IP Address Facts

IP addresses allow hosts to participate on IP based networks. An IP address:

- Is a 32-bit binary number represented as four octets (four 8-bit numbers). Each octet is separated by a period.
- IP addresses can be represented in one of two ways:
    - Decimal (for example 131.107.2.200). In decimal notation, each octet must be between 0 and 255.
    - Binary (for example 10000011.01101011.00000010.11001000). In binary notation, each octet is an 8-character number.
- To convert from binary to decimal, memorize the decimal equivalent to the following binary numbers:

10000000	01000000	00100000	00010000	00001000	00000100	00000010	00000001
128	64	32	16	8	4	2	1

- For each bit position with a 1 value, add the decimal values for that bit together. For example, the decimal equivalent of 10010101 is:
  128 + 16 + 4 + 1 = 149
- The IP address includes both the network and the host address.
- The subnet mask is a 32-bit number that is associated with each IP address that identifies the network portion of the address. In binary form, the subnet mask is always a series of 1's followed by a series of 0's (1's and 0's are never mixed in sequence in the mask). A simple mask might be 255.255.255.0.
- IP addresses have a default *class*. The address class identifies the range of IP addresses and a default subnet mask used for the range. The following table shows the default address class for each IP address range.

Class	Address Range	First Octet Range	Default Subnet Mask
A	1.0.0.0 to 126.255.255.255	1-126 (00000001--01111110 binary)	255.0.0.0
B	128.0.0.0 to 191.255.255.255	128-191 (10000000--10111111 binary)	255.255.0.0
C	192.0.0.0 to 223.255.255.255	192-223 (11000000--11011111 binary)	255.255.255.0
D	224.0.0.0 to 239.255.255.255	224-239 (11100000--11101111 binary)	n/a
E	240.0.0.0 to 255.255.255.255	240-255 (11110000--11111111 binary)	n/a

- When using the default subnet mask for an IP address, you have the following number of subnet addresses and hosts per subnet:
    - There are only 126 Class A network IDs (most of these addresses are already assigned). Each class A address gives you 16,777,214 hosts per network.
    - There are 16,384 Class B network IDs. Each class B address gives you 65,534 hosts per network.
    - There are 2,097,152 Class C network IDs. Each class C address gives you 254 hosts per network.
    - Class D addresses are used for multicast groups rather than network and host IDs.

- Class E addresses are reserved for experimental use.

As you are assigning IP addresses to hosts, be aware of the following special considerations:

Address	Consideration
Network	The first address in an address range is used to identify the network itself. For the network address, the host portion of the address contains all 0's. For example: • Class A network address: 115.0.0.0 • Class B network address: 154.90.0.0 • Class C network address: 221.65.244.0
Broadcast	The last address in the range is used as the broadcast address and is used to send messages to all hosts on the network. In binary form, the broadcast address has all 1's in the host portion of the address. For example, assuming the default subnet masks are used: • 115.255.255.255 is the broadcast address for network 115.0.0.0 • 154.90.255.255 is the broadcast address for network 154.90.0.0 • 221.65.244.255 is the broadcast address for network 221.65.244.0 **Note:** The broadcast address might also be designated by setting each of the network address bits to 0. For example, 0.0.255.255 is the broadcast address of a Class B address. This designation means "the broadcast address for this network."
Host Addresses	When you are assigning IP addresses to hosts, be aware of the following: • Each host must have a unique IP address. • Each host on the same network must have an IP address with a common network portion of the address. This means that you must use the same subnet mask when configuring addresses for hosts on the same network. The range of IP addresses available to be assigned to network hosts is identified by the subnet mask and/or the address class. When assigning IP addresses to hosts, be aware that you cannot use the first or last addresses in the range (these are reserved for the network and broadcast addresses respectively). For example: • For the class A network address 115.0.0.0, the host range is 115.0.0.1 to 115.255.255.254. • For the class B network address 154.90.0.0, the host range is 154.90.0.1 to 154.90.255.254. • For the class C network address 221.65.244.0, the host range is 221.65.244.1 to 221.65.244.254. **Note:** A special way to identify a host on a network is by setting the network portion of the address to all 0's. For example, the address 0.0.64.128 means "host 64.128 on this network."
Local Host	Addresses in the 127.0.0.0 range are reserved to refer to the local host (in other words "this" host or the host you're currently working at). The most commonly-used address is 127.0.0.1 which is the loopback address.

Because IP addresses assigned to hosts must be unique, the use of IP addresses on the Internet is controlled by organizations that ensure that no two organizations are given the same range of IP addresses to assign to hosts.

- The Internet Assigned Numbers Authority (IANA) manages the assignment of IP addresses on the Internet. IANA is operated by the Internet Corporation for Assigned Names and Numbers (ICANN).
- IANA allocates blocks of IP addresses to Regional Internet Registries (RIRs). An RIR has authority for IP addresses in a specific region of the world.
- An RIR assigns a block of addresses to Internet Service Providers (ISPs).
- An ISP assigns one or more IP addresses to individual computers or organizations connected to the Internet.

Subnetting Facts

Subnetting is the process of dividing a large network into smaller networks. When you subnet a network, each network segment (called a *subnet*) has a different network address (also called a *subnet address*). In practice, the terms *network* and *subnet* are used interchangeably to describe a physical network segment with a unique network address.

From a physical standpoint, subnetting is necessary because all network architectures have a limit on the number of hosts allowed on a single network segment. As your network grows, you will need to create subnets (physical networks) to:

- Increase the number of devices that can be added to the LAN (to overcome the architecture limits)
- Reduce the number of devices on a single subnet to reduce congestion and collisions
- Reduce the processing load placed on computers and routers
- Combine networks with different media types within the same internetwork (subnets can not be used to combine networks of different media type on to the same subnet)

Subnetting is also used to efficiently use the available IP addresses. For example, an organization with a class A network ID is allocated enough addresses for 16,777,214 hosts. If the organization actually uses only 10,000,000 host IDs, over 6 million IP addresses are not being used. Subnetting provides a way to break the single class A network ID into multiple network IDs.

- Subnetting uses *custom* rather than the default subnet masks. For example, instead of using 255.0.0.0 with a Class A address, you might use 255.255.0.0 instead.
- Using custom subnet masks is often called *classless* addressing because the subnet mask cannot be inferred simply from the class of a given IP address. The address class is ignored and the mask is always supplied to identify the network and host portions of the address.
- When you subnet a network by using a custom mask, you can divide the IP addresses between several subnets. However, you also reduce the number of hosts available on each network.

The following table shows how a Class B address can be subnetted to provide additional subnet addresses. Notice how by using a custom subnet mask the Class B address looks like a Class C address.

	Default Example	**Custom Example**

Network Address	188.50.0.0	188.50.0.0
Subnet Mask	255.255.0.0	255.255.255.0
# of Subnet Addresses	One	254
# of Hosts per Subnet	65,534	254 per subnet
Subnet Address(es)	188.50.0.0 (only one)	188.50.1.0 188.50.2.0 188.50.3.0 (and so on)
Host Address Range(s)	188.50.0.1 to 188.50.255.254	188.50.1.1 to 188.50.1.254 188.50.2.1 to 188.50.2.254 188.50.3.1 to 188.50.3.254 (and so on)

Note: It is possible to use subnet masks that do not use an entire octet. For example, the mask 255.255.252.0 uses six extra binary bits in the third octet. For the Network+ exam, you do not need to know how to work with such custom masks.

Be aware of the following additional facts about custom subnet masks:

- While subnetting divides a large address space into multiple subnets, *supernetting* combines multiple smaller network addresses into a single larger network. For example, this allows multiple Class C addresses to be combined into a single network.
- *Classful* addresses are IP addresses that use the default subnet mask. They are classful because the default subnet mask is used to identify the network and host portions of the address. *Classless* addresses are those that use a custom mask value to separate network and host portions of the IP address.
- Using classless addresses is made possible by a feature called Classless Inter-Domain Routing (CIDR). CIDR allows for non-default subnet masks (variable length subnet mask or VLSM). Routers use the following information to identify networks:
    - The beginning network address in the range
    - The number of bits used in the subnet mask

  For example, the subnet 199.70.0.0 with a mask of 255.255.0.0 is represented as 199.70.0.0/16 (with 16 being the number of 1 bits in the subnet mask).

Addressing Method Facts

The following table lists several options for assigning IP addresses.

Method	Uses
Dynamic Host Configuration Protocol (DHCP)	A DHCP server is a special server configured to pass out IP address and other IP configuration information to network clients. • When a client boots, it contacts the DHCP server for IP configuration information.

	• The DHCP server is configured with a range of IP addresses it can assign to hosts (Microsoft calls these ranges *scopes*).
	• The DHCP server can also be configured to pass out other IP configuration such as the default gateway and DNS server addresses.
	• The DHCP server ensures that each client has a unique IP address.
	• The DHCP server can be configured to not assign specific addresses in the range, or to assign a specific address to a specific host.
	• The DHCP server assigns the IP address and other information to the client. The assignment is called a *lease*, and includes a lease time that identifies how long the client can use the IP address.
	• Periodically and when the client reboots, it contacts the DHCP server to renew the lease on the IP address.
	• The DHCP lease process uses frame-level broadcasts. For this reason, DHCP requests typically do not pass through routers to other subnets. To enable DHCP across subnets:
	  ○ Enable BootP (DHCP broadcast) requests through the router.
	  ○ Configure a computer for BootP forwarding to request IP information on behalf of other clients.
	• You can configure a DHCP server to deliver the same address to a specific host each time it requests an address. Microsoft calls this configuration a *reservation*.
	• DHCP is a TCP/IP protocol. Any client configured to use DHCP can get an IP address from any server configured for DHCP, regardless of operating system.
	
	Use DHCP for small, medium, or large networks. DHCP requires a DHCP server and minimal configuration.

Automatic Private IP Addressing (APIPA)	APIPA is a Microsoft implementation of automatic IP address assignment without a DHCP server. Using APIPA, hosts assign themselves an IP address on the 169.254.0.0 network (mask of 255.255.0.0). With APIPA: • The host is configured to obtain IP information from a DHCP server (this is the default configuration). • If a DHCP server can't be contacted, the host uses APIPA to assign itself an IP address. • The host only configures the IP address and mask. It does not assign itself the default gateway and DNS server addresses. For this reason, APIPA can only be used on a single subnet. Use APIPA: • On small, single-subnet networks where you do not need to customize the IP address range. • As a fail safe for when a DHCP server is unavailable to provide limited communication capabilities.
Alternate IP configuration	With an alternate IP configuration, the system attempts to use DHCP for TCP/IP configuration information. If a DHCP server cannot be contacted, the static configuration values are used. When you configure an alternate IP address, APIPA is no longer used. Use an alternate configuration: • If you have a computer (such as a laptop) that connects to two networks: one

	with a DHCP server and another without a DHCP server. • If you want to provide values to properly configure the computer in case the DHCP server is unavailable.
Static (manual) assignment	Using static addressing, IP configuration information must be manually configured on each host. Use static addressing: • On networks with a very small number of hosts. • On networks that do not change often or that will not grow. • To permanently assign IP addresses to hosts that must have always have the same address (such as printers, servers, or routers). • For hosts that cannot accept an IP address from DHCP. • To reduce DHCP-related traffic. **Note:** Static addressing is very susceptible to configuration errors and duplicate IP address configuration errors (two hosts that have been assigned the same IP address). Static addressing also disables both APIPA and DHCP capabilities on the host.

Name Resolution

As you study this section, answer the following questions:

- How are host names organized in DNS?
- What is the difference between a forward lookup and a reverse lookup?
- What is the role of the root servers in DNS?
- What is the difference between a zone and a domain in DNS?
- What is the difference between an A record and a PTR record?

After finishing this section, you should be able to complete the following tasks:

- Configure DNS zones and records to identify individual hosts.
- Configure preferred and alternate DNS server addresses on a Windows host.

This section covers the following exam objectives:

- 1.1 Explain the function of common networking protocols
  - DNS
- 3.2 Identify the functions of specialized network devices
  - DNS server

DNS Facts

The Domain Name System (DNS) is a hierarchical, distributed database that maps logical host names to IP addresses. The DNS hierarchy is made up of the following components:

- . (dot) domain (also called the *root* domain)
- Top Level Domains (TLDs) such as .com, .edu, .gov
- Additional domains such as yahoo.com, microsoft.com, etc.
- Hosts

The fully-qualified domain name (FQDN) includes the host name and all domain names, separated by periods. The final period (for the root domain) is often omitted and implied.

DNS is a distributed database because no one server holds all of the DNS information. Instead, multiple servers hold portions of the data.

- Each division of the database is held in a *zone* database file.
- Zones typically contain one or more domains, although additional servers might hold information for child domains.
- DNS servers hold zone files and process name resolution requests from client systems.

When you use the host name of a computer (for example if you type a URL such as www.mydomain.com), your computer uses the following process to find the IP address.

1. The host looks in its local cache to see if it has recently resolved the host name.
2. If the information is not in the cache, it checks the Hosts file. The Hosts file is a static text file that contains hostname-to-IP address mappings.
3. If the IP address is not found, the host contacts its preferred DNS server. If the preferred DNS server can't be contacted, it continues contacting additional DNS servers until one responds.
4. The host sends the name information to the DNS server. The DNS server then checks its cache and Hosts file. If the information is not found, the DNS server checks any zone files that it holds for the requested name.
5. If the DNS server can't find the name in its zones, it forwards the request to a root zone name server. This server returns the IP address of a DNS server that has information for the corresponding top-level domain (such as .com).
6. The first DNS server then requests the information from the top-level domain server. This server returns the address of a DNS server with the information for the next highest domain. This process continues until a DNS server is contacted that holds the necessary information.
7. The DNS server places the information in its cache and returns the IP address to the client host. The client host also places the information in its cache and uses the IP address to contact the desired destination device.

You should know the following facts about DNS:

- A *forward* lookup finds the IP address for a given host name. A *reverse* lookup finds the host name from a given IP address.
- An *authoritative* server is a DNS server that has a full, complete copy of all the records for a particular domain.
- Zone files hold records that identify hosts.
    - A records map host names to IP addresses.
    - PTR (pointer) records map IP addresses to host names.
- *Recursion* is the process by which a DNS server or host uses root name servers and subsequent servers to perform name resolution. Most client computers do not perform recursion, rather they submit a DNS request to the DNS server and wait for a complete response. Many DNS servers will perform recursion.

- Some DNS servers might forward the name resolution request to another DNS server and wait for the final response rather than performing recursion.
- Root DNS servers hold information for the root zone (.). Root servers answer name resolution requests by supplying the address of the corresponding to top-level DNS server (servers authoritative for .com, .edu, and such domains).
- On very small networks, you could configure a HOSTS file with several entries to provide limited name resolution services. However, you would have to copy the HOSTS file to each client. The work involved in this solution is only suitable for temporary testing purposes or to override information that might be received from a DNS server

IPv6 Facts

The current IP addressing standard, version 4, will eventually run out of unique addresses, so a new system is being developed. It is named IP version 6 or IPv6. The IPv6 address is a 128-bit binary number. A sample IPv6 IP address looks like: 35BC:FA77:4898:DAFC:200C:FBBC:A007:8973. The following list describes the features of an IPv6 address:

- The address is made up of 32 hexadecimal numbers, organized into 8 quartets.
- The quartets are separated by colons.
- Each quartet is represented as a hexadecimal number between 0 and FFFF. Each quartet represents 16-bits of data (FFFF = 1111 1111 1111 1111).
- Leading zeros can be omitted in each section. For example, the quartet 0284 could also be represented by 284.
- Addresses with consecutive zeros can be expressed more concisely by substituting a double-colon for the group of zeros. For example:
    - FEC0:0:0:0:78CD:1283:F398:23AB
    - FEC0::78CD:1283:F398:23AB (concise form)
- If an address has more than one consecutive location where one or more quartets are all zeros, only one location can be abbreviated. For example, FEC2:0:0:0:78CA:0:0:23AB could be abbreviated as:
    - FEC2::78CA:0:0:23AB or
    - FEC2:0:0:0:78CA::23AB

    But *not* FEC2::78CA::23AB

- The 128-bit address contains two parts:

Component	Description
Prefix	The first 64-bits is known as the *prefix*. ○ The 64-bit prefix can be divided into various parts, with each part having a specific meaning. Parts in the prefix can identify the geographic region, the ISP, the network, and the subnet. ○ The *prefix length* identifies the number of bits in the relevant portion of the prefix. To indicate the prefix length, add a slash (/) followed by the prefix length number. Full quartets with trailing 0's in the prefix address can be omitted (for example 2001:0DB8:4898:DAFC::/64). ○ Because addresses are allocated based on physical location, the prefix

	generally identifies the location of the host. The 64-bit prefix is often referred to as the *global routing* prefix.
Interface ID	The last 64-bits is the *interface ID*. This is the unique address assigned to an interface. ○ Addresses are assigned to interfaces (network connections), not to the host. Technically, the interface ID is *not* a host address. ○ In most cases, individual interface IDs are not assigned by ISPs, but are rather generated automatically or managed by site administrators. ○ Interface IDs must be unique within a subnet, but can be the same if the interface is on different subnets. ○ On Ethernet networks, the interface ID can be automatically derived from the MAC address. Using the automatic host ID simplifies administration.

IPv6 adds the following features which are not included in IPv4:

Feature	Description
Auto-configuration	Because hardware IDs are used for node IDs, IPv6 nodes simply need to discover their network ID. This can be done by communicating with a router.
Built-in Quality of Service	Built-in support for bandwidth reservations which make guaranteed data transfer rates possible. (Quality of service features are available as add-ons within an IPv4 environment, but are not part of the native protocol.)
Built-in Security Features	IPv6 has built-in support for security protocols such as IPSec. (IPSec security features are available as add-ons within an IPv4 environment.)
Source Intelligent Routing	IPv6 nodes have the option to include addresses that determine part or all of the route a packet will take through the network.

Although not yet widely adopted, you can implement IPv6 if your systems support it. As implementation of IPv6 proceeds, there will be cases when compatibility with IPv4 is required. Three strategies are recommended by IETF for IPv6 to IPv4 compatibility configuration:

Strategy	Description
Dual Stack	With a *dual stack* configuration, both the IPv4 and IPv6 protocol stacks run concurrently on a host. IPv4 is used to communicate with IPv4 hosts, and IPv6 is used to communicate with IPv6 hosts. When implemented on hosts, intermediate routers and switches must also run both protocol stacks. Use a dual stack configuration to enable a host to communicate with both IPv4 and IPv6 hosts.
Tunneling	*Tunneling* wraps an IPv6 packet within an IPv4 packet, allowing IPv6 hosts or sites to communicate over the existing IPv4 infrastructure. With tunneling, a device encapsulates IPv6 packets in IPv4 packets for transmission across an IPv4 network, and then the packets are de-encapsulated to their original IPv6 packets by another device at the other end. Tunneling solutions include: • Intra-site Automatic Tunnel Addressing Protocol (ISATAP) for implementations within a site

	• 6-to-4 tunneling for implementations across sites • Teredo for tunneling between two hosts Use tunneling to allow an IPv6 host to communicate with another IPv6 host through an IPv4 network.
Network Address Translation-Protocol Translation (NAT-PT)	NAT-PT is a protocol that converts the IPv6 packet header into an IPv4 packet header, and vice versa. This method is different than tunneling because the packet headers are converted between the IPv4 and IPv6, whereas tunneling wraps the IPv6 packet into an IPv4 packet. Use NAT-PT to allow IPv4 hosts to communicate with IPv6 hosts.