is set for 8 days [1]. Suggestions for changing the length of the lease will be discussed later on.

An administrator can also aggregate multiple scopes on a DHCP server into something called a *superscope*. A superscope allows the DHCP server to lease out addresses from multiple scopes to clients on a single physical network. For example, the administrator might need to support DHCP clients that reside on the same physical LAN, but exist in different logical IP networks. Though relay agents will need to be used, the superscope can be used to aid in the administration of the DHCP server.

DHCP Client-Server Interactions

Next we review the client-server interactions that take place during the DHCP IP address lease process, which starts with the client as shown in Figure 1. When a DHCP client first boots up and connects to the network, it broadcasts a UDP *DHCPDiscover* message requesting IP address information from a DHCP server. If there are DHCP servers available, each of them will respond to the *DHCPDiscover* message with a *DHCPOffer* message that includes an IP address

for the client. The IP address is a currently un-leased address from the server's range of available IP addresses. The client will accept the first offer it receives by broadcasting a *DHCPRequest* message. This broadcast message serves a couple of purposes. First, it includes the IP address that it was offered from the DHCP server's *DHCP Offer* message it accepted, and it lets the other DHCP servers on the network know they can return their offered IP address back in to their respective pools. The transaction is complete when the DHCP server responds with a *DHCP Ack* broadcast to the client acknowledging the request and offering it the remaining IP configuration options.

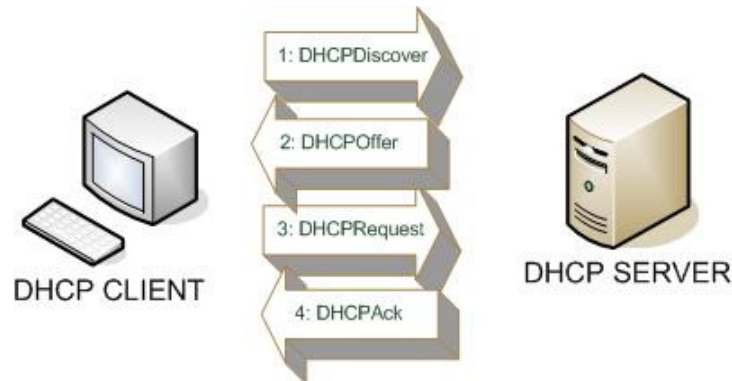


Figure 1: DHCP Client-Server message interactions

Now that the DHCP client has obtained IP addressing information, it does not get to use that configuration information indefinitely. As we mentioned before, the IP address is leased out to the DHCP client for a predefined amount of time called a lease. The DHCP client will first attempt to renew the lease when 50 percent of the original lease time has passed [2]. At this time, the DHCP client will send a *DHCPRequest* message to the DHCP server that originally configured the client. If the DHCP server and lease are still available, the server will respond with a *DHCP Ack* message renewing the lease with the client. If the DHCP server is available, but the lease is no longer available, the DHCP server will respond with a *DHCPNack* message. At this point, the DHCP client starts the process to obtain a brand new lease.

If there is no response from the server, the client will wait until 87.5 percent of the lease time has passed [2]. At this point, the client will broadcast a *DHCPRequest* message to attempt to renew the lease with any DHCP server. If no DHCP server is available by the time the lease expires, the client disassociates itself from the lease and begins the process to obtain a new lease.

Automatic Private IP Addressing

One last feature we should mention that is part of recent Microsoft Windows operating system DHCP clients is Automatic Private IP Addressing, or APIPA. This feature allows DHCP clients to obtain IP configuration information in the event a DHCP server is unavailable. APIPA is meant to ensure computers are

able to communicate if no DHCP server is available. Note that this is meant only for very small installations. Since the automated configuration does not support a default gateway, it only supports systems on a single subnet.

The auto-configuration begins like this. The DHCP client starts up and is unable to contact a DHCP server. At this point, the DHCP client will randomly choose an address from the Microsoft reserved class B network 169.254.0.0 with a subnet mask of 255.255.0.0 [2]. The DHCP client will then test the address using the ARP protocol. If a conflict is found, the DHCP client will select a different address. If no conflict is detected, the client will configure itself to use that address until a DHCP server is available. Recognizing an APIPA address is a great troubleshooting tool since having systems on the network configuring themselves with APIPA assigned addresses is a clear message that something is wrong with the DHCP server.

DHCP Server hardware recommendations

Before we get in to the attacks and best practices of configuring a DHCP server infrastructure, we shall discuss the basic hardware recommendations that Microsoft suggests for a DHCP Server. Although the DHCP Server service is critical for most networks to function, the actual hardware requirements are not very great. Microsoft's Windows Server 2003 DHCP development team set up a DHCP server running Windows Server 2003 in a lab environment in order to stress-test the service [3]. The system had the following hardware specifications as seen in Table 1:

Table 1: DHCP testbed server configuration

Processor:	Two x86 Family 6 Model 7 Stepping 3 Intel ~498 MHz
Physical Memory:	256 MB
Network Adapters:	(3) Ethernet 802.3 100 Mb/sec
Subnets Serviced:	Six, four of which are separated from the test server by routers running the DHCP Relay Agent Service
Operating System:	Windows Server 2003, Enterprise Edition
Number of Scopes:	5,155
DHCP database size at maximum load:	2 GB
Additional factors:	Several thousand exclusion ranges, option values, and reservations are configured in the scopes of the server

The DHCP server was subjected to both valid and invalid DHCP client lease and renewal requests for 1,152 hours (48 days).

The exact results of the lab test are out of the scope of this paper, but the test demonstrated that the server was able to successfully accommodate the load without any errors or failures. The hardware used for the lab test, while ancient

by today's standards, proves that "big iron" is not necessary. That is not to say you should put any old machine into production as your DHCP server. DHCP is a disk-intensive service with frequent and intense activity on the server hard disks. Microsoft recommends purchasing hardware with a RAID solution [1], but in my experience, this usually isn't necessary.

To make evaluating the performance of your DHCP server easier, Microsoft includes a number of performance counters that can be used to monitor various areas of server activity [4]. Once the DHCP service is installed, you can use these metrics and counters to assess the function and configuration of your server and make hardware decisions.

Also, note that the DHCP server itself is required to have a statically assigned IP address. When it comes to the IP address configuration for servers and critical workstations, there are a couple schools of thought. One suggestion is to make every device on your network, besides your DHCP server of course, a DHCP client. This includes not only client PCs, but also servers and devices such as network printers. You can use DHCP reservations to ensure that these DHCP clients are assigned that same IP address each time they start up. While this option requires less work in configuring those particular clients, it is better practice instead to statically configure those critical devices such as servers and vital workstations. Doing this eliminates the possibility of having rogue or misconfigured DHCP servers assign incorrect IP address information, resulting in a Denial of Service (DoS) situation. Rogue DHCP servers will be discussed later on.

Common Types of Attacks

Before we delve into the techniques for securing a DHCP server, we will discuss a few typical types of attacks that can compromise a DHCP infrastructure. Recommendations on how to prevent such attacks will be offered. In addition, the next section on best practices will offer more suggestions on how to best protect your DHCP service.

Unauthorized DHCP Servers

There are a number of different ways that an unauthorized DHCP server can appear on your network. Every deployment of Windows 2000 Server and Windows Server 2003 can function as a DHCP server. In addition, other devices such as broadband routers can also run DHCP server services. Having an unauthorized DHCP server on your network can create much havoc. There is nothing to stop these servers from handing out bogus or incorrect IP addressing information to clients on your network. This creates a DoS situation since these clients will not be able to interact with other systems on the same network.

Windows 2000 Server's Active Directory introduced the concept of DHCP server authorization. The idea is that before a DHCP server can begin servicing clients

on the network, it needs to be authorized in the Active Directory first. Further, this authorization can only be done by a member of the Enterprise Admins group [5]. In effect, what this does is prevent rogue Windows 2000 Server or Windows Server 2003 DHCP servers from handing out IP addressing information. When a computer running Windows 2000 Server or Windows Server 2003 that has the DHCP server service installed starts up, the Active Directory is queried. The IP address of the DHCP server is compared to a list of authorized DHCP servers. If the IP address is on the list, the DHCP server considers itself authorized and will begin servicing clients. If a match is not found, the DHCP server assumes that it is unauthorized and will not respond to DHCP requests. It is important to note that only domain controllers or member servers can be DHCP servers when an Active Directory is present. If you wish to use a stand alone DHCP server, it cannot be on a subnet with any other authorized DHCP servers. If a stand-alone DHCP server detects an authorized server on the same subnet, it will cease leasing IP address information to requesting clients. This detection process occurs every 60 minutes for authorized servers and every 10 minutes for unauthorized servers [6].

Unfortunately, this mechanism only works for Windows 2000 Server or Windows Server 2003 DHCP servers. DHCP servers running other operating systems, including Windows NT4, will continue to hand out IP addressing information regardless of whether their IP address is authorized to. The only way to prevent these rogue DHCP servers from appearing on your network is to ensure the proper physical security of your network infrastructure.

One way to assist administrators in surveying their networks for unauthorized DHCP servers is the DHCP Server Locator Utility (`dhcplloc.exe`) command line tool that is included in the Windows XP Support tools. This tool displays the active DHCP servers on the subnet and can send out an alert if any unauthorized DHCP servers are detected [7].

Unauthorized DHCP clients

Unauthorized clients also pose a threat to the DHCP infrastructure. For instance, it's possible that a hacker could use a compromised system to create a DoS situation by submitting multiple DHCP requests to the DHCP server using spoofed MAC addresses. Depending on the size of the scope, it is possible that this client could lease all of the server's available IP addresses so none were available for legitimate clients. Note that due to the prevalence of unsecured wireless networks, a hacker could potentially achieve that same result by using his own equipment and conducting the attack from a remote location. Even those systems that have DHCP reservations are not immune to this type of threat. It is possible to configure the rogue client to drain the pool not only of unreserved addresses, but those addresses set up with reservations as well. For instance, the rogue client could sniff out the *DHCPDiscover* broadcasts from clients with reservations and create a list of those MAC addresses. The rogue client could then attempt to disrupt communication or hijack the lease. The best

defense against this type of attack is to ensure that the hosts on your network are free from infection and that your network is secure from unauthorized clients.

Compromised server giving out bad info

Compromised servers can prove to be an even bigger threat than they might seem. A compromised DHCP server can create a DoS attack by leasing out incorrect IP addressing information to clients, such as invalid IP addresses, incorrect subnet masks, and so on. Doing so would prevent these clients from being able to communicate with other systems on the network. But as Tulloch describes, the consequences of a compromised DHCP server can be even more damaging. For instance, the DHCP server may be configured to provide clients the correct IP address and subnet mask information, but also provide them with incorrect DNS server settings that point them to rogue DNS servers as shown in Figure 2. These DNS servers, under the control of the hacker, could then redirect clients to websites also under the hacker's control that contain malicious code to infect the client [8]. Tulloch also describes a situation where instead of providing incorrect DNS information; the compromised DHCP server provides a rogue gateway address as shown in Figure 3. This gateway, under the hacker's control, will read and forward all the traffic it receives to the legitimate gateway [8]. The user would never know that his outbound traffic is being exposed. It is critical that DHCP server administrators take the proper steps to ensure their servers remain secure.

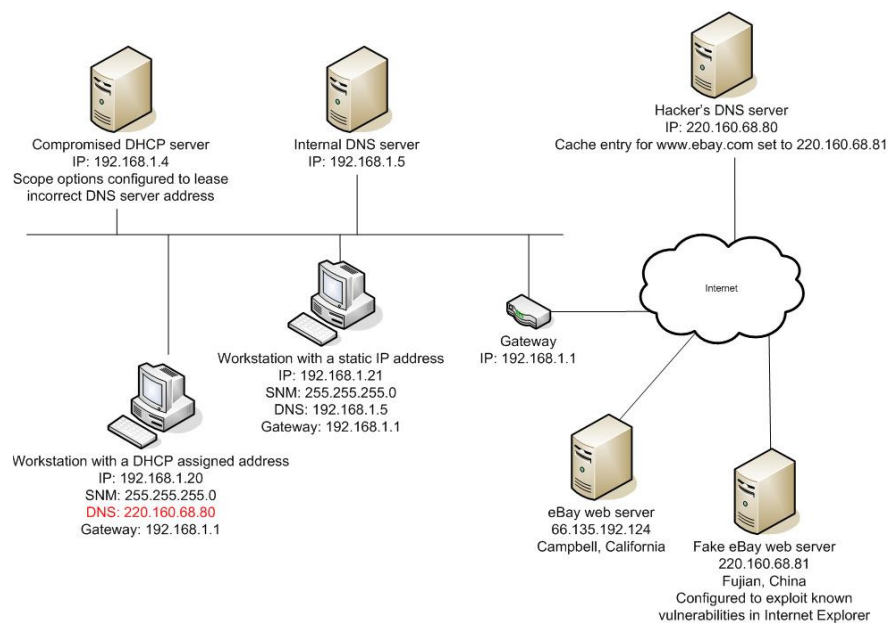


Figure 2: Rogue DNS attack

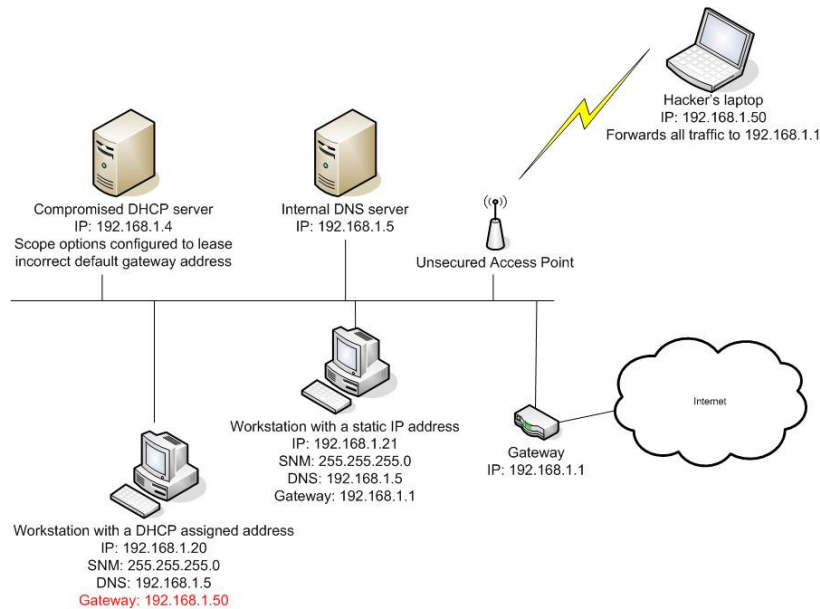


Figure 3: Rogue gateway attack

DHCP Server overwrites a valid DNS resource record

We will be discussing DNS in depth later in this paper, but because of the integration of DHCP and DNS, we will mention this type of attack here. Most Active Directory installations have both forward lookup DNS zones and reverse lookup DNS zones. By default, Windows 2000, 2003, and Windows XP DHCP clients register their Host (A) resource record in the forward lookup zone, while the DHCP server registers the Pointer (PTR) resource record on the client's behalf. More important is the fact that the DHCP server owns the PTR record. DNS servers configured to allow only secure dynamic updates have their discretionary access control list configured so that only the owner of a resource record is authorized to modify it. If an attacker were able to alter the DHCP server's configuration, they could configure the server to register and own both resource records, thereby preventing the client from updating its IP information in the DNS database.

Best Practices

In this section we will discuss the best practices for the installation and configuration of a DHCP infrastructure. Some of these recommendations will apply to any DHCP server installation while others may only apply to specific types of installations. It is up to the administrator to determine which methods apply to their specific network.

Physically secure your DHCP server

While much of the discussion surrounding how to secure a DHCP server infrastructure will have to do with proper installation and configuration, no security guide would be complete without at least mentioning the aspect of

physical security. Without physical security, you have no security. If an attacker can gain physical access to your DHCP server computer, all the configuration hardening steps that the administrator may have taken become moot. Not only can the attacker attempt to gain local access to the console, they may even be able to steal the machine! The DHCP server or servers should be physically isolated so that only authorized personal can access them. Note that this includes not only outside intruders, but employees as well. For instance, housekeeping personnel can unknowingly create a DoS attack by unplugging the server in order to use the outlet for a vacuum or other device. In the very least, the critical servers on the network should be secured behind a locked door, and access to the room should be highly controlled. Large installations may warrant anything from biometrically controlled access to security guards and security cameras.

Restrict who has administrative access to the DHCP service

Only members of the Administrators group and the DHCP Administrators group are able to administer DHCP servers [9]. This applies whether they are using the DHCP console or the Netsh command line utility. Administrators should ensure only the proper user accounts are members of these groups. Regular auditing of group memberships should be part of every administrator's task list.

Keep the DHCP server computer up-to-date with patches

You are going to want to run your DHCP server on an operating system that is Active-Directory-aware such as Windows 2000 Server or Windows Server 2003. Since the DHCP service runs on top of the server operating system, you need to make sure that the server is kept up to date with all of the Microsoft patches and service packs. Any vulnerability that can be exploited within the operating system, whether it is related to DHCP or not, will affect the DHCP Server service. Fortunately, tools such as Windows Update and the new Windows Server Update Services (WSUS) can help administrators ensure their server computers are up-to-date.

Backup your DHCP server's configuration

The bane of existence for any administrator is backups. The need to consistently create backups of all your data in the hope you never have to use them is a very time-consuming process. Not only does an administrator need to create the backups, they need to verify the backups completed successfully, conduct test restores to guarantee the procedure for restoring data works, and finally ensure the backups are kept in a safe place. Most administrators have their hands full keeping the network up and running, let alone enough time and resources to ensure every piece of data can be recovered in a moment's notice. Nevertheless, administrators of DHCP servers should backup the DHCP server's configuration often.

The DHCP server database in the Windows Server 2003 family uses the Exchange Server JET storage engine [10]. Upon installation of the DHCP

service, a number of files are created in the %systemroot%\System32\dhcp directory as shown in Table 2.

Table 2: DHCP server database files

dhcp.mdb	The DHCP server database file
dhcp.tmp	A temporary file used as a swap file during database maintenance operations
J50.log and J50####.log	Transaction logs used to recover data if necessary
J50.chk	A checkpoint file

There are a number of different ways to backup the DHCP database. The administrator is free to choose whichever method works best in their environment. By default, the DHCP server conducts synchronous backups automatically every 60 minutes [11]. During a synchronous backup, the entire DHCP database is saved. This includes:

- All of the scopes (including superscopes and multicast scopes)
- Reservations
- Leases
- All options, such as server options, scope options, reservations options, and class options.
- All related registry keys.

These backups are stored in the %systemroot%\System32\dhcp\backup folder. You can change the path of the backups by selecting a different folder in the DHCP server properties as shown in Figure 4.

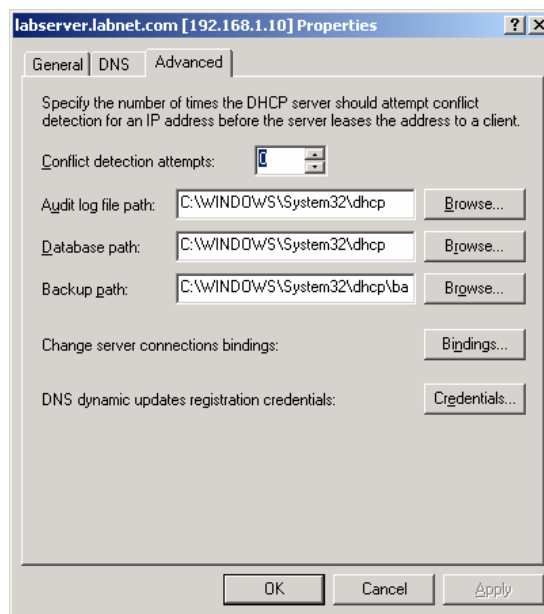


Figure 4: DHCP Server backup path

You can also perform manual backups, known as asynchronous backups. Manual backups, shown in Figure 5, are performed within the DHCP console and backup the same information as the automated backups [11]. Once the manual backup is completed, you should move the backup files to another drive or external backup media.

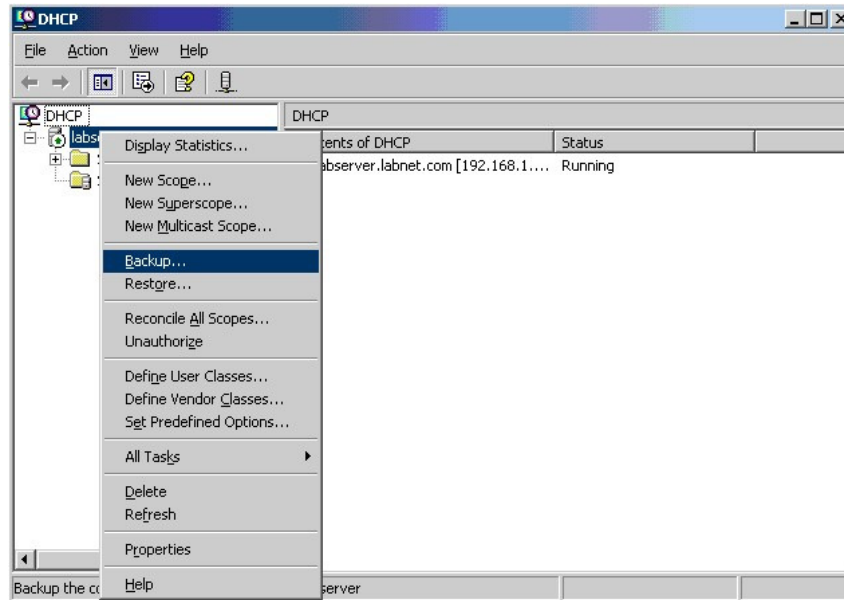


Figure 5: Manual backup of DHCP data

Finally, administrators who prefer using the command line and writing scripts can use the Netsh command-line utility. Using the *dump* command, the administrator can dump the configuration of the DHCP server to the command prompt window, or more usefully, direct it to a text file. To use this command, go to a command line and type:

```
netsh dhcp server dump > dhcp_server_backup.txt
```

This command creates an ASCII file called `dhcp_server_backup.txt` with the configuration information of the DHCP server. If the DHCP server fails, you can copy this file to a new DHCP server and use the netsh command to restore the configuration.

```
netsh exec dhcp_server_backup.txt
```

Note that this method will not backup or restore lease information, but it will allow you to save and recover the most important information such as scopes, options, and etc. In addition, none of the methods described above backup the DNS dynamic update credentials the DHCP service uses to register DHCP client systems in DNS [11]. These accounts will be discussed later on.

DHCP Server redundancy and fault tolerance

Unlike the databases of Active Directory, the DHCP database is not distributed. If you want to create redundancy in your DHCP server infrastructure, you have only a few choices. One solution is to deploy a DHCP server cluster [14]. While this may be the most fault-tolerant solution, it is also the most complex. Only the largest networks may want to consider this solution. Another option is to set up two separate DHCP servers on the same subnet, though caution must be exercised here. It might seem easy enough to set up two separate machines and configure them each the same way. But if we configure each DHCP server with the same scope, there's nothing stopping the first DHCP server from handing out an IP address from its scope to a client and the other DHCP server handing out the exact same IP address to a different client. In order to make this work, we can use the "80/20 design rule" (Figure 6) suggested in the Microsoft DHCP documentation. In order to support a single network and scope using two DHCP servers, you configure one DHCP server with 80 percent of the addresses and the other server with the remaining 20 percent. This way, if one of the servers goes down, the other server can continue to lease new addresses and renew existing clients.

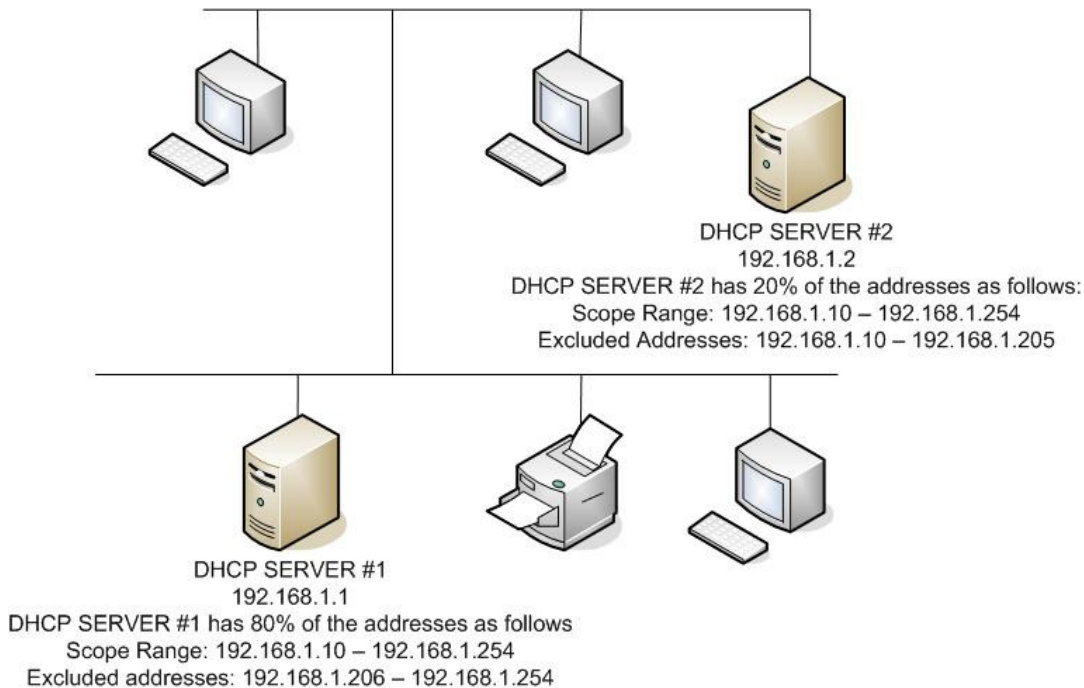


Figure 6: "80/20" DHCP design configuration

Review the Audit Logs

By default, the DHCP service enables audit logging of DHCP events. You can disable audit logging via the DHCP server's properties tab, but it is a good idea to leave logging turned on. These logs provide the administrator with information regarding which clients are requesting addresses from the DHCP server as well

as information regarding BAD_ADDRESS entries that have occurred due to an address conflict. Such conflicts can occur if a rogue DHCP server is assigning an IP address that is already in use or if a rogue client has a static IP configured that is part of the DHCP server's address range. Regardless of the reason, the BAD_ADDRESS log entry will let the administrator know that there is a problem somewhere within the DHCP infrastructure. Audit logs may also provide warnings that unauthorized clients are present on the network. If an unusually high number of IP address lease requests are being directed toward a DHCP server, it may be evidence that the DHCP server is under attack and in danger of having its address pool depleted.

By default, these logs are stored in the %systemroot%\System32\dhcp folder. Several log files are created, one for each day of the week [12]. There are a number of disk checks in place to ensure that the audit logs do not consume too much disk space and that the log files do not grow too rapidly.

Monitor the performance of the DHCP server

Earlier we discussed that even though the DHCP service does not require "big iron" to run, administrators will still want to be aware of the hardware requirements of the DHCP server service. Fortunately, Microsoft includes a number of performance counters that administrators can use to benchmark the performance of their system. There are over a dozen different metrics such as "Requests/sec", "Discovers/sec", and "Active Queue Length" [4]. A complete discussion of all the available performance counters is beyond the scope of the paper, but administrators should be aware they exist to help in measuring the performance of the DHCP server and troubleshooting the service when problems arise.

Make use of the DnsUpdateProxy group

Although DNS will be discussed later in this paper, the integration of DHCP and DNS warrants discussion of this feature here. We have already mentioned that the DHCP server can register records on behalf of its clients in the dynamic DNS database. It is important to note that in DNS zones that use secure dynamic updates as shown in Figure 7, this can cause resource records to become outdated, or stale.

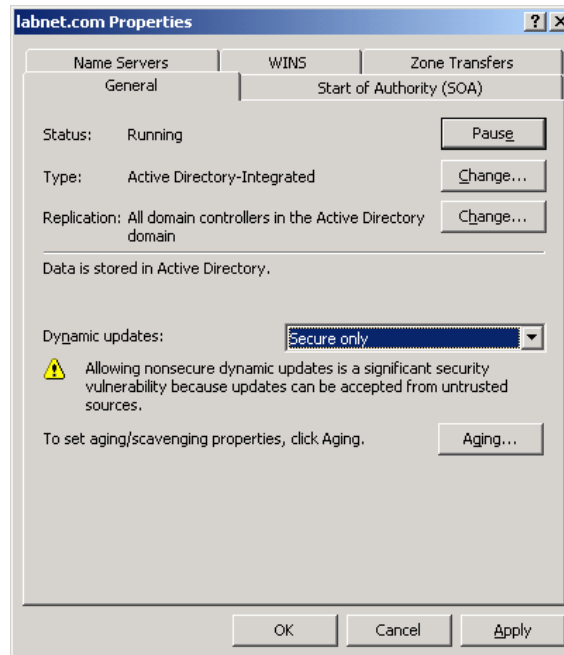


Figure 7: DNS configured to accept only secure updates

Consider the following scenario. Suppose that you have an NT4 client on the network. Since NT4 is not capable of updating its own resource records, the DHCP server can register the appropriate records on the client's behalf. In doing so, the DHCP server becomes the owner of the records and therefore the only computer that can update the DNS records for that name. Now, if that client is upgraded to Windows 2000 or Windows XP, the client will not be able to take ownership of the record. For this reason, Microsoft has included a security group called the DnsUpdateProxy group. Adding the DHCP server computer account to the group prevents it from taking ownership of the resource records it registers on behalf of the clients. In addition, the records are not secure, which means that additional configuration steps need to be followed when using Active Directory-integrated DNS zones that allow only secure updates. Microsoft Windows 2000 Server with Service Pack 1 or later and Windows Server 2003 allow you to configure the DHCP server service to use another user account when registering DNS records [13]. Therefore, instead of being registered under the DHCP server's computer account, this service account is used. On a Windows Server 2003 computer, you can configure the alternate credentials on the Advanced tab of the DHCP server properties page as shown in Figure 8.

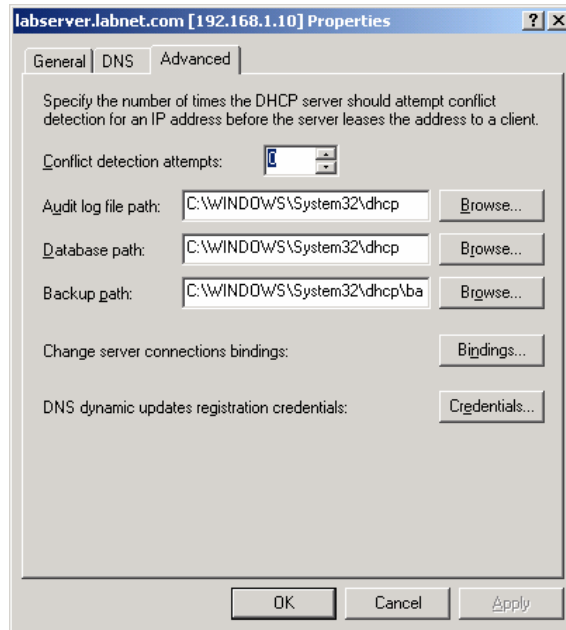


Figure 8: Configuring alternate DNS dynamic update credentials

This is especially important when running DHCP on a domain controller computer. When DHCP is run on a domain controller, the DHCP service inherits the security permissions of the domain controller. Without specifying alternate credentials, the DHCP service will inherit the authority to modify or delete any DNS record in the zone [5].

Adjust lease length based on the needs of the network

For most implementations, the default lease period of 8 days will work fine. In other cases, you may want to increase or decrease the length of the leases to either reduce network traffic or make better use of your address space. For instance, in a large network where the majority of the clients are workstations that never move, you may want to increase the lease duration to 2 or 3 weeks. This can reduce the amount of network broadcast traffic. On the other hand, if a particular subnet of your network had remote access or wireless clients, you might choose to reduce the lease period to only a few days to make better use of the available scope addresses.

Ensure the proper placement and number of DHCP servers

As we mentioned before, the DHCP server service does not require an enormous amount of computing horsepower to run. And while most networks can be supported by a single DHCP server, administrators will still need to conduct proper planning when designing their DHCP server infrastructure. Proper use of DHCP relay agents will ensure DHCP clients on other subnets can access IP address information. In addition, the need for fault tolerance may dictate the need for two DHCP servers. Finally, bandwidth issues may require the use of multiple DHCP servers. If your network includes slow links, you may want to install a DHCP server on both sides of the link so clients may be served locally.

Use IPSec to secure the DHCP server:

In an environment where running the DHCP server service is the only network service the Windows 2000 Server or Windows Server 2003 computer has, high security installations may choose to implement IPSec filters on the computer. These filters listed in Table 3 will restrict traffic accepted by the server to DHCP-related traffic only [5].

Table 3: IPSec rules to secure a DHCP server

Service	Protocol	Source Port	Destination Port	Source Address	Destination Address	Action	Mirror
One Point Client	ANY	ANY	ANY	ME	MOM server	Allow	Yes
Terminal Services	TCP	ANY	3389	ANY	ME	Allow	Yes
Domain Member	ANY	ANY	ANY	ME	Each DC's IP address	Allow	Yes
DHCP Server	UDP	68	67	ANY	ME	Allow	Yes
All in-bound traffic	ANY	ANY	ANY	ANY	ME	Block	Yes

Windows Internet Name Service

Microsoft Windows networks make use of two different name resolution technologies. In this section we will discuss securing the Windows Internet Name Service, or WINS. The acronym WINS is in fact a bit of a misnomer. The Windows Internet Name Service has nothing to do with the Internet. Instead, WINS was designed to provide a way to resolve the NetBIOS names of computers on a network to IP addresses. As with DHCP, in order to fully understand the reasons and logistics of securing a WINS infrastructure, we should briefly review how WINS works. Again, this is not meant to be a complete tutorial on WINS, and the reader is expected to have a basic understanding of WINS.

Although Windows 2000 and 2003 Active Directory networks are built on DNS, WINS is still required in almost every Microsoft network. Ever since Windows 2000 and Active Directory were in development, administrators have been promised that the death of NetBIOS is near. In fact, there are Microsoft documents that state that WINS servers are not needed in a network consisting entirely of Windows 2000-based computers [15]. Unfortunately, those promises have been greatly exaggerated. Administrators will need to run WINS servers for many years to come.

The fact of the matter is that even if you no longer need to support legacy clients such as Windows 95, Windows 98, or Windows NT, if you run any network applications that use NetBIOS names, you will need to have a WINS infrastructure in place. Even Microsoft Exchange Server 2003, which was released in October of 2003, requires NetBIOS name resolution for full functionality [16].

WINS Basics

WINS provides a distributed database of NetBIOS names and IP addresses and was created to solve the problems of broadcast-based name resolution along with removing the burden of maintaining static LMHOSTS files. Computers on the network could be configured with any number of NetBIOS names to identify the resources they provide on the network. For example, a Windows 2000 client will have a unique NetBIOS name which is exclusive to a single process running on the computer as well as a group name which might address multiple processes running on a number of computers.

Before WINS, NetBIOS name resolution was accomplished in a couple different ways. One method of finding out another system's IP address from its NetBIOS name was by a broadcast. While this method worked fine on a small network, once the network grew the amount of broadcast traffic on the network also grew. In addition, since the broadcasts would not pass routers, this method would not work to find systems on other subnets. To remedy this, administrators used files

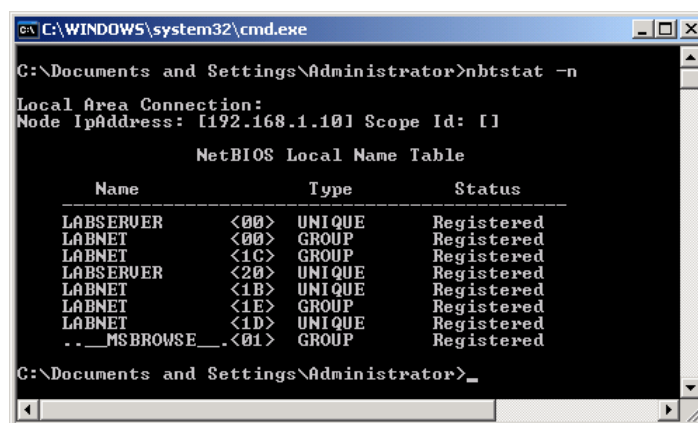
called LMHOSTS files. These files would reside on each system and provide static mappings of NetBIOS names and IP addresses. While this solved the problem of finding systems beyond routers, it added a major administrative burden. WINS was developed to solve these issues.

There are two main components to a WINS infrastructure, the WINS servers and the WINS clients. The WINS server handles name registration and name release requests from clients and responds to name resolution queries from WINS clients by returning the IP address of the name being queried. In addition, they also replicate with other WINS servers. WINS clients will register and release their NetBIOS names with the WINS server and will make use of the WINS server for name resolution requests.

When a computer running Microsoft Windows starts up, it will try to register its NetBIOS names and IP address with the first WINS server with which it has been configured. It will send one registration request for each NetBIOS-based network service running on the computer [17]. A detailed analysis of the types of NetBIOS names is beyond the scope of this paper, but let's discuss a few examples.

Every computer on the network will register the *computername<00>* NetBIOS name [18]. This is the Workstation service and is an example of a unique name. A computer that has Microsoft file and printer sharing enabled will register the *computername<20>* NetBIOS name [18]. Domain controllers will register the *domain name<1C>* name [18]. This is an example of a group name.

You can see a list of the NetBIOS names of any computer by typing **nbtstat -n** at the command prompt. Figure 9 shows the output of running the command on domain controller.



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Administrator>nbtstat -n
Local Area Connection:
Node IpAddress: [192.168.1.10] Scope Id: []

NetBIOS Local Name Table

Name                Type                Status
-----
LABSERUER           <00>                UNIQUE              Registered
LABNET              <00>                GROUP               Registered
LABNET              <1C>                GROUP               Registered
LABSERUER           <20>                UNIQUE              Registered
LABNET              <1B>                UNIQUE              Registered
LABNET              <1E>                GROUP               Registered
LABNET              <1D>                UNIQUE              Registered
.._MSBROWSE_..     <01>                GROUP               Registered

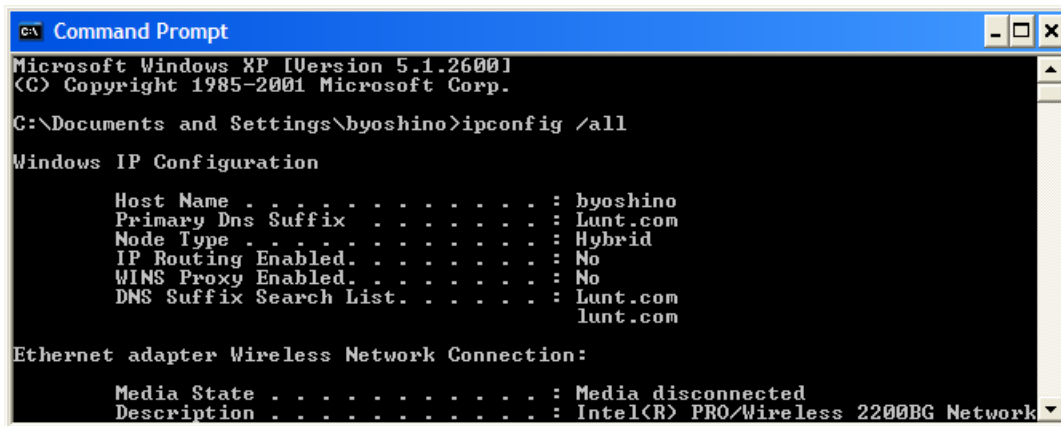
C:\Documents and Settings\Administrator>
```

Figure 9: NetBIOS names of a sample domain controller

This registration process allows the client to confirm that its name on the network is unique. If the name is already used on the network by an active client, the WINS server will respond with a “fail” message and the client will consider the name unregistered. Similar to DHCP, the name registrations have a finite lifetime. By default, it is 144 hours (6 days) [18]. In addition, clients will send name refresh requests. When a workstation shuts down normally, it will send a name release request to the WINS server so that another system can register the same name if necessary.

NetBIOS Name Resolution

By default, when Microsoft Windows 2000, XP, or Windows Server 2003 clients start up in an Active Directory network, they are configured to use the hybrid mode (h-mode) resolution processes. You can check to see what mode of name resolution a computer is set to use by typing **ipconfig /all** at a command prompt as shown in Figure 10.



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\byoshino>ipconfig /all

Windows IP Configuration

    Host Name . . . . . : byoshino
    Primary Dns Suffix . . . . . : lunt.com
    Node Type . . . . . : Hybrid
    IP Routing Enabled. . . . . : No
    WINS Proxy Enabled. . . . . : No
    DNS Suffix Search List. . . . . : lunt.com
                                     lunt.com

Ethernet adapter Wireless Network Connection:

    Media State . . . . . : Media disconnected
    Description . . . . . : Intel(R) PRO/Wireless 2200BG Network
```

Figure 10: IP configuration settings

When WINS clients need to resolve the NetBIOS name of another computer on the network, they use the following process [19].

1. The client checks to see if the name it wants to query is its own local NetBIOS name.
2. The client checks its local NetBIOS name cache.
3. The client forwards the NetBIOS query to its configured primary WINS server. If the primary WINS server fails to respond, the client will attempt to contact any other WINS servers it is configured with in the order they are listed.
4. The client broadcasts the NetBIOS query to the local subnet.
5. If it is configured to use one, the client checks the LMHOSTS file for a match to the query.
6. The client tries its HOSTS file and then a DNS server, if it is configured with one.

WINS Server Replication

Since WINS is a distributed database, in order to achieve database consistency, WINS servers need to replicate with each other on a regular basis. If a company has multiple physical locations, they may choose to install a WINS server at each office. If the clients at each office are configured to register with their local WINS server, administrators will need to create a replication topology to ensure every client is able to resolve the NetBIOS name of every other client on the network. These replication partnerships take the form of either *push* or *pull* replications. In a push replication, a WINS server will notify its replication partner based on the frequency of updates made on the WINS server itself [20]. Pull replications are instead based on an amount of time called a replication interval [20]. Administrators can adjust both of these values in the replication properties dialog box as shown in Figures 11 and 12.

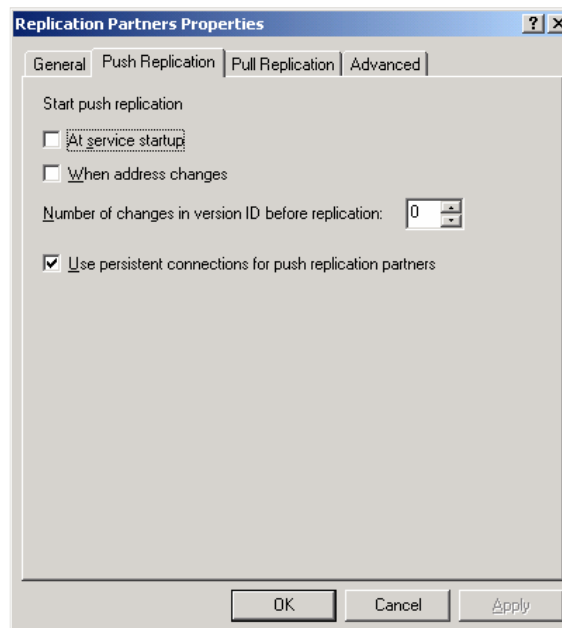


Figure 11: WINS Server Push replication settings

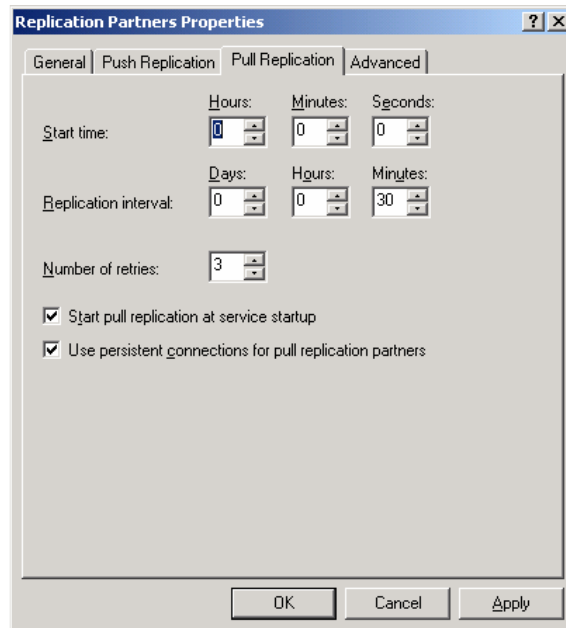


Figure 12: WINS Server Pull replication settings

There are a number of aspects that should be taken into account when planning WINS server replication. One of the most important of these is fault tolerance. Even very small networks should have at least two WINS servers installed. Windows 2000 and XP clients can be configured with multiple WINS servers to use for name registration and name resolutions. Of course, in order to be of any use, these two servers need to replicate with each other to ensure each of them has a complete database.

Additionally, administrators need to decide on an acceptable *convergence time*. The convergence time is the time required to replicate a new WINS database entry from the WINS server that owns it to every other WINS server on the network [20]. In small networks, this time may be minimal, but in global networks, this time may be measured in hours. Along with the replication interval options we previously discussed, administrators can reduce the convergence time by designing an efficient replication topology. In most cases, a hub-and-spoke model can be used as shown in Figure 13. It is up to the administrator to determine what amount of convergence time is acceptable.

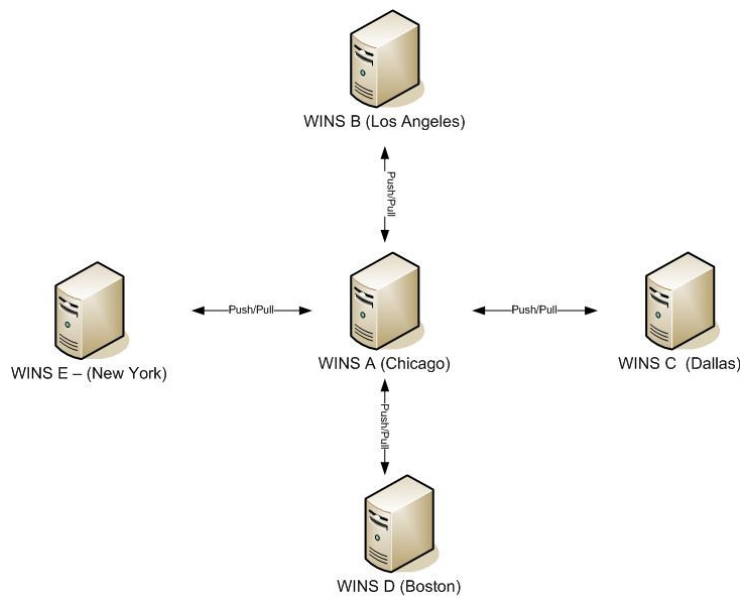


Figure 13: Hub-and-spoke WINS replication

Finally, while WINS was developed to reduce the amount of broadcast traffic on the network, administrators need to take the available bandwidth into consideration when planning replication topologies.

Automatic Partner Configuration

Windows 2000 Server and Windows Server 2003 WINS servers offer administrators an automated partner configuration service. If configured in this manner, WINS servers on the network discover each other and create replication partnerships automatically. These partnerships are of the push/pull type [20], though later we will discuss that configuring replication relationships manually is often the better alternative.

WINS Server hardware recommendations

The Windows Server 2003 WINS development team demonstrated that a computer with a single Intel Pentium II 350 MHz processor, 128 MB of RAM, and a standard IDE hard drive was able to handle an average of 300 name registrations per second and an average of 350 name queries per second [17]. Additionally, Microsoft suggests that a conservative estimate is one WINS server and one backup for every 10,000 computers on the network [17]. While the lab test hardware specs are ancient by today's standards, administrators need to take care in planning their WINS server hardware deployments to ensure that their WINS infrastructure remains stable and operational.

Like DHCP, the WINS service requires frequent and intensive disk activity. Microsoft suggests using a server with a RAID solution to improve disk-access times [21], but in most installations, this probably won't be necessary. Like the DHCP service, WINS servers have built-in counters that you can load into performance monitor to evaluate your hardware. In addition, you may choose to adjust your Burst Handling configuration depending on your network needs. Burst Handling will be discussed later on.

Finally, since your WINS server or servers are a critical piece of the network infrastructure, every WINS server should be configured with a static IP address.

Ways a WINS server can be attacked

In order to secure a WINS infrastructure, we need to understand how WINS can be attacked. In this section we will review some of the common attack vectors used to compromise WINS. Later, we will discuss some of the best practices administrators can use to secure WINS.

Unauthorized modification of WINS server configuration

As with any other critical service, if an attacker is able to gain administrative control of the WINS server, there is no end to the amount of damage they can do. They can modify replication topologies, add and/or remove records from the database, or even decommission the WINS server completely.

DoS attacks on WINS servers

A simple way to attack a WINS server is to perform a DoS attack against the server. An attacker can either compromise an existing machine on the network, or attack the server remotely via a wireless network. By sending numerous name registration or name resolution requests the attacker can attempt to overload the queue and prevent the server from responding to legitimate requests.

Attacking Replication between WINS servers

WINS is a distributed database. Depending on the size of your network, you may have a number of WINS servers servicing clients on different parts of your network. For instance, your company might have multiple locations; one in Chicago and another in the suburbs. In this case, you might install one WINS server at the Chicago office and other WINS server in the suburban office and configure the two as replication partners. By default, clients will register with the first WINS server they are configured with in their TCP/IP properties. If an attacker can succeed in preventing replication, clients in the Chicago office will not be able to communicate with the clients in the suburban office since the Chicago office WINS server will not contain the NetBIOS records for the clients in the suburban office. One method an attacker can use to prevent replication between WINS servers would be to perform a DoS attack against one of the

servers. Yet another method would be to gain control of a WINS server and modify the replication configuration of a particular WINS server.

False NetBIOS record registration

When a client attempts to register its host and group records with a WINS server, the WINS server will check to see if any current records exist with the same name. If a record exists, the WINS server will attempt to contact the current owner of the record. If the owner does not respond, the WINS server will replace the old record with the new one [5]. An attacker can attempt to hijack a record by performing a DoS attack on the original client so that it is unable to respond to the WINS server's request to defend its ownership of the record.

Registration of Incorrect records

Although host and group records are different from each other, an attacker can attempt to create a situation where they register a host record with the same name as a group name. For instance, if a hacker succeeds in registering a host name of SALES in a WINS database that also had a group record for a domain called SALES, they can prevent domain authentication requests since clients will be unable to locate the proper domain records to find an authentication server [5].

Best Practices

Fortunately, there are a number of steps administrators can take to harden their WINS infrastructure against attack. As with the section on securing DHCP, not all of these recommendations will apply to all installations. It is up to the administrator to determine the particular needs of their organization and proceed accordingly.

Use the default settings

The "out of the box" settings for the WINS server should be used in most WINS server installations. The preconfigured settings provide optimal security and should only be altered when absolutely necessary [21].

Physically secure the WINS server

Regardless of what steps an administrator might take to secure a WINS infrastructure, unless they have taken steps to ensure the physical security of the server, the configuration aspects of hardening a WINS server are worthless.

Audit membership of Administrative Groups

Members of the local Administrators group and the Server Administrators group on a WINS server computer have full control of the WINS server configuration console [5]. Membership of these groups should be reviewed on a regular basis to ensure that no unauthorized personal have rights to administer the WINS servers.

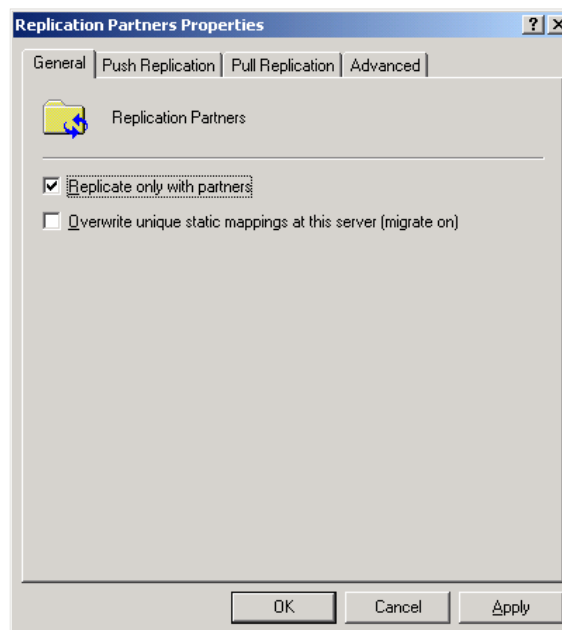
Validate WINS server replication topology

Administrators should perform regular checks to ensure that networks with multiple WINS servers have the correct replication settings. This includes not only replication partner settings, but bandwidth considerations as well.

Use static entries for critical NetBIOS applications

To prevent an attacker from attempting to hijack a record with a false update, an administrator might choose to use a static record for a critical NetBIOS name. Administrators can ensure these records are not overwritten on a Windows Server 2003 WINS server by clearing the checkbox “Overwrite unique static mappings at this server (migrate on)” as shown in Figure 14.

Figure 14: Protecting static entries from being overwritten



Eliminate NetBIOS Applications

While this suggestion is included in the Windows Server 2003 resource kit [5] NetBIOS applications will be around for a while. Even the latest version of Microsoft Exchange Server still requires NetBIOS name resolution for full functionality.

Use detailed logging to troubleshoot problems

Although detailed logging can impose a performance burden on a WINS server, it can be useful in troubleshooting WINS server problems. For instance, you can review the event logs generated by the WINS server to look for Burst Handling events.

Bursts occur when a large number of clients try to register their NetBIOS names simultaneously [22]. The Burst Handling feature (Figure 15) uses a threshold

value, which by default is 500 [22]. If more than 500 clients attempt to register their names at the same time, the WINS server goes into Burst Handling mode and any additional clients are sent positive responses by the WINS server. This response includes a varied, but shorter than usual Time-to-Live (TTL). This helps regulate the load on the WINS server and slows the rate of name refresh and renewal rates.

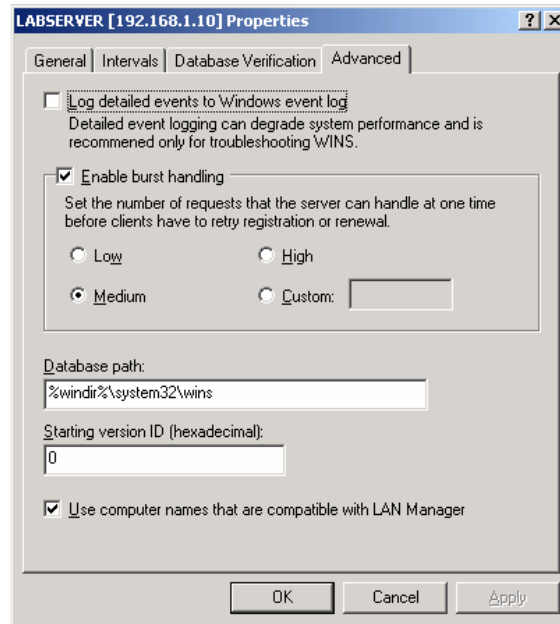


Figure 15: WINS Server burst handling configuration

While a burst handling event may be a sign of a denial of service attack, it can also mean that the WINS server is underpowered.

Keep the number of WINS servers to a minimum

A single WINS server can adequately service up to 10,000 clients [17]. The only reason you would need to add more WINS servers is if you wanted to add fault tolerance or if your physical network topology dictates their need. Either way, you should strive to keep the number of WINS servers to a minimum, thereby reducing administrative overhead and the potential for database corruption. In the event your network requires multiple WINS servers, Microsoft suggests the use of a hub-and-spoke configuration which offers the best convergence time and least chance for database corruption [21].

Manually configure WINS replication partners

If you keep the number of WINS servers to a minimum, it is not that difficult to manually configure your replication topology. Although small networks may choose to use the automated partner configuration feature, configuring them manually allows you to ensure that all replication partners receive a complete database and that only authorized servers are allowed to participate in replication.

Keep the WINS server computer up-to-date with patches

Since the WINS service runs on top of the server operating system, you need to make sure that the server is kept up to date with all of the Microsoft patches and service packs. Any vulnerability that can be exploited in the operating system, whether it is related to WINS or not, can affect the WINS Server service. Fortunately, tools such as Windows Update and the new Windows Server Update Services (WSUS) can help administrators ensure their server computers are up to date.

Secure your WINS server with IPSec

In installations that require very high security, administrators can use IPSec filters to limit the traffic accepted by a WINS server [5]. Using the rules listed in Table 4, you can ensure that only WINS related traffic reaches the WINS server. Note that these rules assume that the server functions only as a WINS server and a domain member.

Table 4: IPSec rules to protect a WINS Server

Service	Protocol	Source Port	Destination Port	Source Address	Destination Address	Action	Mirror
One Point Client	ANY	ANY	ANY	ME	MOM Server	Allow	Yes
Terminal Services	TCP	ANY	3389	ANY	ME	Allow	Yes
Domain Member	ANY	ANY	ANY	ME	Each DC's IP address	Allow	Yes
WINS Resolution Server	TCP	ANY	1512	ANY	ME	Allow	Yes
WINS Resolution Server	UDP	ANY	1512	ANY	ME	Allow	Yes
WINS Replication Client	TCP	ANY	42	ME	WINS replication partner	Allow	Yes
WINS Replication Client	UDP	ANY	42	ME	WINS replication partner	Allow	Yes
WINS Replication Server	TCP	ANY	42	WINS replication partner	ME	Allow	Yes
WINS Replication Server	UDP	ANY	42	WINS replication partner	ME	Allow	Yes
All inbound traffic	ANY	ANY	ANY	ANY	ME	Block	Yes

Configure your WINS servers to point to themselves

WINS servers themselves need to register their own unique and group NetBIOS names. Each WINS server should be configured to point to itself in its TCP/IP properties to prevent errors.

Perform regular backups of the WINS database

WINS uses the Jet database format to store its data [23]. The files listed in Table 5 are used by WINS and stored in the %systemroot%\System32\wins directory:

Table 5: WINS Server database files

Wins.mdb	This is the WINS service database file which contains two tables: an IP address-to-Owner ID mapping table and a Name-to-IP address mapping table
Winstmp.mdb	temp file used as a swap file during index maintenance operations
J50.chk	A checkpoint file used to indicate the location of the last information successfully written from the transaction logs to the database
J50.log and j50####.log	files of all transactions done with the WINS database
Res#.log	Reserved log files used when the server runs out of space

The WINS database can easily be backed up through the WINS console shown in Figure 16. Administrators should be sure to regularly backup the WINS database as part of their disaster recovery plan.

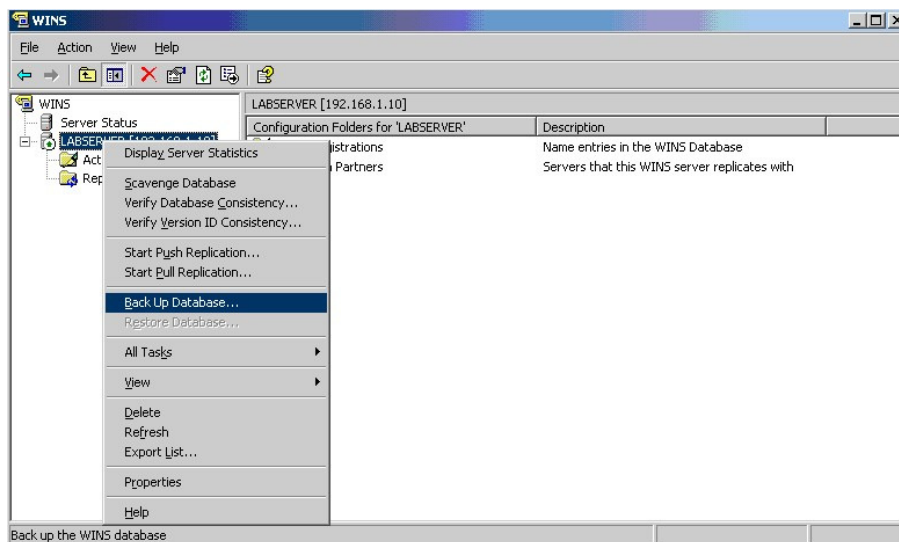


Figure 16: Backup up the WINS database

Domain Name System

Whereas as earlier editions of Microsoft networks were built on WINS, Windows 2000 and later operating systems were built to use DNS, or the Domain Name System, as their name resolution service. Introduced in 1984, DNS was born in the early days of the Internet [24]. Today, DNS is a globally distributed hierarchical database of hostnames and IP addresses. And while administrators of NT4 domains could run a fully functional network with little knowledge of DNS, today's Active Directory administrators are required to have an intimate understanding of DNS. Not only is it the backbone of Active Directory; DNS is a crucial service required for the function of the Internet.

Have you checked DNS?

DNS plays such an important role in Active Directory networks, that the question posed above is usually one of the first troubleshooting steps offered when asking for help with an Active Directory networking problem. Clients make use of DNS servers for everything from resolving World Wide Web addresses to finding a domain controller with which to authenticate. If your Windows XP workstation can't log on to the domain, it's probably a DNS issue. If you want to promote a new Windows Server 2003 computer to a domain controller but run into an error, it's probably a DNS issue. If your customers can access your web site by using the IP address, but not by URL, you can take a guess where the problem lies: DNS!

DNS is such a complex topic that entire books have been written about it [30], [31]. In the next section, we will briefly review of some of the basic concepts crucial to understanding the role DNS plays in an Active Directory network. This is certainly not meant to be a complete discussion of how DNS works and the reader is expected to have a strong understanding of the basic concepts of DNS. Nevertheless, this review should help reinforce the reasons why DNS needs to be secured and the steps to do so.

DNS Basics

If a DNS server is not available when you try to create the first domain controller for the first domain in the first tree of your first forest, the domain controller promotion process will not proceed until you either install a DNS server or ask it to build one for you. This example alone should drive home the fact that DNS is required for nearly every operation of Active Directory.

There are basically two components to DNS, the DNS client service and the DNS server service. DNS clients make use of DNS servers to resolve host name queries as well as register their own host name records. Every computer on an Active Directory network will need to be configured with at least one DNS server. These DNS servers, along with being DNS clients themselves, allow clients to

locate other clients on the network. For instance, when a Windows XP workstation boots up and attempts to find an Active Directory domain controller with which to authenticate, it contacts its preferred DNS server. The DNS server will then check its database and return the information to the client.

Similarly, after logging on if the user wants to perform the electronic equivalent of reading the sports section at work, they can point their web browser to, for example, www.espn.com. The DNS client on their machine will again query its preferred DNS server for the IP address of that web site. This query will most likely trigger an entire series of queries resulting in the client computer finally getting an answer to its question.

In an Active Directory network, the computers that can serve as DNS servers will be either Windows 2000 Server or Windows Server 2003 computers. That's not to say your DNS infrastructure must be based off of Microsoft's implementation of DNS. For instance, a fully functional Active Directory can be built using current versions of BIND running on a Linux machine or Lucent's QIP software. In fact, large enterprises may be forced to build their Active Directory installations atop non-Microsoft DNS servers. This is often because the business groups that manage the directory services are different than the groups that run the network infrastructure services. The infrastructure group may have a well-established and stable DNS infrastructure running BIND, and there is little chance that these UNIX gurus would be willing to scrap it all and install Microsoft servers. Although you won't be able to make use of a number of security features offered when running DNS on Windows Server boxes, you'll have very little trouble getting Active Directory up and running. The important point is that whatever non-Microsoft DNS server you choose to install, it needs to support three key capabilities:

- Support for dynamic updates
- Support of SRV records
- Support for names with underscores

Of course, the DNS Server service on Windows 2000 Server and Windows Server 2003 computers support all of these features.

DNS is a hierarchical database

We mentioned before that DNS is a globally distributed hierarchical database. Active Directory administrators will not only have to concern themselves with securing their own portion of the namespace but will also need to ensure the security of communications between their internal namespace and the rest of the hierarchy. For instance, when clients on the internal network need to resolve host addresses on the Internet, a number of DNS-related communications will need to take place. This includes queries not only between the local client and its preferred DNS server, but also between the preferred DNS server and the other authoritative DNS servers out on the Internet.

DNS replication

In order to provide fault tolerance and load balancing, you will want to set up multiple DNS servers to support your Active Directory. There are a number of different replication types. Primary and Secondary zones support a master/slave relationship. For instance, you can only add, modify, and delete records in a primary zone. Secondary zones are read-only zones. They are copies of primary zones and get their data by transferring zone data from primary zones. You can also integrate the zone with Active Directory. These zones are appropriately called Active Directory-integrated zones, and offer a number of advantages such as increased security and multimaster replication.

Caching

Caching is another major responsibility of DNS servers. When a DNS server receives an answer to a query, it will cache the record for a predetermined amount of time. If a second query comes in for the same record, and the record is still considered valid, the server will not have to use valuable bandwidth to process the query again. Instead, it can return the answer out of its cache. The caching aspect of DNS servers is so important that some network administrators install DNS servers strictly as caching-only DNS servers. In other words, these DNS servers are authoritative for no zones of their own. Instead, they exist solely to reduce the time and bandwidth needed to resolve queries.

A final note

DNS is framework upon which Active Directory is built. Although administrators can implement a number of the security practices discussed in the upcoming sections with a limited understanding of how DNS works, in order to truly benefit, the reader should have a strong understating of how DNS works and the name resolution process. Administrators with a strong understanding of DNS will be better able to make the proper decisions regarding how to secure their DNS infrastructure.

DNS Server Hardware Recommendations

The hardware recommendations for DNS servers are much more complex than that of DHCP or WINS servers. There are a number of factors that need to be considered when deciding not only the hardware requirements for a particular server, but also how many of each type of server the environment will need. For instance, administrators will need to decide how many zones the DNS server will host, how large each zone will be, and how many queries the DNS server is expected to receive. Regardless of your particular environment, planning should begin with a review of Microsoft's recommendations.

When Microsoft developed DNS for Windows Server 2003, the DNS development team tested the DNS server service on readily available hardware

to compile statistics that could be used as a benchmark for estimating DNS server performance [25]. The test configuration was configured as follows:

Processor: One (1) Intel Pentium III 733 MHz processor
RAM: 256 MB
Hard drive: 4GB

The server, running Windows Server 2003 had only the DNS server service installed. During the more-than-four-day test, the server was subject to both name resolution queries and dynamic updates requests simultaneously. The results of the test can be seen in Table 6.

Table 6: DNS Server development team test results

Queries/sec	Dynamic Updates/sec	Processor Utilization
9500	1300	75%

The zone types used were dynamic standard primary zones. Note that if Active Directory-integrated zones were used, the rate at which the server could process updates would be decreased due to the need to read and write to the Active Directory database [25]. Additionally, if only secure dynamic updates were allowed, the update rate would also decrease [25].

This test was not meant to specify the exact performance specifications of a DNS server running Windows Server 2003 but instead to provide a point of reference so that administrators could begin to plan server capacity. Windows Server 2003 includes over 60 performance counters in System Monitor that can be used to measure and gauge multiple aspects of server function [25].

As mentioned earlier, before administrators can begin to plan the hardware for their DNS server infrastructure, they will need to perform a review of their needs first. Most environments will have multiple DNS servers. Some will host multiple zones while others might serve as caching-only DNS servers. Administrators will also need to decide if they will implement Active Directory-integrated DNS zones, since this will dictate that the DNS service be installed on a domain controller.

Often, adding RAM to a DNS server results in the most noticeable performance benefits [26]. This is because the DNS Server service fully loads all the zones it is configured with in to memory at startup. Fortunately, there are four performance counters dedicated to surveying memory performance [25].

Common types of Attacks on DNS

Securing DNS begins with first understanding how DNS can be attacked. In this section we will review some of the attacks common to DNS. Later, we will discuss some best practices administrators can use to secure DNS.

Denial of Service attacks

As we mentioned before, a stable and available DNS infrastructure is critical for Active Directory to function. If DNS servers are not available, then clients will not only have trouble resolving host names of clients out on the Internet, but they can also have difficulty in finding hosts and services on the local intranet. For instance, unless a local DNS server is available to answer queries for the SRV records of domain controllers, clients will not be able to log on. The DoS attack can occur in any number of ways, such as a compromised client on the network attempting to overwhelm the local DNS server with DNS name resolution requests.

Unauthorized Modification of DNS Records

The inherent support of dynamic updates can lead to the unauthorized modification of DNS records. Unless the DNS zones are properly secured, it might be possible for an attacker to register or alter existing records so that clients attempting to contact certain hosts on the network are unsuccessful or possibly even redirected to hosts under the control of the attacker. In addition, if an attacker is able to gain local control of the server, they can attempt to add static DNS records to the DNS zones.

Cache Pollution

Similar to the above attack, an attacker can attempt to add false information to the DNS server's cache in order to redirect the clients to a system under the hacker's control. As we discussed earlier, Windows 2000 Server and Windows Server 2003 DNS servers will check to see if the queried name exists in its local cache. If the record exists in the cache, the DNS server will return the cached information instead of using the hierarchy to resolve the name. Therefore, if an attacker is able to "pollute" the cache with false information, the DNS server will return these false records instead of contacting the authoritative DNS server for the zone.

Unauthorized Zone Transfers / Footprinting

The DNS zones that support an Active Directory network contain an enormous amount of information. While the hostnames and IP addresses of your client workstations may not seem to be the most confidential information in your organization, it is the other record types, specifically SRV records, which you wouldn't want an attacker to know. These records can provide information such as which servers in your network function as domain controllers, global catalog servers, and so on. If an attacker can successfully footprint your network, they will know exactly which servers to target their attacks on.

Using DHCP to provide false name server information to clients

This was discussed in the DHCP section, but it is worth mentioning here again. If an attacker is able to compromise the client's preferred DNS server settings, they

can successfully redirect the client to a DNS server under the control of the hacker without any detection by the client.

Best Practices

Ensure the physical security of your DNS servers

The third “Immutable Law of Security” as described by Microsoft’s Security Response Center is that “If a bad guy has unrestricted physical access to your computer, it’s not your computer anymore” [27]. We discussed this in the sections on DHCP and WINS, but it is such an important topic we will touch on it again here. Whereas your DHCP and WINS servers hold critical and confidential information about your network, Active Directory’s dependence on the DNS Server services makes them even more so. Should an attacker succeed in compromising, stealing, or even destroying your DNS servers, Active Directory will cease to function. An attacker may take a less obvious route and simply copy the zone files to a floppy disk or USB drive. As discussed earlier, the zone files contain all the information an attacker could want to footprint your network. Finally, in situations where administrators make use of Active Directory-integrated DNS zones, the DNS Server service resides on domain controllers. Should one of these servers become physically compromised, you have few choices for recovery. A common joke is that the tool you run on your domain controllers after a physical attack is FDISK, alluding to the fact the entire domain will have to be “nuked” and rebuilt from the ground up.

Use Active Directory-integrated zones with secure dynamic updates

Instead of using primary zones, Microsoft suggests administrators use AD-integrated DNS zones whenever possible [5]. In addition, these zones should be configured to allow only secure updates. Along with providing fault tolerance, AD-integrated zones offer a number of security advantages. Along with prohibiting non-secure dynamic updates, AD-integrated zones also allow you to define access control lists to control granular access to both complete zone files as well as individual records [32].

Secure the DNS Server against cache pollution

Both Windows 2000 and Windows Server 2003 allow the administrator to implement DNS cache protection on DNS servers. This option, found in the Advanced tab of the DNS server’s properties page (Figure 17) forces the server to inspect the responses from other DNS servers. If the DNS server receives a response to a resource record query from a server that is not authoritative for that DNS zone, the server will not cache the record. By default, cache pollution protection is enabled by default on Windows 2000 SP3 and later DNS servers [28].

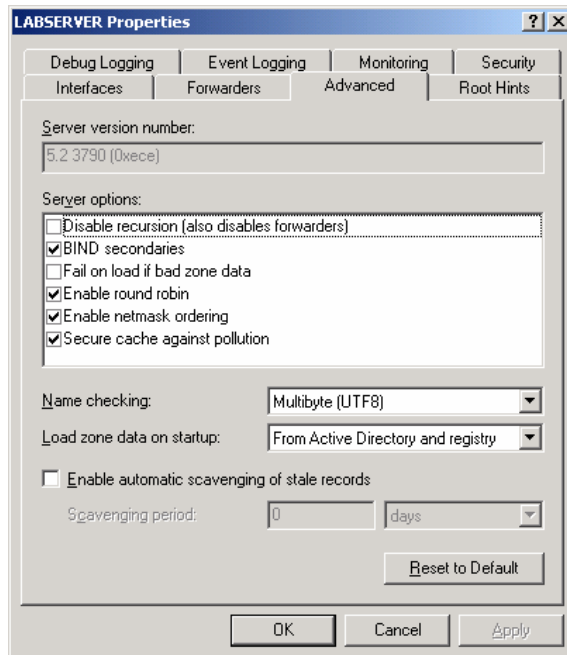


Figure 17: Cache Pollution protection setting

Slave your internal DNS servers

Earlier we discussed the caching aspect of the DNS Server service. We can also use this concept to increase the security of our DNS infrastructure. Suppose an attacker sets up a DNS server for some domain. This person configures the DNS server to exploit some vulnerability in the DNS Server service such as a buffer overflow or similar threat when responding to queries. Then they send out a spam e-mail with a link to their website. If the attacker can succeed in getting just one of your internal clients to click on the link, they will succeed in having your internal DNS server do a name resolution request to the malicious DNS server. If you have been lax about your patching policy, the attacker may be able to gain control of your machine.

In order to prevent this, you can set up an additional DNS server outside of your internal network; perhaps sitting in the DMZ. This would be a stand alone machine and not a member of the Active Directory domain. This server is a hardened DNS server which contains no zone files of its own. Its only responsibility is to perform name resolution requests that have been sent to it by your internal DNS servers. You then configure your internal DNS server to forward all of the name resolution requests to this server. Additionally, you prohibit the server from performing recursive requests in the event the server is unresponsive or unavailable.

Even if an attacker is able to compromise the hardened DNS server in the DMZ, they will not have been able to attack any of your internal DNS servers. Note that prohibiting the internal DNS servers from performing recursive queries is critical. When using forwarders, administrators can configure a time-out value so

that name resolutions can still occur even if the forwarder is down. Without disabling this feature, an attacker could use it as an exploit. For example, the attacker might suspect your network is using a forwarder and configure their DNS server to respond to the initial name resolution request slowly, hoping the initial query will time out. If the next query comes from a different IP address in your address block, the attacker might assume that the internal DNS server gave up on the forwarder and chose to resolve the name itself. The attacker's DNS server will then respond quickly with the malicious reply.

In order to “slave” a DNS server, we configure it to use a hardened forwarder, or for better fault tolerance, forwarders, and then prohibit it from performing recursive queries on its own in the event the forwarder is down or otherwise unavailable. To prevent the DNS server from performing recursive queries on its own, we check the box labeled “Do not use recursion for this domain” on the Forwarders tab of the DNS Server properties dialog box (Figure 18).

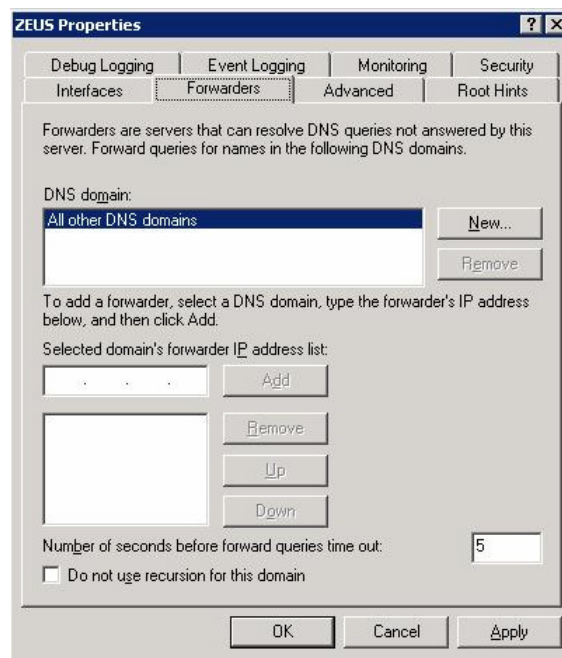


Figure 18: Disabling recursion on a DNS server

Use “Split-Brain” DNS

The concept of Split-Brain DNS can be complicated, but understating it is a necessity for any DNS administrator. Using Split-Brain DNS allows administrators to support their internal Active Directory network without risking any exposure of the zones on the DNS servers to the outside world. Before discussing how Split-Brain DNS works, let's discuss the alternative; having one DNS server or set of DNS servers which support the internal network and respond to external queries as well.

Let's assume we have a simple single Active Directory forest which contains a single domain named yoshino.com. This domain has two domain controllers and two DNS servers. The DNS servers are member servers of the domain. One of them, called dns1.yoshino.com, holds the primary zone for the yoshino.com domain while the other, dns2.yoshino.com, holds a secondary zone. All of the computers in the yoshino.com domain have their preferred and alternate DNS servers set to dns1.yoshino.com and dns2.yoshino.com respectively. Additionally, since dns1.yoshino.com is the authoritative DNS server for the yoshino.com domain, all computers register their resource records with dns1.yoshino.com.

Yoshino Corp. also has an Internet presence. They registered the yoshino.com domain and gave their registrar two of their public IP addresses to use as the two name server computers for the yoshino.com domain. They also created two rules on their firewall to forward name resolution requests coming in on those IPs to their two internal name servers.

When someone on the Internet wants to browse www.yoshino.com or send mail to someone in the yoshino.com domain, their local DNS server will eventually resolve the name using either dns1.yoshino.com or dns2.yoshino.com. While this configuration is fully functional, it poses a number of major security risks, such as exposing the entire DNS domain to the Internet. In addition, a situation is created where hosts on the Internet are able to communicate directly with critical servers on the intranet.

Instead, Yoshino Corp. should implement a split-brain DNS architecture. In this set up, Yoshino Corp. will install two additional DNS servers outside of their firewall. These DNS servers can be hardened Windows 2000 Server or Windows Server 2003 computers which are not members of the Active Directory domain. These servers are called ns1.yoshino.com and ns2.yoshino.com. ns1.yoshino.com will hold a primary DNS zone for the yoshino.com domain and ns2.yoshino.com will hold a secondary zone. The key point to understand here is that the internal DNS servers know nothing about the external DNS servers and vice-versa. As far as the internal DNS servers are concerned, they are the only name servers in existence for the yoshino.com domain. The same goes for the external DNS servers.

The internal clients will continue to use the internal DNS servers for all their name resolution needs. As for the outside world, Yoshino Corp. will have the registrar point to ns1.yoshino.com and ns2.yoshino.com. The zones on the external DNS servers will know nothing about the Active Directory domain. In fact, they will only contain a handful of records such as a host records for www, ftp, and a mail exchange record.

An alternate split-brain DNS architecture also exists. For instance, a number of domain name registrars and ISPs offer DNS domain name hosting as part of

name registration or Internet access packages. For instance, suppose that Yoshino Corp. decided to acquire Internet access through Covad Communications, a nationwide provider of DSL and T1 technologies. Covad offers DNS zone hosting to its customers, and instead of hosting their own external DNS servers, Yoshino Corp. chose to allow Covad to host their external zone. All the administrators of Yoshino Corp. would have to do is tell their registrar that the authoritative name servers for the yoshino.com domain were the name servers provided by Covad and then add the necessary records to the zone. This provides a number of benefits. It reduces the amount of computer hardware and software needed to support the external yoshino.com zone, and it transfers the responsibility of securing the zone to someone else, a good move as long as the third-party is trusted.

Allow zone transfers to authorized DNS servers only

Windows 2000 Server and Windows Server 2003 DNS servers allow you to restrict zone transfers between servers. Administrators should be sure to configure which servers are allowed to participate in zone transfers by using the Zone Transfers tab in the properties of the DNS zone (Figure 19).

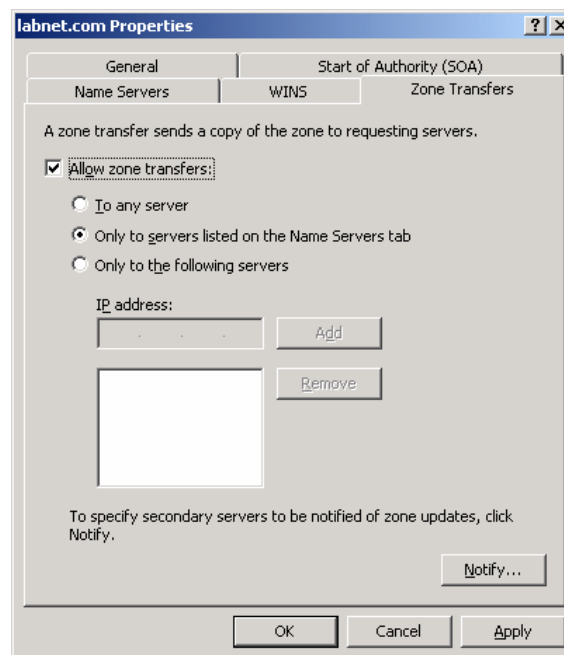


Figure 19: Restricting Zone Transfers

Restrict administrative access to the DNS server

There are a number of groups that have administrative access to the DNS Server service. Besides the local Administrators group on the DNS server computer, the DNSAdmins, Enterprise Admins, Domain Admins, and the Enterprise Domain Controllers groups all have administrative access to the DNS server service [5]. Administrators should conduct regular audits to ensure only the proper accounts are members of these groups.

Periodically review the DNS server log

The DNS server offers a number of logging options as shown in Figure 20. These logs can provide valuable information about the health of the DNS Server service as well as alert administrators of potential problems. Administrators should leave the default value of auditing all events.

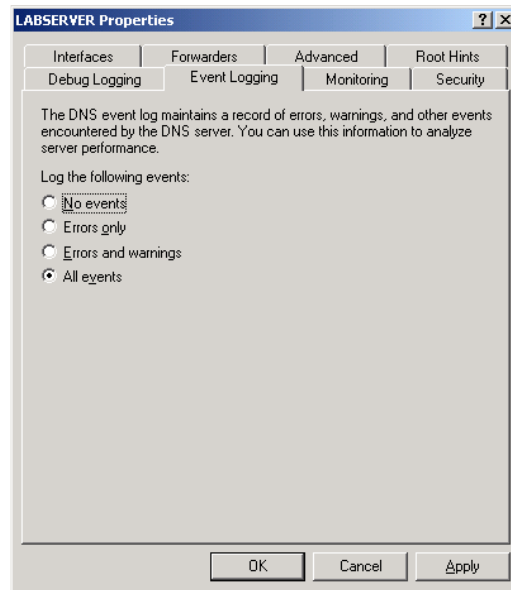


Figure 20: DNS Server logging options

In addition, Microsoft suggests that the size of the DNS Service log be a minimum of 16MB on all domain controllers and that the logs be configured to "Overwrite Events As Needed" as shown in Figure 21 [5].

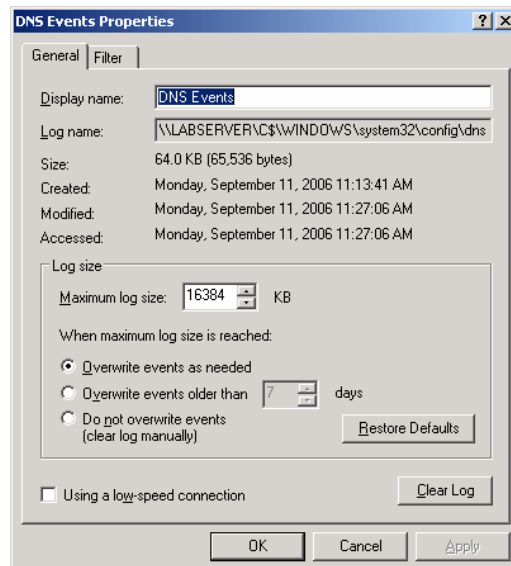


Figure 21: DNS log file handling options

Configure your firewall to allow DNS traffic between your internal and external DNS servers only

In order to ensure that your internal DNS servers remain as secure as possible, your firewall should be configured to ensure the only DNS traffic that passes through it is between your internal DNS servers and the external forwarders. This will ensure that external computers on the Internet will be unable to attempt zone transfers against your internal DNS servers and also prevents internal clients from using external DNS servers.

Protect Primary and Secondary Zone files with NTFS

If AD-integrated zones are not an option, administrators should ensure that the DNS zones files of primary and secondary zones are stored on an NTFS volume. NTFS volumes allow administrators to use granular permissions to control access to the files. The zones files should be secured so that only the System group is allowed Full Control [29]. By default, the zone files are stored in the %systemroot%\System32\dns folder.

Works Cited

- [1] (2005, January 21). DHCP best practices. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/75cd0e1f-f464-40ea-ac88-2060e6769f331033.mspx?mfr=true>
- [2] (2003, March 28). How DHCP technology works. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/8006f246-2029-4bad-b9f0-4f31a56b05901033.mspx?mfr=true>
- [3] (2005, January 21). Planning DHCP Networks. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/f/?en/library/3040afd1-e82b-4ded-8fcd-aa8fe021fcc11033.mspx>
- [4] (2005, January 21). DHCP performance monitoring reference. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/b08e1563-bee1-49f9-9b71-d145b4785b591033.mspx?mfr=true>
- [5] Smith, Ben & Brian Komar (2005). Microsoft Windows Security Resource Kit, Second Edition. Redmond, Washington: Microsoft Press.
- [6] (2005, January 21). Authorizing DHCP servers. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/9a4157c4-3c2f-4871-9ffe-7d405781f2cf1033.mspx?mfr=true>
- [7] (2003, March 28). Dhcploc overview. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/8fa42e83-ec08-4a9b-9057-8909f7ed433e1033.mspx?mfr=true>
- [8] Tulloch, Mitch (2004, July 20). DHCP server security (part 1). Retrieved August 29, 2006, from WindowsSecurity.com Web site:
<http://www.windowsecurity.com/articles/DHCP-Security-Part1.html>
- [9] (2005, January 21). Security information for DHCP. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/77d015fc-67fa-4ce9-86dc-3f22c1c535b61033.mspx?mfr=true>
- [10] (2005, January 21). The DHCP database. Retrieved August 29, 2006, from Microsoft TechNet Web site:

- <http://technet2.microsoft.com/WindowsServer/en/library/8cf0b3bf-0ea2-4dcf-a3b9-d71ba386f5e51033.mspx?mfr=true>
- [11] (2005, January 21). Backing up the DHCP database. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/252b4139-6a25-41c6-906e-812731d9475e1033.mspx?mfr=true>
- [12] (2005, January 21). Audit logging. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/753fcae1-8b02-48de-b2af-f431277cf72a1033.mspx?mfr=true>
- [13] (2005, January 21). Using DNS servers with DHCP. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/d0e19b57-c368-46c2-b017-caf25ae150ec1033.mspx?mfr=true>
- [14] (2005, January 21). Cluster support for DHCP servers. Retrieved August 29, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/5df3d4e9-e846-413a-bd9a-99645ac580991033.mspx?mfr=true>
- [15] (2000). Microsoft windows 2000 server TCP/IP core networking guide. Redmond, Wa: Microsoft Press.
- [16] (2006, March 30). Exchange server 2003 and exchange 2000 server require NetBIOS name resolution for full functionality. Retrieved August 30, 2006, from Microsoft Help and Support Web site:
<http://support.microsoft.com/?id=837391>
- [17] (2005, January 21). Planning WINS networks. Retrieved August 30, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/68713916-5b06-4201-9a9c-da1a670588ac1033.mspx?mfr=true>
- [18] Minasi, Mark, Christa Anderson, Michele Beveridge, C.A. Callahan, & Lisa Justice (2003). Mastering Windows Server 2003. San Francisco, CA: Sybex, Inc.
- [19] (2005, January 21). How WINS works. Retrieved August 30, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/054a2711-40c0-4cad-bbb2-3756c256043b1033.mspx?mfr=true>

- [20] (2005, January 21). Configuring WINS replication. Retrieved August 30, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/cc3b6bc5-78c3-4007-9c76-526c3deaab031033.mspx?mfr=true>
- [21] (2005, January 21). WINS best practices. Retrieved August 30, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/ed9beba0-f998-47d2-8137-a2fc52886ed71033.mspx?mfr=true>
- [22] (2005, January 21). Burst handling. Retrieved August 30, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/7ad27f84-cfae-4bce-8049-1841534043971033.mspx?mfr=true>
- [23] (2005, January 21). The WINS database. Retrieved August 30, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/58f80979-0ed5-40fb-881f-50a4f63bdbd61033.mspx?mfr=true>
- [24] (2003, March 28). How DNS works. Retrieved September 8, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/19a63021-cc53-4ded-a7a3-abaf82e7fb7c1033.mspx?mfr=true>
- [25] (2005, January 21). Monitoring DNS server performance. Retrieved September 8, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/5e81fbe2-764a-47c4-bc7a-0da6f447897b1033.mspx?mfr=true>
- [26] (2005, January 21). Server planning for DNS. Retrieved September 8, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/949f3a45-84e2-487f-80d7-bce184b28a061033.mspx?mfr=true>
- [27] 10 immutable laws of security. Retrieved September 8, 2006, from Microsoft TechNet Web site:
<http://www.microsoft.com/technet/archive/community/columns/security/essays/10imlaws.mspx?mfr=true>
- [28] (2005, April 12). Description of the DNS server secure cache against pollution setting. Retrieved September 8, 2006, from Microsoft Help and Support Web site: <http://support.microsoft.com/kb/316786/>
- [29] (2005, January 21). Checklist: securing your DNS infrastructure. Retrieved September 8, 2006, from Microsoft TechNet Web site:

- <http://technet2.microsoft.com/WindowsServer/en/library/77a3cf08-63b7-46e1-a2e5-b34d645a7dd41033.mspx?mfr=true>
- [30] Langfeldt, Nicolai (2000). The concise guide to DNS and BIND. Indianapolis, Indiana: Que.
- [31] Liu, Cricket, Matt Larson, & Robbie Allen (2003). DNS on windows 2003 server, third edition. Sebastopol, CA: O'Reilly.
- [32] (2005, January, 21). Active directory integration. Retrieved October 5, 2006, from Microsoft TechNet Web site:
<http://technet2.microsoft.com/WindowsServer/en/library/77a3cf08-63b7-46e1-a2e5-b34d645a7dd41033.mspx?mfr=true>