40555A Networking Fundamentals

Module 6: Name resolution

Contents

Learning objectives based on MTA exam objectives

#	Lesson title	Learning objectives	Exam objectives mapped
1	Overview of name resolution	• Describe name resolution. • Identify various methods for name resolution.	3.4.6 LMHOSTS file 3.4.5 HOSTS file
2	Resolving host names with DNS	• Describe hostnames. • Describe Domain Name System (DNS). • Explain how DNS name resolution works. • Configure DNS settings in the Windows 10 operating system. • Describe DNS zones and domains. • Identify different DNS zone types. • Identify types of common resource record.	3.4.4 Steps in the name resolution process 3.4.5 HOSTS file 3.4.4 Steps in the name resolution process 3.4.1 DNS 3.4.2 Resource records
3	Resolving NetBIOS names	• Describe NetBIOS. • Describe broadcast name registration, resolution, and release. • Explain how to resolve names with an LMHOSTS file. • Describe WINS.	3.4.4 Steps in the name resolution process 3.4.3 Windows Internet Name Service (WINS) 3.4.6 LMHOSTS file

#	Lesson title	Learning objectives	Exam objectives mapped
		- Implement NetBIOS name resolution in Windows Server.	

Module overview

For most of us, it's probably been a long time since we last entered a phone number into our mobile device before we called home. Instead, we just use a contact name that automatically references a number. It's the same with networking; we use computer names rather than logical numbering systems, like IPv4 or IPv6 addresses, to connect a web browser to a web server.

In this module, we'll examine *name resolution*, the process that resolves a name to the appropriate IPv4 or IPv6 address. As we'll discover, there are several ways in which computers can resolve names, and we'll learn more about at each of them.

Objectives

After completing this module, you will be able to:

- Identify the main methods of name resolution.
- Describe how to resolve host names with Domain Name System (DNS).
- Describe how to resolve network basic input/output system (NetBIOS) names.

Lesson 1: Overview of name resolution

Depending on the operating system your computers and devices use, there are several ways in which a name can be resolved into the corresponding IPv4 or IPv6 address. Before we get into that, let's discuss the basic name resolution process.

Objectives

After you complete this lesson, you will be able to:

- Describe name resolution.
- Identify various methods for name resolution.

What is name-resolution?

Computers can communicate over a network by using a name in place of an IP address. *Name resolution* is the process used to find an IP address that corresponds to a name, such as a hostname.

Note

Computers and devices often have several names. For example, Windows 10 computers have both a computer name and a hostname. We'll discuss this in more detail later.

The process of name resolution really consists of three steps. At a high level, they are: name registration, name discovery, and name release.

- Name registration. During this process, which Figure 1 depicts, a computer registers its names so other computers don't use the same name. Sometimes, a computer registers a name with a service that records the registered information. In other situations, a computer merely states its intention to use a name.

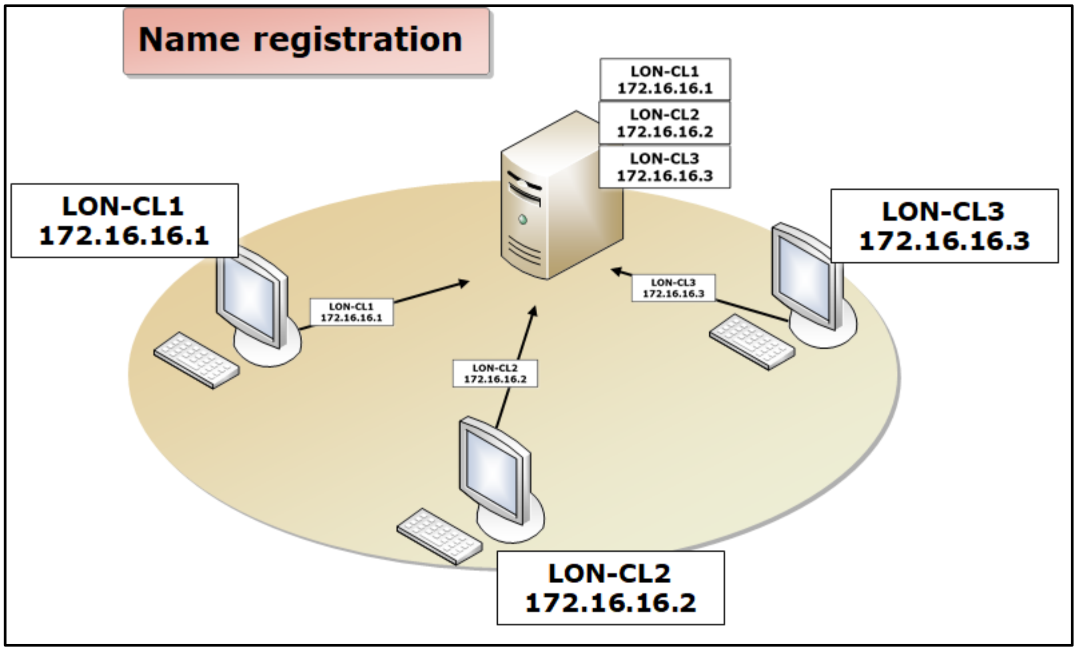


Figure 1. Name registration

Name registration occurs during computer startup, and occasionally thereafter; for example, when you reinitialize a network interface card (NIC).

Note

A computer that *states* its name does so using a broadcast or multicast message. You might remember that a broadcast is sent to every node on a network (usually the local subnet), while a multicast is sent to computers and devices that are registered to use a particular multicast group. For further information about the way broadcasts and multicasts work in TCP/IP, refer to Module 5 of this course.

- Name discovery. During the discovery phase, as Figure 2 illustrates, a computer or other device wants to resolve a name that it has to the corresponding IP address. The precise details of this stage vary according to the type of name and the method used to register names. Sometimes, a computer contacts a server device that holds a list of names and the corresponding IP addresses.

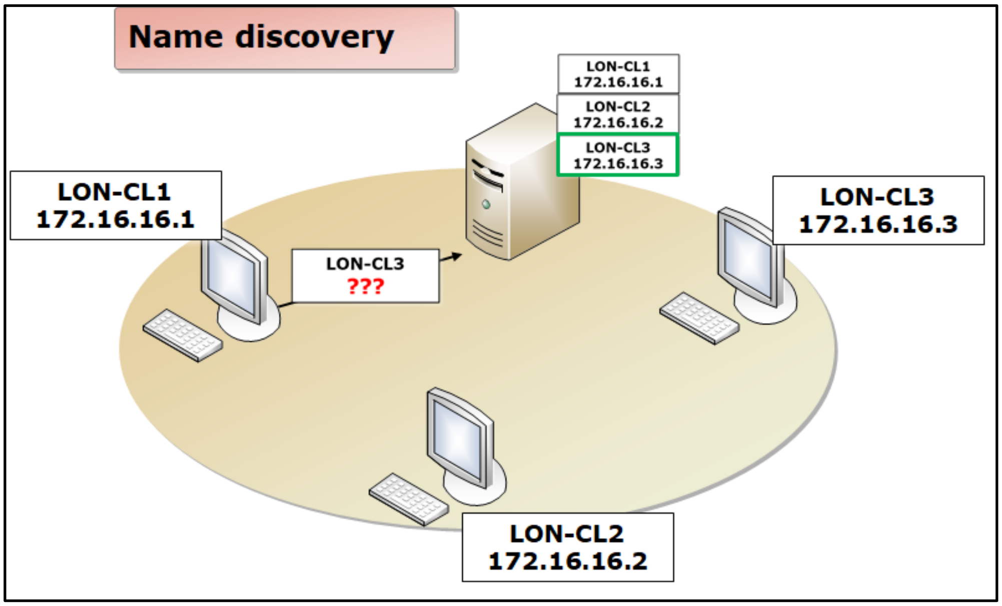


Figure 2. Name discovery

The server responds with the appropriate IP address. In other situations, the discovering computer broadcasts its requests (or uses a multicast) onto the local network. The appropriate computer device answers the request and provides its own IP address.

- Name release. When a computer is shutting down, as Figure 3 depicts, it has the option to release the name it has registered. One purpose of name release is to make the name available for another computer to use. But in reality, that’s never likely to be useful because all computers and devices (hosts) on an IP network should have unique names anyway. However, releasing the name has a secondary effect; it removes the name to IP address mapping from any server with which the name was registered. This can help to optimize name discovery because a querying computer can determine that a particular host is not online – because its name has been released.

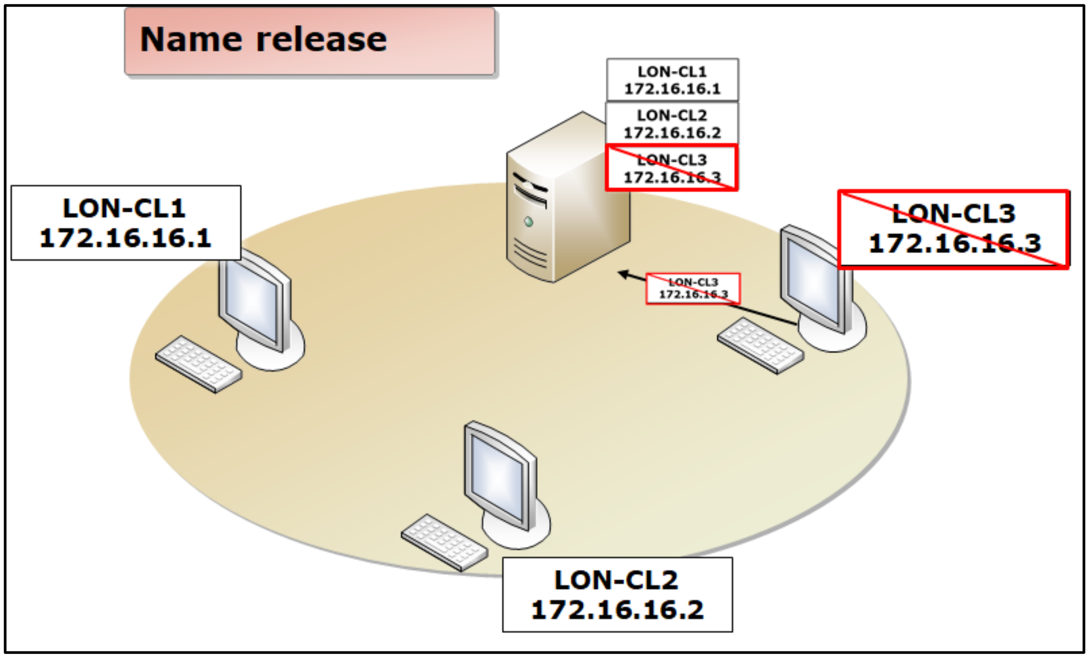


Figure 3. Name registration

Name-resolution methods

Over the years, a number of different methods were developed to perform name resolution and its associated stages of registration, discovery, and release. These methods include the following:

- Broadcast
- Link-Local Multicast Name Resolution (LLMNR)
- LMHOSTS file
- Windows Internet Naming Service (WINS)
- HOSTS file
- DNS

Broadcast

Broadcast name resolution developed with early network operating systems. Microsoft introduced a network called MS-Net back in early 1980s; this network operating system (NOS) was based on a proprietary IBM protocols and was designed to connect devices on a local area network (LAN) only.

Because it was only intended for use with LANs, MS-Net used a network protocol stack that did not support a network layer, and therefore didn't support routing. Broadcasts then were a suitable (or at least workable) solution for name registration, resolution, and release.

Note

In an IP-based network, broadcasts are not typically propagated by routers. In effect, this means that the maximum scope of a broadcast message is the local subnet. Using broadcast name resolution limits a computer to discovering the IP addresses of hosts in the local subnet only.

When a computer starts, it broadcasts its intention to use a specific name (or names). Any computer already configured with that name (or names) will challenge the registration, and the computer that is starting is unable to use that name. Typically, this results in failure of associated services. Computers that recognize the broadcast name registration message create or update a local table of names and if implemented on an IP network, the associated IP addresses as well.

Note

You might be wondering, “If MS-Net wasn’t based on IP, then what was the name being resolved into?” If you’re thinking that, kudos to you, because that’s a good question.

In pre-IP days, resolution was based on determining the media access control (MAC) address of the NIC. In IP-based networks, this step (IP to MAC) is managed by the Address Resolution Protocol (ARP) and is broadcast based. We’ll come back to ARP later.

When a computer shuts down, it broadcasts a name release message. Computers that recognize the broadcast name release message update their local name table.

Finally, when a computer wants to resolve a name it can check its local name table to verify whether if it has the name to IP mapping already. If not, then it broadcasts onto the local network asking which computer has the name it’s querying, and what the related IP address is.

As we have mentioned, the significant problem with the broadcast-based name resolution method is that it’s limited to the local subnet. This might be fine for a small office or home office network, but it’s unlikely to be useful for larger enterprise networks. In larger networks it’s also the case that routers are frequently used to connect subnets, both in extended LAN and wide area network (WAN) situations. As we have learned, routers don’t propagate broadcasts.

Having said that, the major advantage of using broadcast-based name resolution is that you don’t need to install a server to manage the name registration/release and resolution processes, because it’s handled by the client computers themselves.

LLMNR

When Microsoft introduced the Windows Vista operating system back in late 2006, it was fully integrated with networking and its primary network transport was IPv4. As we shall learn shortly, the preferred mechanism for managing name resolution on IP-based networks is DNS.

However, DNS requires a designated server—usually several servers—to provide a central point for registration, resolution, and release. For a small network, management and administrative overhead or deploying and maintaining DNS servers is high.

LLMNR is designed to enable small network users (typically on a single subnet) to perform name registration, resolution, and release without requiring a DNS or other name server. This is because LLMNR uses multicast communications over User Datagram Protocol (UDP). Although multicasts can transit routers, it's necessary to instruct the routers to propagate the specific multicast traffic. This means that as with broadcast-based resolution, LLMNR is restricted to a local subnet.

The advantage of using LLMNR is that you don't need to deploy and configure a DNS server. Therefore, consider using LLMNR for subnets that don't have a local DNS server. Because multicast traffic is restricted to the local subnet, name registration, release, and resolution traffic doesn't transit WAN links.

Note

By default, a client computer with Windows 10 installed will only use LLMNR as a means to resolve names when DNS resolution has failed. It's also important to note that LLMNR is enabled on Windows Vista and later Windows operating systems. You can disable LLMNR by editing the computer registry, or by using Group Policy settings. (However, the details of how to make these changes are out of the scope of this course.)

LMHOSTS file

Before we can start talking about LMHOSTS, and WINS, it's important that you know something about NetBIOS. NetBIOS is a session layer protocol developed for IBM in the early 1980s. It was implemented in Microsoft's early network operating systems, MS-Net, and LAN Manager. NetBIOS provides the following services:

- Name registration
- Session maintenance

Note

A *session* is a reliable connection between two systems for the transfer of data. NetBIOS provides for both session establishment and termination.

- Reliable, connection-oriented communications at the session layer
- Unreliable, connectionless data transfer at the transport layer
- Protocol management

However, NetBIOS is a proprietary protocol no longer widely used. Therefore, it's probably no longer critical for you to understand name resolution based on NetBIOS. However, it's covered on the exam, and there are still some older applications out there that require NetBIOS at the Session layer of a network protocol stack.

Using the LMHOSTS file enables a network administrator to define all the names and their corresponding IP addresses in a file that they then can distribute to all NetBIOS-enabled computers. To do so, you must place the LMHOSTS file in the **C:\Windows\System32\Drivers\etc** folder. By using LMHOSTS, you can avoid using broadcast-based resolution methods, and you can also avoid having to use a name resolution server, such as WINS or DNS.

Note

LMHOSTS can only be used to resolve NetBIOS names, and is not useful for resolving hostnames.

WINS

You can use a WINS server to manage NetBIOS name registration, release, and resolution. The advantage of using WINS is that you can avoid using broadcast-based name resolution. In addition, you can also centralize management of NetBIOS names within your organization.

The key features of WINS are:

- Dynamic name resolution service for legacy clients
- Minimal administrator intervention
- Good integration with Dynamic Host Configuration Protocol (DHCP)
- Enables name registration, release, and resolution traffic to transit routers

WINS consists of two elements:

- WINS server. The WINS server is installed as a Windows Server 2019 operating system feature. The WINS server maintains a database that maps the NetBIOS computer names to IP addresses.
- WINS client. All computers running Windows 10 are WINS capable. You can configure a client computer to use one or more WINS servers by using the **WINS** tab on the **Advanced TCP/IP settings** page for the appropriate NIC. We'll refer to this process a bit later in this module.

HOSTS file

When organizations began deploying TCP/IP-based networks, they identified a need to resolve hostnames into IPv4 addresses. Because there were relatively few hosts on any given network, a simple text file was devised that would contain a list of all the computers on the IP network and their respective IP addresses.

Note

While the preceding methods for name resolution are designed for Windows 10 and Windows Server, both HOSTS and DNS were originally designed for the popular UNIX operating system. These days, all computer operating systems support using HOSTS or DNS .

When client computer needs to resolve a name, it examines (or *parses*) the HOSTS file and reads all entries into its name resolution cache. If the required entry is not available, the client must attempt to resolve the name using another method.

Note

Don't confuse HOSTS and LMHOSTS (discussed in more detail later). HOSTS is used in host name resolution and play a role in DNS name resolution. LMHOSTS is used for NetBIOS name resolution.

The advantage of using a HOSTS file is that you don't need to reply on broadcasts. In addition, because no actual query is sent onto the network there's no need to worry about routers. However, as your network changes and you add, remove, or rename computers, you must update the HOSTS files on all computers in your organization; this process can be time-consuming, and is also error-prone.

Note

In a Windows 10 computer, the Hosts file is stored in the **C:\Windows\System32\Drivers\etc** folder.

DNS

Remember that prior to DNS evolving, hostname resolution was managed by a HOSTS file, which was updated centrally and distributed to participating organizations. Providing the number of hosts and changes remained small, this was feasible. However, as the number of hosts grew, this solution became impractical. The new solution was DNS.

The DNS namespace is structured like an inverted tree starting at the root and working down. The namespace consists of domains and sub-domains which contain resource records. These records contain the information used to resolve queries. You're probably fairly familiar with these domain names: .com, .edu, .and gov, to name a few

DNS consists of two basic elements:

- Name servers. Any computer holding records for the DNS namespace is said to be a *name server*. (There are several different types of name servers, as we'll talk about shortly.) Name servers often contain the requested resource records, in which case they're said to be authoritative for that part of the namespace. If they're not authoritative for the zone, they will have pointers to other name servers, which might be authoritative. Authoritative information is organized into units called *zones*, which we'll be discussing in the next two lessons.

- Resolvers. These are software programs or apps running on client computers. For example, part of the Microsoft Edge browser is responsible for name resolution and is hence a resolver.

DNS maps to level 7 in the OSI model and uses either UDP (port 53) or TCP (port 53) as the underlying protocol. Resolvers first send UDP queries to servers for increased performance and resort to TCP only when needed. Servers use TCP when replicating DNS information.

Some of the advantages of using DNS are that:

- It's a de-facto standard.

- It avoids the need to use broadcasts.

- It enables DNS traffic to transit routers.

- It replaces the need for either HOSTS or LMHOSTS files.

- It's fully scalable from the smallest organization to the entire internet.

- It can be configured to avoid single points of failure, and is hence a resilient system

Generally, unless you have a specific application need, using DNS is the logical choice for almost all organizations.

Note

The name resolution method that you select is largely determined by the type of device that you have, and by the operating system installed on it. Windows 10, for example, supports all these different methods – although you can configure the name resolution method being used, you can also configure the preferred order in which these attempts are made.

In a TCP/IP-based network, the predominant—perhaps the only—method widely used is DNS. We'll examine this is more detail in the next lesson.

What method do you use?

Think about your computer network at home. What method of name resolution does it use? At your workplace, or school or college, what's being used for name resolution? Discuss this with the class.

Lesson 2: Resolving hostnames with DNS

DNS is a name resolution service that resolves names to IPv4 or IPv6 addresses. The DNS service is a hierarchical, distributed database; that is, it's structured logically, allowing many different servers to host the worldwide database of DNS names. Hostnames are resolved by using DNS.

Objectives

After you complete this lesson, you will be able to:

- Describe hostnames.
- Describe DNS.
- Explain how DNS name resolution works.
- Configure DNS settings in Windows 10.
- Describe DNS zones and domains.
- Identify different DNS zone types.
- Identify types of common resource records.

What are hostnames?

Hostnames are names assigned to computers running the TCP/IP protocol. A *fully qualified domain name* (*FQDN*) is the explicit DNS hostname that includes the computer's name and the subdomains through to the root domain. For example, as Figure 4 illustrates, if a computer has the name LON-SVR1 in the contoso.com domain, the FQDN for that computer is LON-SVR1.contoso.com.

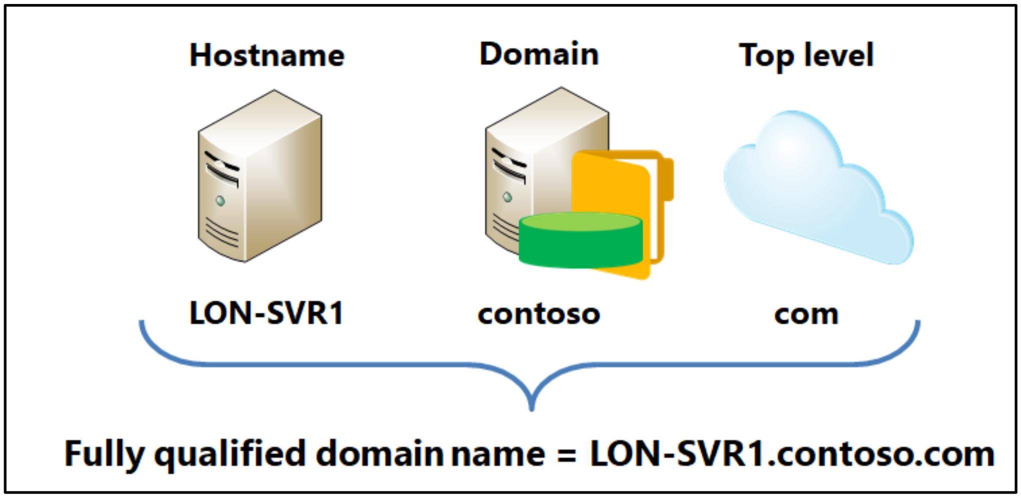


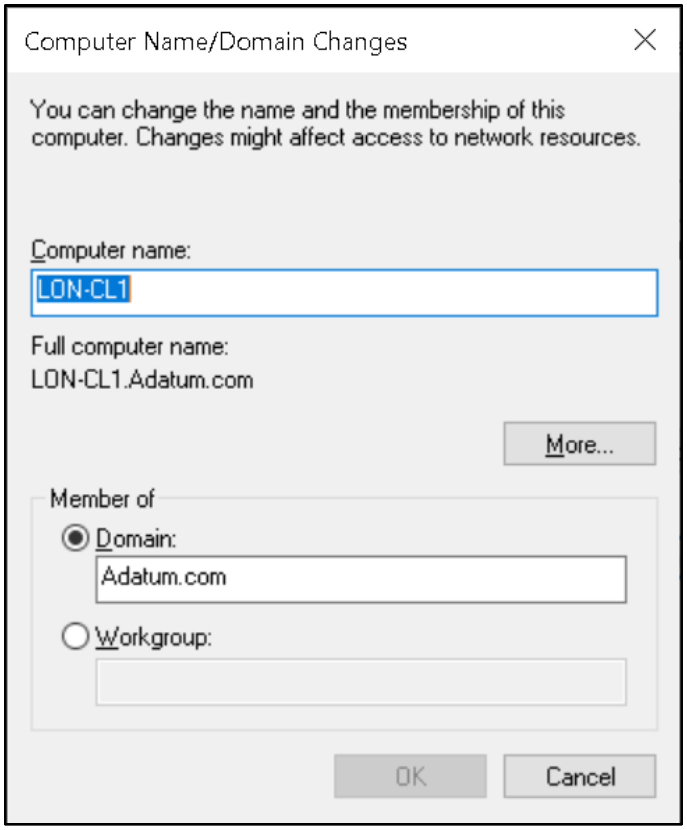
Figure 4. Hostname

DNS naming standards

The following characters are valid for DNS names:

- A through Z
- a through z
- 0 through 9
- Hyphen (-)

In Windows 10, when you assign a computer its computer name you also define its hostname. By default, the FQDN is derived from the Active Directory Domain Services (AD DS) domain name of which the Windows 10 computer is a member, as Figure 5 depicts.



Computer Name/Domain Changes

You can change the name and the membership of this computer. Changes might affect access to network resources.

Computer name: LON-CL1

Full computer name:
LON-CL1.Adatum.com

Member of

● Domain: Adatum.com

○ Workgroup:

Figure 5. Computer Name/Domain Changes dialog box

Creating host names

When you select host names, you should create host names that are intuitive and relatively easy to remember, yet still unique. The following lists some best practices to implement when creating host names:

- Select computer names that are easy for users to remember.
- Identify the owner of a computer in the computer name. For example, JOHN-DOE-01 indicates that John Doe uses the computer.
- Select names that describe the computer's purpose. For example, a file server named PAST-ACCOUNTS-01 indicates that the file server stores information related to past accounts.

- Do not use character case to convey the computer's owner or purpose. DNS is not case-sensitive.
- Match the AD DS domain name to the primary DNS suffix of the computer name.

Note

The DNS suffix is the part of the domain that comes after the computer name in the FQDN. In the example LON-SVR1.contoso.com, LON-SVR1's DNS suffix is contoso.com. The Primary DNS suffix should match the AD DS domain name to keep things simple.

- Use unique names for all computers in your organization. Don't assign the same computer name to different computers in different DNS domains.

What is DNS?

DNS is a worldwide service that allows you to enter a domain name (for example, Microsoft.com), which the computer resolves to an IP address. The benefit is that while IPv4 addresses might be long and difficult to remember (for example, 131.107.0.32), a domain name typically is easier to remember. In addition, you can use hostnames that don't change while the underlying IP addresses can be changed to suit your organizational needs.

Originally, there was one file on the internet that contained a list of all the domain names and their corresponding IP addresses. This list quickly became too long to manage and distribute. DNS was developed to solve the problems associated with using a single file. With the adoption of IPv6, DNS will become even more critical because IPv6 addresses (for example, 2001:db8:4136:e38c:384f:3764:b59c:3d97) are more complex than IPv4 addresses.

DNS consists of:

- DNS servers, which maintain DNS zones.
- DNS zones, which store resource records.
- Resource records, which identify specific computers, apps, or services.

- DNS domains, which separate hosts into a logical namespace.
- DNS resolvers, which are components in apps that issue name resolution queries.

How DNS name-resolution works

A *DNS resolver* is a client app that wants to resolve a name, typically using the process that Figure 6 illustrates:

1. The resolver checks to determine whether the required name is the local hostname.
2. The resolver checks the DNS resolver cache to determine whether the required name was already recently resolved. The resolver also checks the contents of the HOSTS file during this stage and caches any entries in the file.
3. Finally, the resolver attempts to query a DNS server.

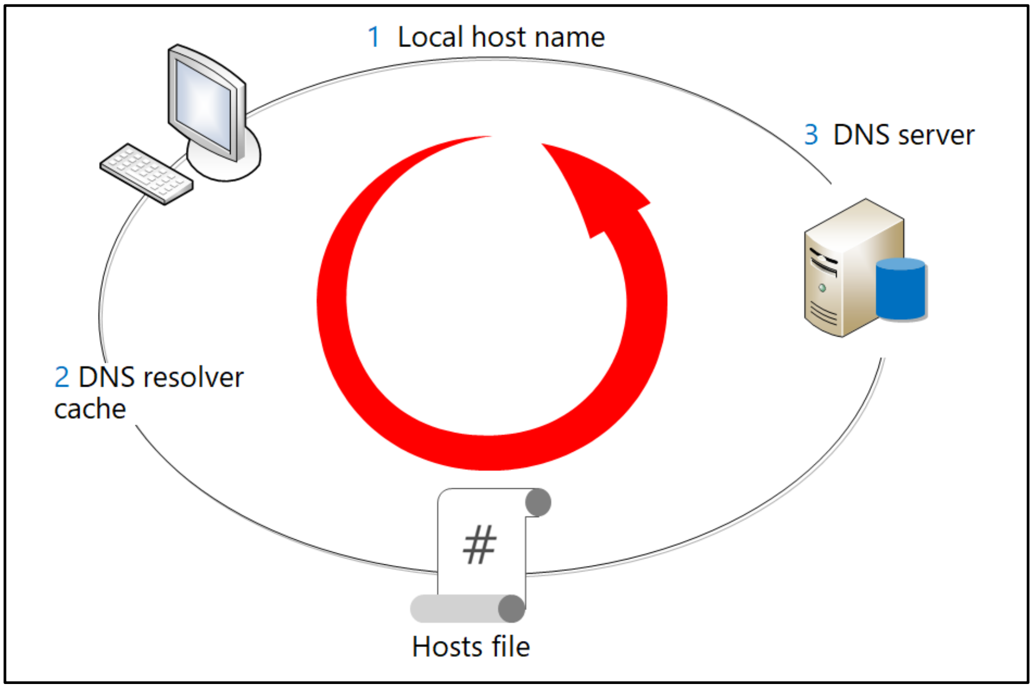


Figure 6. DNS name resolution

DNS uses two different types of queries:

- Recursive queries
- Iterative queries

Recursive queries

A recursive query can have two possible results:

- It returns the IP address of the host requested.
- The DNS server cannot resolve an IP address.

For security reasons, it's sometimes necessary to disable recursive queries on a DNS server. In doing so, the DNS server in question will not attempt to forward its DNS requests to another server. This can be useful when you don't want a particular DNS server communicating outside its local network.

Note

Typically, client computers use recursive queries when they petition a local DNS server for a record. Essentially, the client says, "Give me the answer I want, and if you don't know, don't refer me to another DNS server".

Iterative queries

Iterative queries provide a mechanism for accessing domain name information that resides across the DNS system, which enables servers to quickly and efficiently resolve names across many servers. This typically occurs across the internet.

Note

When a DNS server receives a request that it cannot answer using its local information or its cached lookups, it uses an iterative query to make the same request to another DNS server. Essentially, this time the DNS server says, "Give me the answer I want, and if you don't know, please refer me to another DNS server that might know."

When a DNS server receives an iterative query, it might answer with either the IP address for the domain name (if known), or with a referral to the DNS servers that are responsible for the domain being queried.

How internet queries work

A name resolution client query can take many paths, depending on whether it's public or private, and how the DNS infrastructure is designed. When DNS names are resolved on the internet, a whole system of computers is used instead of just a single server. There are 13 root servers on the internet that are responsible for managing the overall structure of DNS resolution.

Note

When you register a domain name on the internet, you're paying to be part of this system.

Using www.microsoft.com as an example, the name resolution process, as Figure 7 depicts, follows these high-level steps:

1. A workstation queries the local preferred DNS server for the IP address of www.microsoft.com using a recursive query.
2. If the local DNS server doesn't have the information, it queries a root DNS server in the organization for the location of the .com DNS servers using an iterative query.

3. The local DNS server queries a .com DNS server for the location of the Microsoft.com DNS servers.

4. The local DNS server then queries the Microsoft.com DNS server for the IP address of www.microsoft.com.

5. The Microsoft.com DNS server returns the result to the local DNS server, which caches the result.

6. The local DNS server returns the IP address of www.microsoft.com to the workstation, which caches the result.

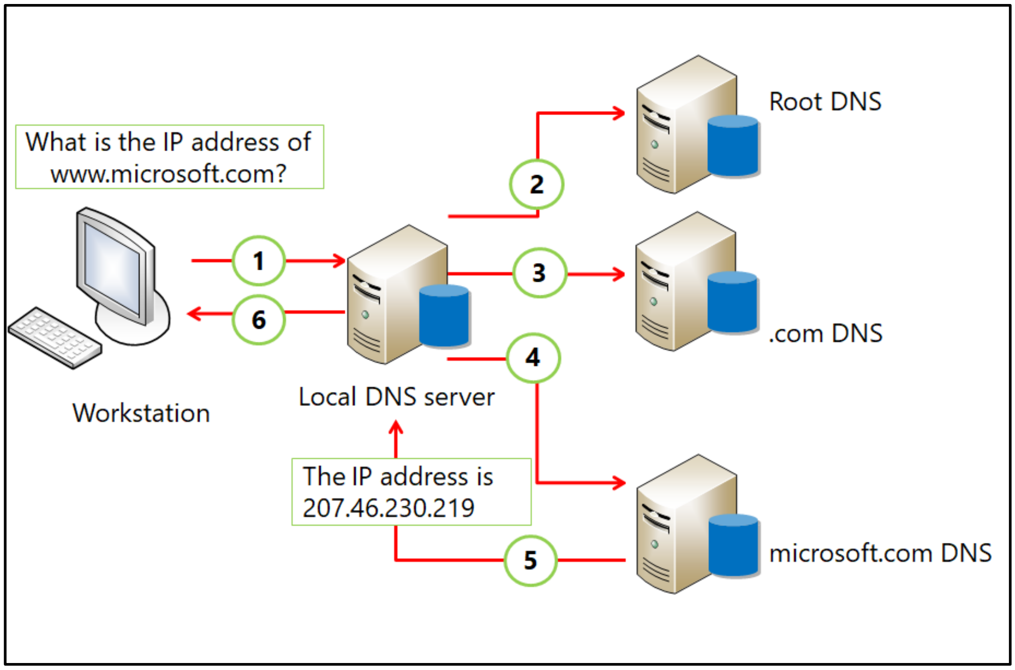

Root DNS
What is the IP address of www.microsoft.com?
2
1
3
6
.com DNS
4
Workstation
Local DNS server
The IP address is 207.46.230.219
5
microsoft.com DNS

Figure 7. Internet query using DNS

Modifying query behavior

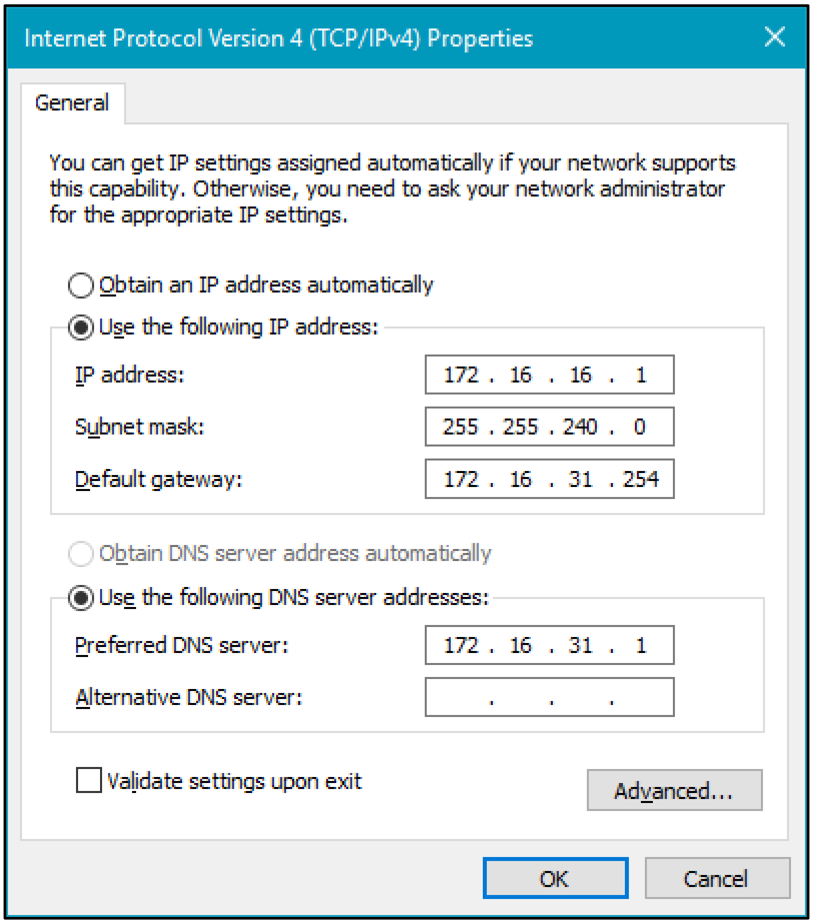
You can change the name resolution process in several ways, but two common options that you can use are:

- Caching. After a local DNS server resolves a DNS name, it will cache the results for approximately 24 hours. Later resolution requests for the DNS name are provided with the cached information.

- Forwarding. A DNS server can be configured to forward DNS requests to another DNS server, instead of querying root servers. For example, requests for all internet names can be forwarded to a DNS server at an internet service provider (ISP), which performs the rest of the resolving chain on behalf of the requesting DNS server and then returns the answer. This arrangement works well because the local DNS server doesn't have to be able to communicate with every DNS server on the internet.

Configuring DNS settings in Windows 10

You can configure DNS client settings on a computer running the Windows operating system by using the settings for each network adapter on the client, and for both IPv4 and IPv6, if enabled. You can specify the DNS server addresses on a per-adapter basis on the **Properties** page of the appropriate TCP/IPv4 or TCP/IPv6 protocol stack, as Figure 8 depicts:



Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address:	172 . 16 . 16 . 1
Subnet mask:	255 . 255 . 240 . 0
Default gateway:	172 . 16 . 31 . 254

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server:	172 . 16 . 31 . 1
Alternative DNS server:	. . .

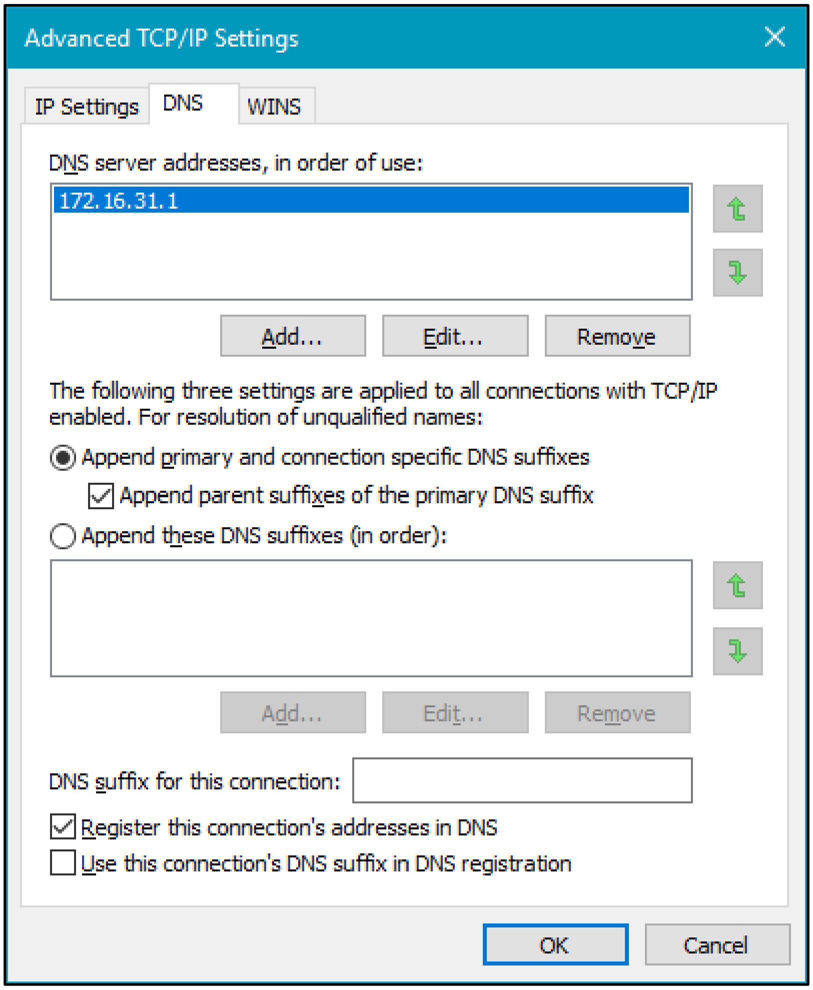
□ Validate settings upon exit

Figure 8. Internet Protocol Version 4 (TCP/IPv4) Properties dialog box, General tab

You can manually configure DNS client settings by performing the following steps:

1. Open **Network Connections**.
2. Right-click or access the context menu of the appropriate network adapter, and then select **Properties**.
3. In the **Properties** window, double-click or press **Spacebar +Enter** to open either **Internet Protocol Version 4 (TCP/IPv4)** or **Internet Protocol Version 6 (TCP/IPv6)**.
4. In the appropriate TCP protocol stack **Properties** window, select **Use the following DNS server addresses**. Then in the **Preferred DNS server** and **Alternate DNS server** text boxes, enter the IP address of the DNS servers.
5. Select **OK** twice, and then select **Close**.

You can also configure advanced properties, as Figure 9 illustrates.



DNS server addresses, in order of use:

172.16.31.1

The following three settings are applied to all connections with TCP/IP enabled. For resolution of unqualified names:

● Append primary and connection specific DNS suffixes

☑ Append parent suffixes of the primary DNS suffix

○ Append these DNS suffixes (in order):

DNS suffix for this connection:

☑ Register this connection's addresses in DNS

□ Use this connection's DNS suffix in DNS registration

Figure 9. Advanced TCP/IP Settings dialog box, DNS tab

To do this, perform the following steps:

1. Open **Network Connections**.
2. Right-click or access the context menu of the appropriate network adapter, and then select **Properties**.
3. In the **Properties** window, double-click or press **Spacebar +Enter** to open either **Internet Protocol Version 4 (TCP/IPv4)** or **Internet Protocol Version 6 (TCP/IPv6)**.
4. In the appropriate TCP protocol stack **Properties** window, select **Advanced**.

5. In the **Advanced TCP/IP Settings** window, select the **DNS** tab, and then configure the available properties. These advanced settings include several options for DNS suffix settings. The DNS suffix of a client specifies the domain namespace in which the client operates. You can also add additional DNS suffixes to enable the client to resolve single-label names for DNS names that exist in other DNS namespaces. Additionally, the advanced settings include the default behavior for the client to register its addresses in DNS, through the check box Register this connection's addresses in DNS.

6. Select **OK** twice, and then select **Close**.

Note

Although you can manually configure DNS server information for clients, this information is typically provided to client computers through a DHCP server.

You can also set DNS server addresses on client computers by using the following Windows PowerShell cmdlet:

```
Set-DnsClientServerAddress -InterfaceIndex 1 -ServerAddresses ("172.16.0.10","172.16.0.21")
```

The preceding command would set the DNS servers addresses 172.16.0.10 and 172.16.0.21 for the network adapter referred to by index 1, with 172.16.0.10 as the preferred server for the interface because it's listed first in the cmdlet. When you specify multiple potential DNS servers on a client, any DNS query issued from the client will follow a preferred order when selecting the server to query.

DNS zones and domains

DNS is made up of logical domains that are stored in databases (known as *zones*) on physical servers.

DNS zones

A DNS zone hosts all or a portion of a domain and its subdomains. In Figure 10, the Microsoft.com domain is separated into two zones: microsoft.com, and example.microsoft.com. The first zone hosts www.microsoft.com and ftp.microsoft.com. The example.microsoft.com domain is delegated to a new zone, which hosts the example.microsoft.com and its subdomains, ftp.example.microsoft.com and www.example.microsoft.com.

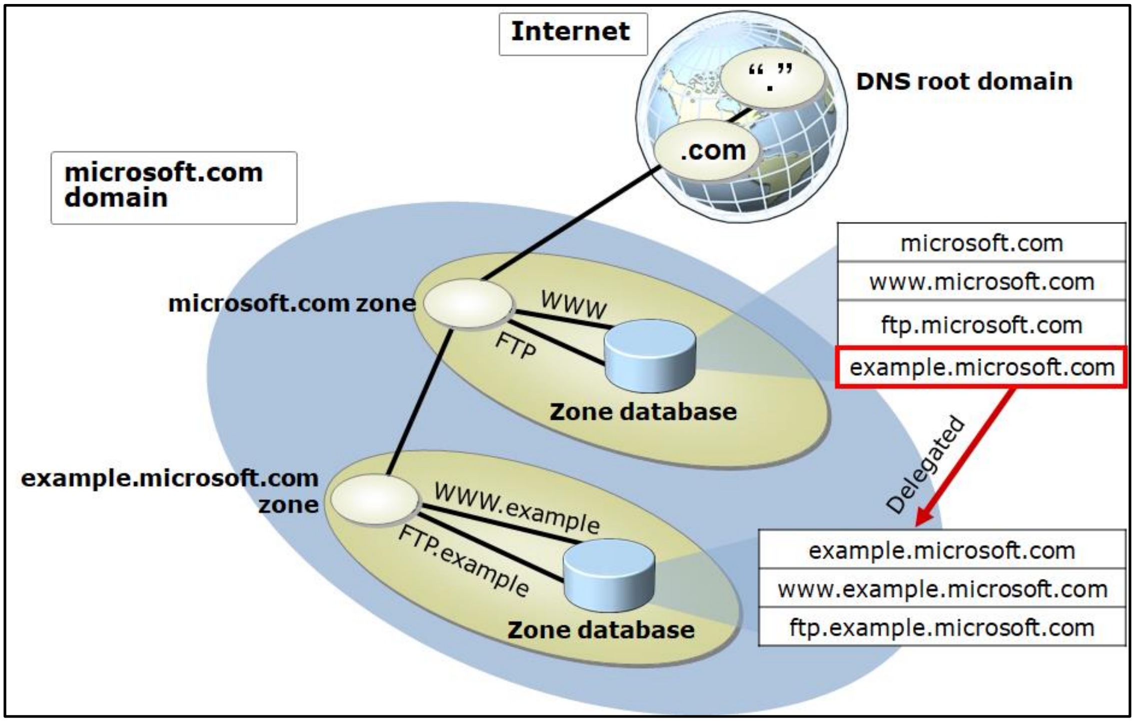


Figure 10. DNS zones and domains

Note

Important. The zone that hosts a root of the domain (Microsoft.com) must delegate the subdomain (example.microsoft.com) to the second zone. If this doesn’t occur, example.microsoft.com will be treated as if it were part of the first zone.

Zone data can be replicated to more than one server. This adds redundancy to a zone because the information needed to find resources in the zone now exists on two servers.

DNS domains

The naming structure used in DNS is called the *DNS namespace*. It's hierarchical, as Figure 11 depicts, which means that it starts with a root domain. That root domain can itself have any number of subdomains underneath it. In turn, each subdomain can have any number of subdomains under it.

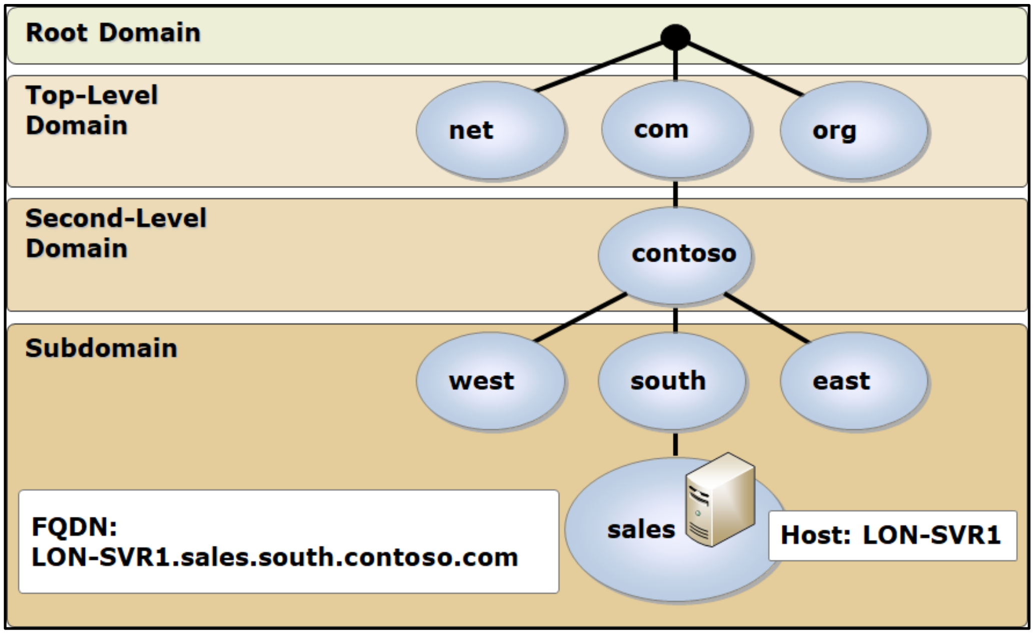


Figure 11. Example DNS namespace

The domain names themselves can either be public (internet facing) or private. If they're private, you can decide on your own how to define your namespace. If they're public, you must work with the Internet Corporation for Assigned Names and Numbers (ICANN) or other internet naming registration authorities that can delegate or sell unique names to you.

At the very root, DNS has a unique namespace, indicated by an empty string space " ". Preceding this is a single dot '.', and below this, in the public namespace is one of several other top-level domain namespaces.

There are three kinds of top-level domains in the public namespace:

- Organizational. This domain is based on the function of an organization; for example, .com, .net, .org, and .edu. Currently there are more than 20 variations, and these are distributed and managed by ICANN.
- Geographical. These are domains designated per country/region. There are more than 200 of these registered. Typically, each country/region has its own domain registration service. For example:
    - United Kingdom is .uk (co.uk is the .com equivalent for UK-based businesses)
    - Italy is .it.
    - Japan is .jp
- Reverse domains. These are special domains used in resolving addresses to names, known as a *reverse lookup*. These domains are in the *name.name* format, such as addr.arpa and ip6.arpa.

Typically, underneath these top-level domains are subdomains. For example, microsoft.com, university.edu, or government.gov. These subdomains can also have subdomains, such as unitedstates.microsoft.com, or physicsdept.university.edu.

Every computer and network node can be identified by its FQDN. For example, LON-SVR1.sales.south.contoso.com, as the Figure 11 depicts.

Zone types

There are two DNS zone types: primary and secondary.

Primary zone

When a zone that a DNS server hosts is a primary zone, the DNS server is the primary source for information about this zone and stores the master copy of zone data in a local file or in AD DS. When the DNS server stores the zone in a file, the primary zone file is named *zone_name*.dns by default and is located in the **%windir%\System32\Dns** folder on the server. When the zone is not stored in AD DS, this is the only DNS server that has a writable copy of the database.

Secondary zone

When a zone that a DNS server hosts is a secondary zone, the DNS server is a secondary source for the zone information. The zone at this server must be obtained from another remote DNS server that also hosts the zone. This DNS server must have network access to the remote DNS server to receive updated zone information. Because a secondary zone is a copy of a primary zone that another server hosts, it cannot be stored in AD DS. Secondary zones can be useful if you're replicating data from non-Windows DNS zones.

Forward and reverse zones

Zones can be either forward, or reverse (also known as *inverse*).

Forward lookup zone

The forward lookup zone resolves host names to IP addresses and hosts the common resource records: A, CNAMES, SRV, MX, SOA, and NS. (We'll discuss more about these resource records later in this lesson.)

Reverse lookup zone

The reverse lookup zone resolves an IP address to a domain name. It hosts SOA, NS, and PTR records. A reverse zone functions in the same manner as a forward zone, but the IP address is the part of the query, while the host name is the returned information. Reverse zones are not always configured, but you should configure them to reduce warning and error messages.

Many standard internet protocols rely on reverse zone lookup data to validate forward zone information. For example, if the forward lookup indicates that training.contoso.com is resolved to 192.168.2.45, you can use a reverse lookup to confirm that 192.168.2.45 is associated with training.contoso.com.

Having a reverse zone is important if you have applications that rely on looking up hosts by their IP addresses. Many applications will log this information in security or event logs. If you discover suspicious activity from a particular IP address, you can resolve the host using the reverse zone information. Many email security gateways use reverse lookups to validate that the IP address that is sending messages is associated with a domain.

Resource records

The DNS zone file stores resource records. Resource records specify a resource type and the IP address to locate the resource. The most common resource record is an A resource record. This is a simple record that resolves a hostname to an IP address. The host can be a workstation, server, or another network device, such as a router.

Resource records also help find resources for a particular domain. For example, when a Microsoft Exchange Server needs to find the server that is responsible for delivering mail for another domain, it will request the MX record for that domain. This record points to the A record of the host that is running the Simple Mail Transfer Protocol (SMTP) mail service.

Resource records can also contain custom attributes. For example, MX records have a preference attribute, which is useful if an organization has multiple mail servers. This will tell the sending server which mail server the receiving organization prefers. SRV records also contain information about which port the service is recognized, and the protocol that you should use to communicate with that service.

The following table describes the most common resource records.

Resource record	Explanation
SOA	Start of authority (SOA) resource record. Identifies the primary name server for a DNS zone.
A	Host address (A) resource record. The main record that resolves a host name to an IPv4 address.
CNAME	Canonical name (CNAME) resource record. An alias record type that maps one name to another. (For example, www.microsoft.com is a CNAME of the A record microsoft.com).
MX	Mail exchanger (MX) resource record. Used to specify an email server for a particular domain.
SRV	Service locator (SRV) resource record. Identifies a service that is available in the domain. AD DS uses these records extensively.
NS	Name Server (NS) resource record. Identifies all of the name servers in a domain.
AAAA	The main record that resolves a host name to an IPv6 address.
PTR	Pointer (PTR) resource record. Used to refer to and map an IP address to a domain name. The reverse lookup zone stores the names.

Have you registered a domain name?

Have you ever registered a domain name for yourself, a social organization to which you belong, or for a workplace? Think about how a computer user across the world somewhere tries to access a website in your registered domain. How does that work?

Lesson 3: Resolving NetBIOS names

Although NetBIOS is a legacy network session management protocol, some large organizations are still relying on apps that use NetBIOS. As such, it's possible that you might encounter NetBIOS within some networks. Therefore, it's important that you understand how best to implement NetBIOS name resolution.

Objectives

After you complete this lesson, you will be able to:

- Describe NetBIOS names.
- Explain how to resolve names with an LMHOSTS file.
- Implement NetBIOS name resolution in Windows Server.

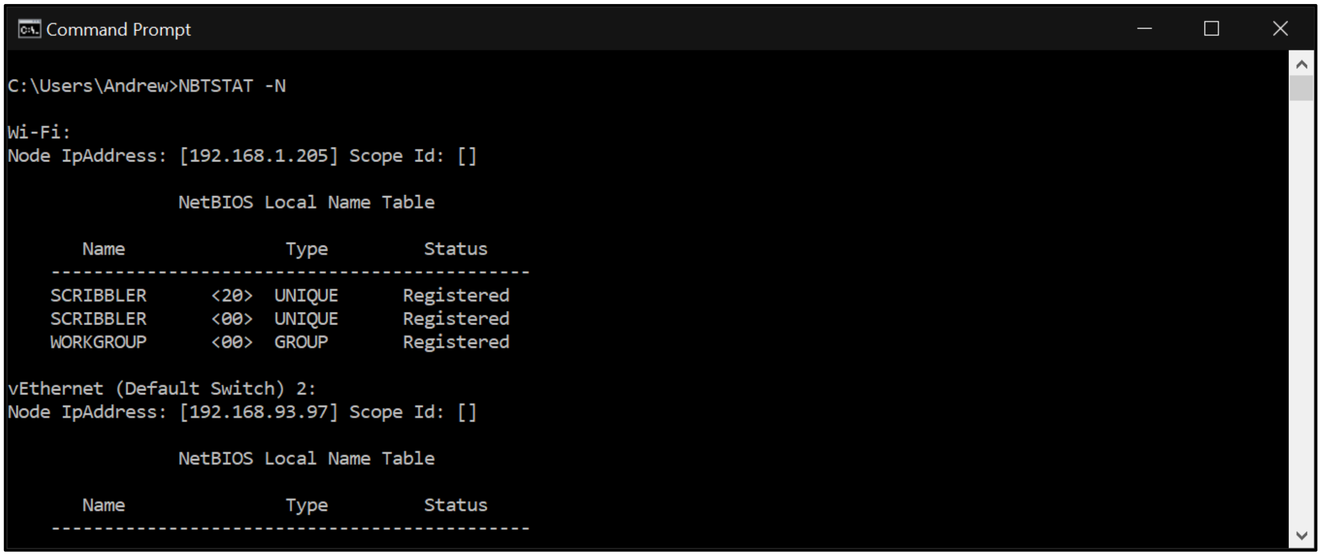
NetBIOS names

A *NetBIOS name* is a 16-byte name used to identify resources at a computer—15 characters for the system name, and a sixteenth for the service. A NetBIOS-enabled computer typically run several services. Windows 10 computers with the default networking components installed run the Workstation service (known as the *Client for Microsoft Networks*), the Server service (the File and Print Sharing service), and possibly others. Some names are unique (associated with one computer) while others are group names (associated with multiple computers or more accurately, multiple IP addresses).

Note

You can determine which names your computer is using by opening a Command Prompt window and running the **NBTSTAT -N** command.

Figure 12 displays the NetBIOS names registered by your computer:



```
C:\Users\Andrew>NBTSTAT -N

Wi-Fi:
Node IpAddress: [192.168.1.205] Scope Id: []

                NetBIOS Local Name Table

       Name               Type         Status
    ---------------------------------------------
    SCRIBBLER      <20>  UNIQUE      Registered
    SCRIBBLER      <00>  UNIQUE      Registered
    WORKGROUP      <00>  GROUP       Registered

vEthernet (Default Switch) 2:
Node IpAddress: [192.168.93.97] Scope Id: []

                NetBIOS Local Name Table

       Name               Type         Status
    ---------------------------------------------
```

Figure 12. NBTSTAT -N command results

The following table displays the most common NetBIOS services.

Registered name	Description
\\computername[00h]	Workstation service. This is a unique name, which means that only this computer registers the name.
\\computername[20h]	Server service. This is also a unique name.
\\domain_name[00h]	Domain name (or workgroup name if the device is not part of an AD DS domain. This is a group name, which means all devices that register this same group name can locate one another as being part of the same NetBIOS workgroup or domain

If a computer is configured to use NetBIOS and it needs to resolve a NetBIOS name into an IP address, then it can use the methods in the following table to resolve NetBIOS names.

Method	Description
Broadcast	Local broadcast
Cache	Examine NetBIOS name cache
LMHOSTS	Local lookup file
Name query server	WINS

The order in which these various methods are attempted is determined by the NetBIOS node type, as described in the following table.

Node type	Description
B-node (broadcast)	Uses UDP datagrams for registration and resolution. This is not ideal as it relies solely on broadcasting. As we know, broadcasts are restricted to the local subnet.
P-node (peer)	Uses a NetBIOS name server (WINS) to resolve names. Never uses broadcasts, therefore if a WINS Server cannot be contacted, resolution fails
M-node (mixed)	Uses b-node by default, and if unsuccessful, switches to p-node
H-node (hybrid)	Uses p-node, then resorts to b-node
Enhanced B-node (MS b-node)	Uses NetBIOS name cache, then broadcast and finally, LMHOSTS

Note

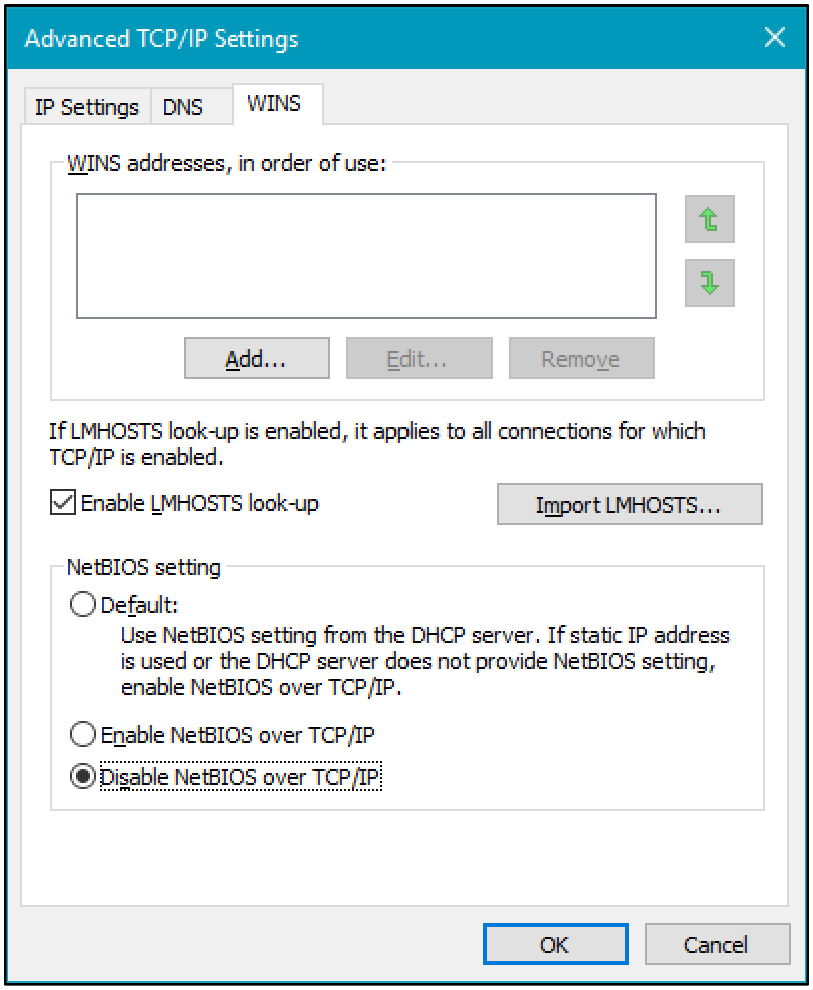
NetBIOS name resolution broadcasts use UDP over ports 137 and 138.

When troubleshooting NetBIOS over TCP/IP, you can use the **NBTSTAT** command-line tool. The following table summarizes the parameters of this useful tool.

Parameter	Description
NBTSTAT –n	Lists NetBIOS names registered by a computer.
NBTSTAT –c	Displays the NetBIOS name cache.
NBTSTAT – R	Reloads entries from the LMHOSTS file (entries with #PRE included).
NBTSTAT –r	Lists how NetBIOS names are being resolved.
NBTSTAT –a	Displays NetBIOS names of a target host; you specify the target by name.
NBTSTAT – A	Displays NetBIOS names of a target host; you specify the target by IP address.
NBTSTAT – RR	Sends a NetBIOS name refresh to a WINS server.

NetBIOS is rarely needed. If you want, you can disable NetBIOS on your Windows 10 computers by using the following procedure:

1. Open **Network Connections**.
2. Right-click or access the context menu of the appropriate NIC, and then select **Properties**.
3. Double-click or press **Spacebar +Enter** to open **Internet Protocol Version 4 (TCP/IPv4)**.
4. In the **Internet Protocol Version 4 (TCP/IPv4)** dialog box, select **Advanced**.
5. In the **Advanced TCP/IP Settings** dialog box, select the **WINS** tab.
6. To disable NetBIOS, select **Disable NetBIOS over TCP/IP**, as Figure 13 depicts, select **OK** twice, and then select **Close**.



WINS addresses, in order of use:

If LMHOSTS look-up is enabled, it applies to all connections for which TCP/IP is enabled.

☑ Enable LMHOSTS look-up

NetBIOS setting

○ Default:
Use NetBIOS setting from the DHCP server. If static IP address is used or the DHCP server does not provide NetBIOS setting, enable NetBIOS over TCP/IP.

○ Enable NetBIOS over TCP/IP

● Disable NetBIOS over TCP/IP

Figure 13. Disabling NetBIOS

Note

You can also disable NetBIOS over TCP/IP using the Registry editor, or by using Group Policy settings.

Notice that you can also disable LMHOSTS lookup and control the order in which WINS servers are contacted. (This is discussed more in the next two topics.)

Using LMHOSTS

Using LMHOSTS provides a means to avoid having to use broadcast-based name resolution. You use a text editor such as Notepad to create an LMHOSTS file, and then place the file in the **C:\Windows\System32\Drivers\Etc** folder.

The following table indicates the syntax of the LMHOSTS file.

Keyword	Description
#PRE	Defines which entries are preloaded permanently into the name cache.
#DOM:domain_name	Allows for domain activity such as logon over router, account database replication, and browsing an internet.
#INCLUDE	Loads NetBIOS name details from a centralized file, for example, from a Universal Naming Convention (UNC) name such as **\\server\netlogon\lmhosts**.
#BEGIN_ALTERNATE	Specifies a range of possible LMHOSTS file locations using UNC format names. The system uses the first located.
#END_ALTERNATE	The end of the alternates list.
#MH	Multiple addresses for a multihomed computer. (A *multihomed computer* is a Windows device with multiple NICs installed, each with a separate IP address.)

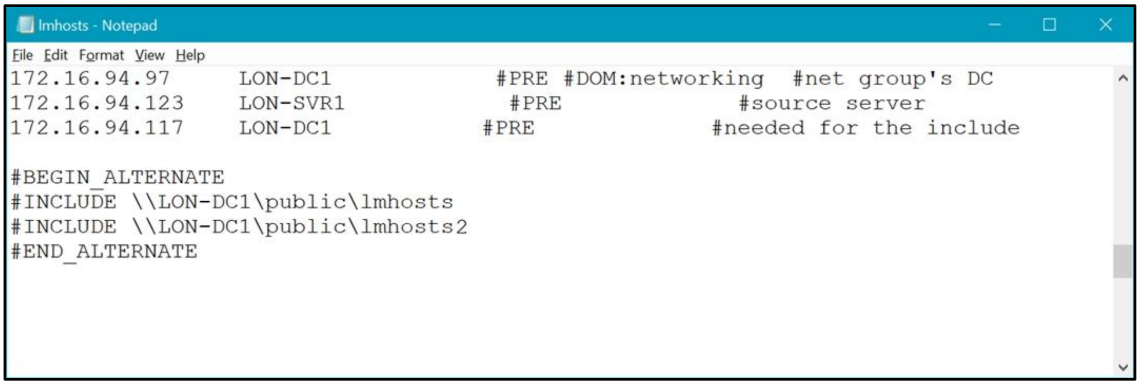
However, there are problems with using LMHOSTS files:

- Inaccuracies in spelling can cause severe problems.
- The file must be manually maintained.
- File changes might occur quite frequently and therefore present a significant management overhead.

There are solutions to these problems though:

- Verify each entry as it's created.
- Centralize the LMHOSTS file.
- Use WINS.

The following Figure 14 is of a sample LMHOSTS file in Notepad:



```
172.16.94.97     LON-DC1           #PRE #DOM:networking  #net group's DC
172.16.94.123    LON-SVR1           #PRE               #source server
172.16.94.117    LON-DC1          #PRE               #needed for the include

#BEGIN_ALTERNATE
#INCLUDE \\LON-DC1\public\lmhosts
#INCLUDE \\LON-DC1\public\lmhosts2
#END_ALTERNATE
```

Figure 14. LMHOSTS file

The first line of the host file identifies the IP address of a domain controller called LON-DC1. The #PRE tag loads the entry into NetBIOS name cache. The #DOM tag indicates that the entry is for a domain controller in the domain called networking. The next two lines identify server entries to be loaded into cache. Finally, the entries between the #BEGIN_ALTERNATE and #END_ALTERNATE tags identify the location of two LMHOSTS files stored on LON-DC1.

Implementing NetBIOS name resolution in Windows Server

To support NetBIOS name resolution properly, as Figure 15 illustrates, you must either deploy and configure WINS, or create and configure a GlobalNames zone in DNS:

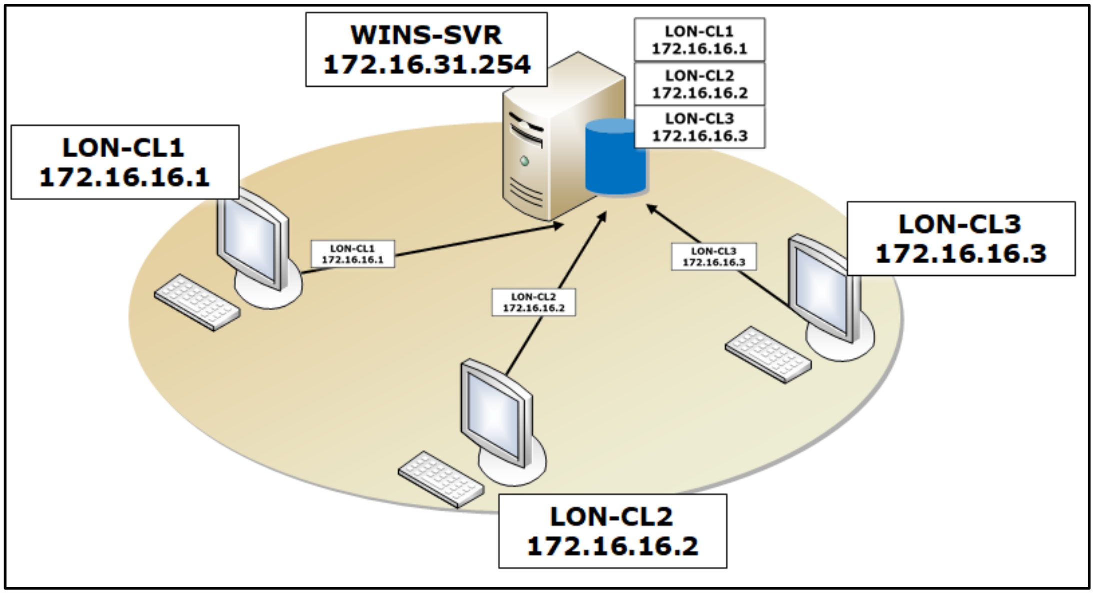


Figure 15. Name resolution diagram

Deploy and configure WINS

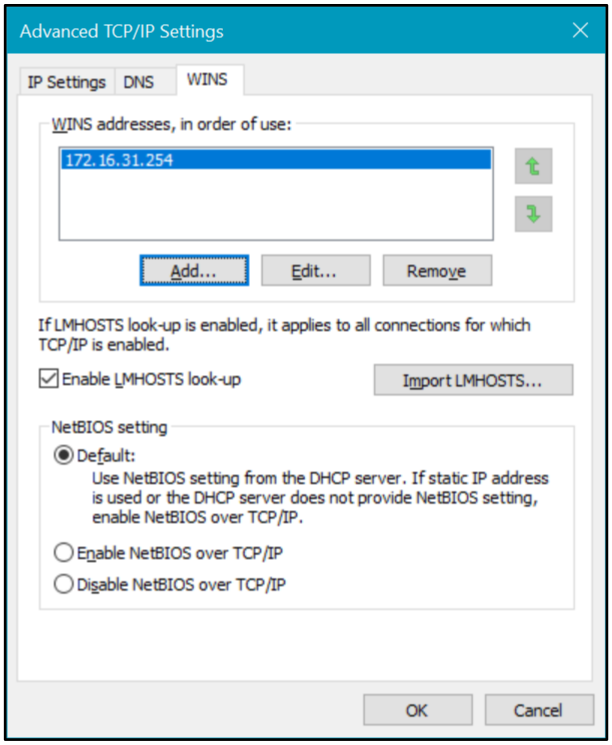
Configuring WINS is straightforward. On a computer running the Window Server operating system, install the WINS Server feature. This installs and configures the WINS database on that computer.

Note

It's a good idea to install at least two WINS servers. If one goes offline, then the other can continue to manage name registration, release, and resolution requests. If you deploy multiple WINS servers, they must be configured as replication partners so that their databases are synchronized. Configuring WINS replication is out of the scope of this course.

After you install one or more WINS servers, you must configure the networked computers to use WINS. This requires that you modify the network settings. To do this on a single computer, use the following procedure:

1. Open **Network Connections**.
2. Right-click or access the context menu of the appropriate NIC, and then select **Properties**.
3. Double-click or press **Spacebar +Enter** to open **Internet Protocol Version 4 (TCP/IPv4)**.
4. In the **Internet Protocol Version 4 (TCP/IPv4)** dialog box, select **Advanced**.
5. In the **Advanced TCP/IP Settings** dialog box, select the **WINS** tab, which Figure 16 depicts.
6. Select **Add**, and then enter the IP address of the WINS server(s) you want to use.
7. Select **OK** twice, and then select **Close**.



WINS addresses, in order of use:

172.16.31.254

If LMHOSTS look-up is enabled, it applies to all connections for which TCP/IP is enabled.

☑ Enable LMHOSTS look-up

NetBIOS setting

● Default:

Use NetBIOS setting from the DHCP server. If static IP address is used or the DHCP server does not provide NetBIOS setting, enable NetBIOS over TCP/IP.

○ Enable NetBIOS over TCP/IP

○ Disable NetBIOS over TCP/IP

Figure 16. Advanced TCP/IP Settings window, WINS tab

Clearly, going around all the computers in your organization and manually configuring the WINS Server address will be incredibly time-consuming. Instead, you can configure the required WINS setting as part of the automatic IPv4 configuration process. Specifically, you can configure the required WINS settings by using DHCP options. When a client obtains an IPv4 configuration from a DHCP server in your organization, they're also configured to use the appropriate WINS settings.

Note

One significant issue with WINS is that it supports only IPv4. You cannot use WINS to resolve IPv6 addresses.

When a client computer starts up, it registers the various NetBIOS names it's using, along with its IPv4 address. The WINS server verifies that no other computer is using the name (or names). Assuming no other computer is using the names, the client computer receives a Positive Name Registration Response.

Note

The only likely situation in which another computer would be using the same name is if the name is registered to another IPv4 address. For example, let's say a computer with the name LON-CL1 was previously using the IPv4 address of 172.16.16.1. But it started up in a different subnet, so it's now using the address 172.16.32.1., In this case, the WINS server will send a challenge to the registered user of the name, LON-CL1, which won't respond. The WINS server will release the name and register it for LON-CL1 using its new IP address.

When the client computer shuts down, it contacts the WINS server and releases its registered names. Theoretically, the names released could be registered by other computers. However, that's unlikely as all computers should have unique names anyway.

Client NetBIOS name-resolution process

When a client computer wants to resolve a name using WINS, it performs the following process:

1. The client computer checks to determine whether the petitioned name is the local computer name.

2. It then checks its cache of remote names. (Note that any NetBIOS name that is resolved is placed in a cache where it will remain for ten minutes.)

3. Next, the client petitions its configured WINS Server.

4. If unsuccessful, the client tries broadcasting for the required record.

5. Finally, the client examines the contents of the LMHOSTS file, (if configured to use the LMHOSTS file).

If the client is also configured to use DNS, it might try both the HOSTS file and petitioning a DNS server for the required record.

Create and configure a GlobalNames zone in DNS

Because NetBIOS and WINS are quite old technologies and not widely used, Microsoft provides another way to manage NetBIOS name resolution by using DNS. Instead of deploying WINS servers and configuring client computers to use WINS, a special DNS zone is created within the organization's DNS hierarchy. This zone contains what are referred to as *single-label names*, which in effect are NetBIOS names. This zone is called the *GlobalNames zone*, and it eliminates the need to use the NetBIOS-based WINS to provide support for single-label names.

Global names are based on CNAME resource records in a special forward lookup zone that uses single names to point to FQDNs. For example, GlobalNames zones would enable clients in both the adatum.com domain and the contoso.com domain to use a single label name (such as *data*), to locate a server whose FQDN is data.contoso.com without having to use the FQDN.

Learning in action: name-resolution

Scenario

You work at Lucerne Publishing in London. This growing publishing company is deploying the Windows 10 and Windows Server 2019 operating systems throughout its offices. In addition, staff have been issued iPhones and iPads that run line-of business apps and enable users to access corporate email. Until recently, most of the IT equipment was deployed on a single floor of a building in Kensington, London, and no real thought has been given to planning network infrastructure. You have been asked to assess the way in which name resolution should be deployed and configured to address the company's needs.

Questions

1. **Lucerne Publishing now has a dozen or more offices throughout the United Kingdom. They're connected by virtual private network (VPN) connections through the internet. You want to deploy a name service that supports all the different device types mentioned. What would you recommend they deploy?**

   A. WINS

   B. DNS

   C. LLMNR

2. **You discover that you must configure all the client computers with the IPv4 address of at least two DNS servers. How can you more easily do this for the Windows 10 computers?**

   A. Manually configure the Network Connections settings on each computer.

   B. Use Group Policy settings to deploy the configuration.

   C. Use DHCP to deploy the DNS settings.

3. **You discover that an app that your company uses in the printing process for physical books requires NetBIOS. How does this change your plans?**

   A. It makes no difference to the plan.

   B. Lucerne can create a GlobalNames zone to satisfy and name resolution requirements arising from the use of NetBIOS.

   C. Lucerne must use WINS as its primary name service.

4. **A branch office that occasionally gets disconnected from the network has only Windows 10 computers. You want to enable name resolution at this branch. What options do you have?**

   A. Use LLMNR because this is designed for Windows 10 locations where there is no DNS or WINS support.

   B. Deploy a DNS server at the location.

   C. Deploy a WINS server at the location.

Test your knowledge

1. **What type of name resolution might be useful for networks that are based on IPv6, where the computers are running Windows 10, and where there is no DNS or WINS?**

   A. Broadcast

   B. Hosts

   C. LLMNR

2. **Which file contains IPv4 addresses and computer names?**

   A. HOSTS

   B. LMHOSTS

   C. DNS zone file

3. **What container stores resource records in the DNS architecture?**

   A. Domains

   B. HOSTS file

   C. Zones

Fill in the blanks for the following sentences.

4. **When a Windows 10 computer shuts down, it (   ) any NetBIOS names it has registered with its configured WINS server.**

5. **The HOSTS file resides in the (   ) folder.**

6. **In (   ) DNS queries, the petitioned DNS server can refer the resolver to other DNS servers.**

7. **True or false: LLMNR uses broadcast-based traffic for name resolution.**

   True

   False

8. **True or false: DNS traffic can transit routers.**

   True

   False

Study the scenarios and answer the questions.

9. **Josh, the owner at Fourth Coffee, has asked your advice about name resolution. In his coffee shop he has a mix of device types: some Windows 10, some iPad devices, and a laptop running MacOS. In addition, clients using the coffee shops use Android, iPhone, and Windows Phone devices to browse the internet and check their email.**

   Josh wants to know how best to manage name resolution within his network. What would you advise?

10. **Contoso has both IPv4 and IPv6 in use throughout its locations across the world. When you visit the head office in New York City, you discover that a line-of-business app that cannot be upgraded or replaced requires NetBIOS. However, the number of devices running this app are relatively few.**

    How would you go about deploying name resolution services? What could you use to address the issue of NetBIOS? What name service could you use to address all of their needs?

Glossary

Term	Definition
A	An IPv4 host record
AAAA	An IPv6 host record
DNS zone	A database that contains resource records
CNAME	An alias resource record
DNS	The Domain Name System provides a hierarchical name resolution service. It's the basis of name resolution on the internet
Forward lookup	A type of DNS zone that is used to resolve names to IP addresses
GlobalNames zone	A specialized DNS zone that can be created on a DNS server running on Windows Server. It's designed to replace WINS functionality and provide NetBIOS name services for organizations that don't use WINS, but require NetBIOS name resolution.
HOSTS	A text file that contains IP address to hostname mappings
LLMNR	A multicast name registration, release, and resolution system designed for Windows Vista, and in use in Windows Server and Windows 10 today. Supports both IPv4 and IPv6.
LMHOSTS	A text file that contains IP address to computer name mappings in Windows computers
NetBIOS	A session-level protocol develop by IBM and used in early Microsoft networks such as MS-Net and LAN Manager. Some apps still use the session management capabilities of NetBIOS.

NetBT	NetBIOS over TCP/IP (NetBT) is the Microsoft implementation of a proprietary session management protocol over an industry standard network transport protocol stack (TCP/IP).
Resource Records	Records that exist in DNS zones, and that identify hosts and the services that they run
Reverse lookup	A type of DNS zone that is used to resolve IP addresses to names
WINS	The Windows Internet Name Services is a proprietary Microsoft name service designed to support NetBIOS apps and services running on computers installed with Windows operating systems.