40555A Networking Fundamentals

Module 1: Overview of networking

Contents

Learning objectives based on the Microsoft Technology Associate (MTA) exam objectives

#	Lesson title	Learning objectives	Exam objectives mapped
1	Introduction to network protocols	• Describe the Open Systems Interconnection (OSI) protocol model. • Explain the TCP/IP suite of protocols. • Identify and explain the functions of Transmission Control Protocol (TCP), User Datagram Protocol (UDP), Internet Protocol (IP), and related protocols. • Explain where devices, protocols, and applications fit within the TCP/IP suite. • Explain frames, packets, and datagrams. • Identify well-known TCP and UDP ports. • Explain how network connections are established between TCP/IP hosts.	3.1.1 OSI model 3.1.2 Transmission Control Protocol model 3.1.3 Examples of devices, protocols, applications, and which OSI/TCP layer they belong to 3.1.4 TCP and UDP 3.1.5 Well-known ports for most used purposes 3.1.6 Packets and frames 3.2.8 IPv4 Ports 3.2.9 IPv4 Packets 3.3.9 IPv6 Ports 3.3.10 IPv6 Packets

#	Lesson title	Learning objectives	Exam objectives mapped
2	What is the internet, an intranet, and an extranet?	• Describe the internet. • Explain intranets. • Explain extranets.	Not applicable
3	Firewalls and security zones	• Describe common network-security threats and mitigations. • Describe firewalls. • Explain how to create security zones by using firewalls.	1.1.2 Security Zones 1.1.3 Firewalls
4	Virtual private networks	• Explain the need for remote access. • List and describe remote access solutions. • Provide an overview of virtual private networks (VPNs). • Describe the available VPN tunnel types. • Describe the available VPN authentication methods.	1.1.1 Virtual Private Networks 3.5.5 Virtual Private Networks

Module overview

We live in an increasingly connected world, and effective communication is becoming more important. To maintain and improve connections, the basic building blocks of networked systems must be efficient and reliable.

In this module, you'll learn about the fundamentals that help enable smooth, reliable communication within the digital world. We'll examine the building blocks of these networks: network protocols, security zones, firewalls, remote access, and VPNs. You'll also learn how to combine these building blocks to create intranets and extranets, and you'll learn how to join devices to the internet.

Objectives

After completing this module, you will be able to:

- Describe networking protocols.
- Define the internet, an intranet, and an extranet.
- Explain how firewalls work and define security zones.
- Understand how to implement VPNs.

Lesson 1: Introduction to network protocols

Any enjoyable conversation must have rules: we take turns communicating, we avoid using offensive language, and we try not to talk with full mouths.

Similarly, *network protocols* are formal standards and policies. Network protocols define rules, procedures, and formats for enabling communications between devices on a network. They’re also responsible for making sure that data is processed and transmitted in a prompt and secure manner.

The OSI reference model enables you to define how network devices communicate. Because the model is based on an open standard rather than a proprietary operating system or network protocol, you can more easily identify how data moves between apps on various networked devices regardless of the operating systems involved.

The OSI reference model includes seven layers, each of which maps to a corresponding software or hardware component (such as devices, protocols, standards, and apps) in a real-world networking system. You can use the OSI model to help design, maintain, and where necessary, troubleshoot a network infrastructure. This lesson explores the OSI reference model and identifies the real-world networking devices, protocols, and apps that correspond to each of these seven layers.

Objectives

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

- Describe the OSI protocol model.
- Explain the TCP/IP suite of protocols.
- Identify TCP, UDP, IP, and related protocols and explain their functions.
- Explain where devices, protocols, and applications fit within the TCP/IP suite.

- Explain frames, packets, and datagrams.
- Identify well-known TCP and UDP ports.
- Explain how network connections are established between TCP/IP hosts.

The OSI model

The International Organization for Standardization (ISO) is an international standard-setting body composed of representatives from various national standards organizations. ISO promotes worldwide proprietary, industrial, and commercial standards, and it was responsible for defining the OSI reference model. The purpose of this model is to help:

- Explain how devices on local area networks (LANs) or wide area networks (WANs) communicate.
- Enable comparison between different network operating systems with their disparate network protocol stacks.
- Standardize network protocol stacks.

The OSI model was devised when various networking protocols were in use throughout organizations in the world. Some of these protocols were proprietary, and because proprietary protocols work only with similar protocol products, interconnecting such devices can become problematic. Imagine if your television used a completely different set of connectors and cables than your surround sound system. You’d have to buy a second sound system made by your television manufacturer.

In contrast, TCP/IP was a popular non-proprietary suite of network protocols that gained widespread use in the 1980s. Microsoft added support for TCP/IP to its operating systems in the late 1980s. When computer systems use common protocols, interconnecting them becomes easier. This means you can choose products from a range of vendors knowing that they will work together. In turn, this increases competition, drives down prices, drives up innovation, and results in open standards that make networking easier.

Today, we only need to concern ourselves with TCP/IP. Having said that, we could make a comparison between TCP/IP and other new and emerging network protocols by using the OSI model. Let’s examine the seven layers of the OSI model.

The seven layers of the model explained

A network protocol such as TCP/IP is often referred to as a *protocol stack*. The OSI model enables you to compare dissimilar protocol stacks so that you can compare them functionally. For two devices—sometimes referred to as *systems* or *hosts*—to communicate with each other, they must be connected in a way that enables electrical signals sent from one to reach the other, as Figure 1 depicts. These electrical signals facilitate data transfer.

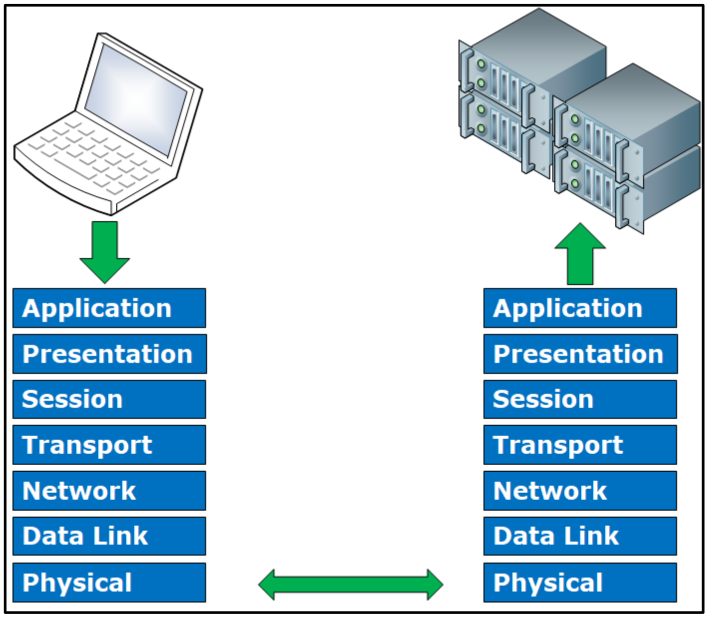


Figure 1. The seven layers of the OSI model

In computer networking, this typically necessitates that devices (hosts) have a network adapter. Depending on your infrastructure, these network adapters might support wired or wireless connections.

The seven layers of the OSI model are:

- Layer 7. This is the application layer where the journey begins. When apps create messages, this layer packages the messages into an appropriate structure for the networking protocols in use and passes them down the protocol stack to the presentation layer. Examples of apps at this layer include web and email servers, web browsers, and email clients.

- Layer 6. The presentation layer is responsible for formatting and converting data into a suitable format to forward down the stack. This might include data encryption, authentication, compression, and conversion.

- Layer 5. The session layer is responsible for establishing, maintaining, and when necessary, releasing secure channels between hosts. It's also responsible for host naming.

- Layer 4. This is the transport layer. Protocols at this layer are responsible for fragmenting data from higher layers into manageable pieces (or *datagrams*) for submission to lower layer protocols. At the receiving end, the manageable pieces must be reassembled in the correct order and missing pieces must be identified.

  To ensure that data passes up to the appropriate app, network ports are used to identify the apps that are running on a specific host. Most network protocol stacks implement both connection-oriented and connectionless-oriented protocols at this layer:

  - Connection-oriented protocols manage the reliable transfer of data between hosts. This layer identifies any data loss, and missing data is resent.

  - Connectionless-oriented protocols make best-effort deliveries, which tend to be faster and use less network bandwidth. However, where data isn't delivered, higher-level apps must recognize the loss of data and resubmit it.

- Layer 3. The network layer is responsible for delivering packets of information between hosts that might or might not be in different subnets. The word *packet* is simply the name of the container that holds information while it transits around a network. Routers and layer 3 switches, which we discuss later in this module, operate at this layer.

  Devices attached to a network that implements a network layer are assigned a logical network address—a unique address that ensures other devices can communicate with the device. Not all network protocol stacks implement a network layer. Because devices are equipped with a network adapter that has a physical address, this layer is also responsible for mapping the logical network layer address to the physical address.

- Layer 2. This is the data-link layer. Network adapters, bridges, and layer 2 switches (all of which we'll discuss later in this chapter) operate at this layer and distribute frames between connected devices. A *frame* is the name given to the data structure at layer 2. Layer 2 is responsible for error checking and error correction in as much as it ensures that transmissions between devices are without errors. Each device at this layer has a media access control (MAC) address—typically a 48-bit serial number provided by the network adapter vendor. The way in which devices communicate over physical media is defined in the network's topology.

- Layer 1. This is the physical layer. It handles bits of information that are merged onto the medium to which the network adapter is connected. This layer governs the way in which data is transmitted and received; for example, analog versus digital, baseband versus broadband, or wired versus wireless.

By way of an example, imagine that you're browsing the internet using a browser. In the address bar, you enter the URL of a web server to which you want to connect, and then you press Enter. The app on your computer (in this case, a browser) passes the request to access the remote server down the OSI model. Each layer performs its established task until finally, at the physical layer, electrical signals are sent to the remote server computer.

At the remote server computer, the signals are received at the physical layer and pass up the OSI model until the message from your computer reaches the web service app that's running at the upper layer. This process, with some variation depending on precisely what communication is taking place, occurs each time two devices communicate. In some instances, not all layers are used. For example, when routing between hosts, only layers 1 through 3 are used.

The TCP/IP suite of protocols

When the OSI model was devised, there were many networking protocols. Today, the most common is the TCP/IP suite of protocols. TCP/IP implements networking protocols across a four-layer architecture that broadly maps to the OSI model's seven layers, as Figure 2 depicts:

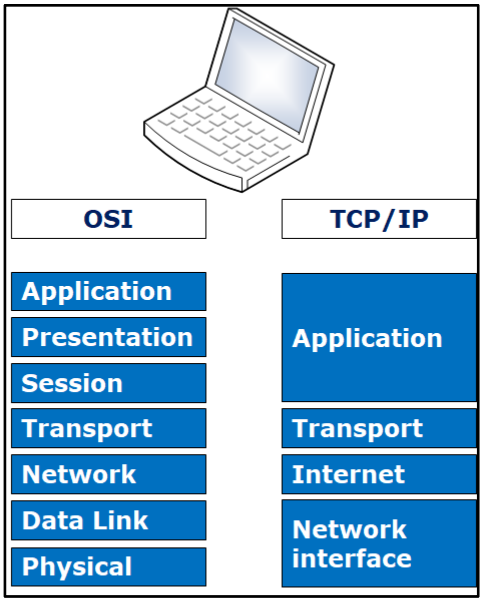


Figure 2. The TCP/IP suite of protocols

The four main layers of the TCP/IP protocol suite

The most common of these protocols are TCP, UDP, IP, Address Resolution Protocol (ARP), and Internet Control Message Protocol (ICMP). The following sections introduce these protocols and explain how they relate to networking.

Application layer

The top level of the architecture is the application layer. Microsoft provides a number of programming interfaces at this level, including:

- Winsock. Windows sockets (Winsock) provides a standard set of application programming interfaces (APIs) to several transport protocols (IPv4 and IPv6). These APIs enable networked apps to communicate through defined pathways.

- NetBIOS. Previously a widely implemented programming interface, network basic input/output system (NetBIOS) provides standard naming and messaging services. However, it’s less commonly used now.

These programmatic interfaces allow hosts to identify the correct program or service at the remote host with which they wish to communicate. Winsock is based on a port number, while NetBIOS uses a process identifier as part of the computer name.

Transport layer

Transport protocols provide communication sessions between computers. The two transport protocols are:

- TCP. This protocol provides connection-oriented delivery. This means that delivery is reliable and that packets are delivered in the correct sequence. Having packets arrive in the correct sequence is critical. For example, consider streaming a song from a website. If the song were broken into small pieces (packets) and sent to your phone, but there are missing packets or the packets arrive in the wrong order. The song won't sound as good and might not even be recognizable.

- UDP. This protocol provides connectionless delivery. There's no guarantee that packets will arrive in the correct sequence, and reliability is a function of higher-level protocols. Some apps can cope with packets going astray or arriving out of sequence. The receiving app requests the out-of-sequence or missing packets, and the sending app retransmits them.

Internet layer

Internet protocols take the data from the transport layer, known as *datagrams*, and inserts it into a suitable container (called *packets*) for routing onto the network. This process is known as *encapsulation*. The four protocols used at this layer are:

- ARP. For network communication to work, each computer must know the physical address of the corresponding computer with which it wishes to communicate. Each computer has a unique hardware address (MAC address). ARP is used for determining this unique hardware address.

- ICMP. ICMP sends messages and reports on packet delivery errors.

- IGMP. Internet Group Management Protocol (IGMP) is used for multicast routing. *Multicasting* is the process of sending data to multiple systems by using a group address. However, routers must be configured to pass this kind of traffic.

- IP. IP is responsible for addressing and routing packets between hosts and networks.

Network interface layer

This layer is responsible for putting frames onto the physical network. Remember that a *frame* is simply the name given to the container for the information that's handled at this layer. In Windows operating systems, this is handled by the *network device interface specification* (*NDIS*). NDIS is a driver specification that provides media access to higher-level protocols, regardless of how many network adapters are installed on the system or if they're from different hardware vendors.

Note

Choosing a transport protocol is a bit like choosing how a package is delivered. In fact, it's not even a choice you can make. You take your package to a mail service provider and hand it over the counter. The provider then uses several factors to decide how to ship your package, such as by truck, train, or even plane.

TCP, UDP, IP, and related protocols

Let's examine the TCP, UDP, IP and related protocols in more detail, as Figure 3 depicts.

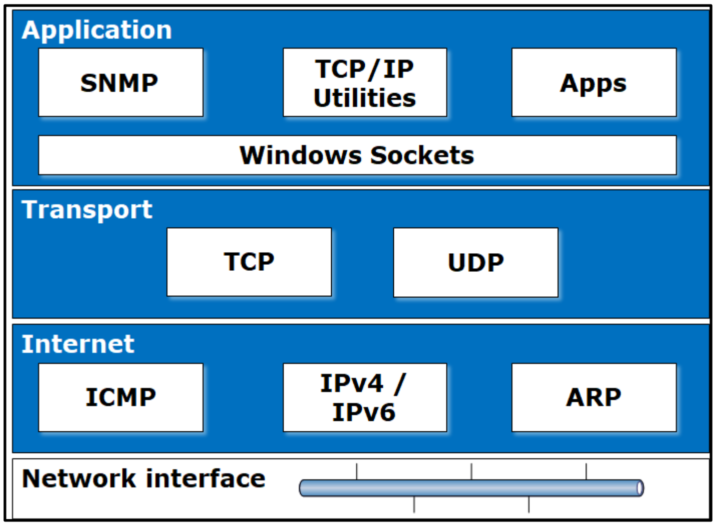


Figure 3. TCP, UDP, IP, and related protocols

TCP

The most common higher-level protocol in the suite is TCP. It provides a reliable connection-oriented packet delivery service built on or encapsulated within IP.

TCP guarantees the delivery of packets, ensures proper sequencing of the data, and provides a so-called checksum feature that validates both the packet header and its data for accuracy. If the network either corrupts or loses a TCP packet during transmission, TCP is responsible for retransmitting the faulty packet. This reliability makes TCP the protocol of choice for session-based data transmission, client-server applications, and critical services such as email.

Because TCP headers require additional bits to provide proper information sequencing and a checksum to ensure reliability of both the TCP packet header and the packet, TCP is slower than UDP. To guarantee successful data delivery, the protocol also requires that the recipient acknowledge successful receipt of data. Performance can be enhanced by using a sliding window, in which a pre-agreed number of packets are sent simultaneously. Acknowledgements are required for these groups of packets.

UDP

UDP provides faster communications, but at the price of reliability. UDP offers best effort, connectionless, and unreliable delivery that relies on higher-level protocols to provide reliability. Applications that don’t require an acknowledgement of data receipt use UDP.

Note

Because of the importance of reliability, websites (HTTP and HTTPS) use TCP datagrams. Conversely, network services such as Dynamic Host Configuration Protocol (DHCP) use UDP datagrams because reliability isn’t as critical, and the services can easily resend if necessary.

IP

IP provides packet delivery for all higher-level protocols in the TCP/IP suite. Primarily responsible for routing, IP provides best-effort delivery of an unreliable and connectionless nature. This means that delivery is not guaranteed, and a packet might be lost, delivered out of sequence, duplicated, or delayed. The two versions of IP are Internet Protocol version 4 (IPv4) and Internet Protocol version 6 (IPv6), which are discussed later in this course.

Note

The IP layer determines which route to take. Remember your package? Assuming it was sent on a truck, the driver has some choice over which route to take when delivering the package to its destination. The destination stays the same, but not the route. Do they take the interstate, or do they stick to rural roads to avoid rush hour? IP must make similar decisions. It knows where it's going with a packet, but it can choose between a variety of routes.

ARP

ARP is one of the maintenance protocols that supports the TCP/IP suite. If two systems are to communicate across a TCP/IP network, the system that sends the packet must map the final destination's IP address with its physical address.

Note

Let's use an example of making a phone call. When you select a contact in your mobile phone's address book, your phone dials a corresponding number that's associated with a unique subscriber ID. This subscriber ID maps to a telephone with that number. The mapping and the subscriber ID are stored on a data card in your contact's phone.

Several layers of mapping are going on here. You have a contact with a stored phone number, which you can change without changing the contact. Your contact can also change their phone without updating their phone number; the contact's phone number can move between telephone providers. Some means of resolving all this exists at various layers of the mobile telephone network. Similarly, an IP-based network uses names to map to logical IP addresses to make to physical device addresses.

An IP-based system will broadcast an ARP/Reverse Address Resolution Protocol (RARP) packet to the local network to determine the target host's MAC address. The response caches in the ARP cache for several minutes.

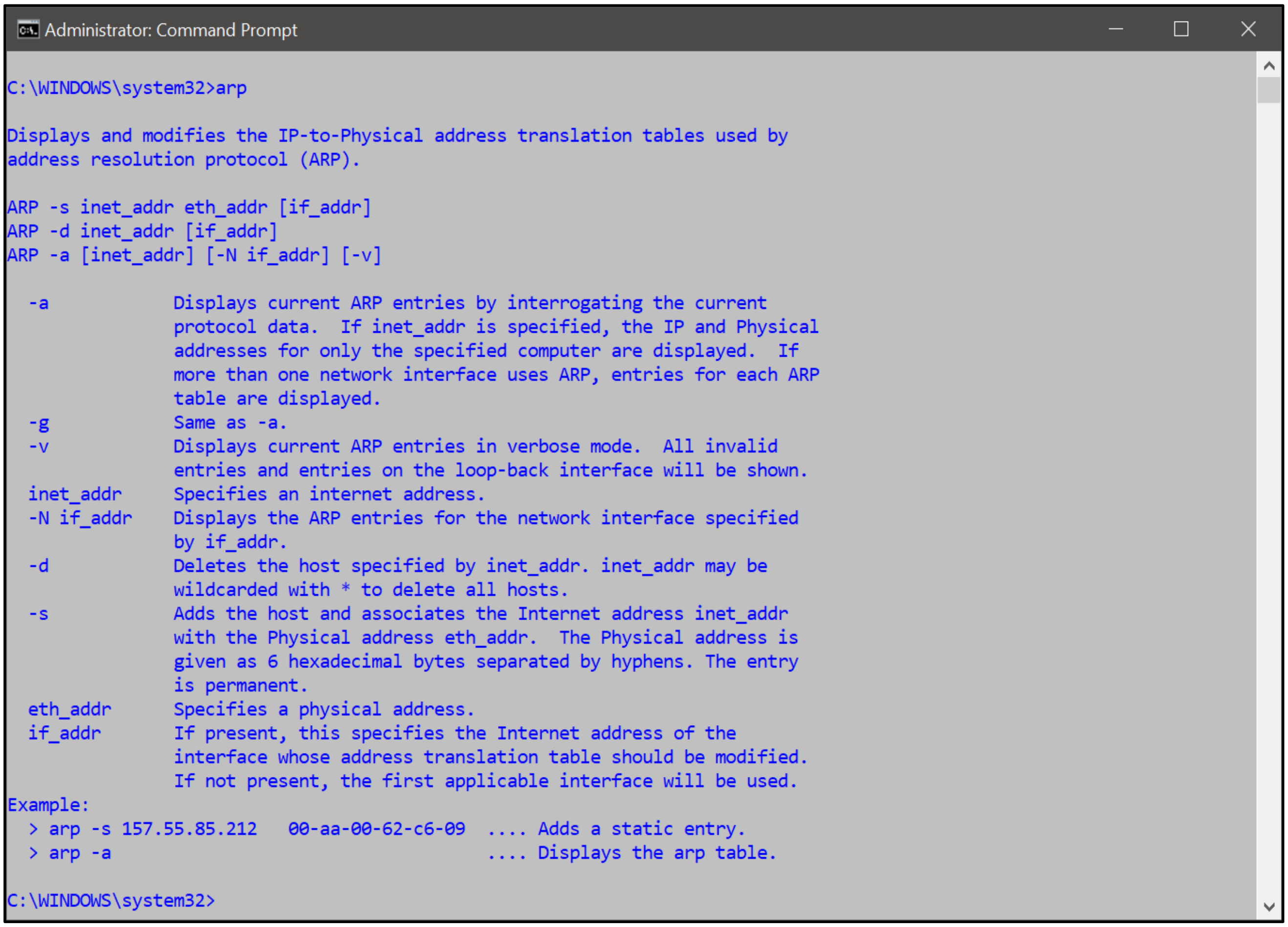
Note

A *broadcast* is a type of network communication that's not directed to any specific device. It's the equivalent of speaking without addressing anyone specifically. Unless you know the MAC address of a computer, you must call out to the local network, which is the equivalent of saying, "Hey you with the IP address of 192.168.1.175. What's your MAC address?" All the other nodes on the network hear this but disregard it because they know their IP address is not 192.168.1.175. Don't worry, we'll talk about IP addresses later.

When the requesting host receives the physical address, both the IP and physical addresses are stored locally as an entry in the ARP cache. All hosts maintain an ARP cache that includes their own IP-to-physical address (MAC address) mapping. The ARP cache is always checked for an IP-to-physical address mapping before initiating a broadcast. In Windows, ARP cache entries expire after 10 minutes. Although the potential life of an entry is 10 minutes, if it remains unused for two minutes, it's removed from the cache.

ARP will only ever resolve the MAC addresses for local hosts. Communications to remote hosts occurs through the default gateway or some other router. In this situation, it's the router's MAC address that would be resolved.

You can use the **arp.exe** command-line tool in Windows to examine the ARP cache, and where necessary, to make changes to the entries in the cache. Figure 4 is output from the **arp.exe** command. You can use the switches displayed in the output to manage the ARP cache.



```
C:\WINDOWS\system32>arp

Displays and modifies the IP-to-Physical address translation tables used by
address resolution protocol (ARP).

ARP -s inet_addr eth_addr [if_addr]
ARP -d inet_addr [if_addr]
ARP -a [inet_addr] [-N if_addr] [-v]

  -a            Displays current ARP entries by interrogating the current
                protocol data.  If inet_addr is specified, the IP and Physical
                addresses for only the specified computer are displayed.  If
                more than one network interface uses ARP, entries for each ARP
                table are displayed.
  -g            Same as -a.
  -v            Displays current ARP entries in verbose mode.  All invalid
                entries and entries on the loop-back interface will be shown.
  inet_addr     Specifies an internet address.
  -N if_addr    Displays the ARP entries for the network interface specified
                by if_addr.
  -d            Deletes the host specified by inet_addr. inet_addr may be
                wildcarded with * to delete all hosts.
  -s            Adds the host and associates the Internet address inet_addr
                with the Physical address eth_addr.  The Physical address is
                given as 6 hexadecimal bytes separated by hyphens. The entry
                is permanent.
  eth_addr      Specifies a physical address.
  if_addr       If present, this specifies the Internet address of the
                interface whose address translation table should be modified.
                If not present, the first applicable interface will be used.
Example:
  > arp -s 157.55.85.212   00-aa-00-62-c6-09  .... Adds a static entry.
  > arp -a                                    .... Displays the arp table.

C:\WINDOWS\system32>
```

Figure 4. Output from the **arp.exe** command

ICMP

ICMP is another of the maintenance protocols. It's a mechanism for reporting errors resulting from delivery problems. It also allows two devices on an IP network to share status and error information.

For example, the ping tool uses the ICMP echo request and echo reply packets to determine whether an IP system on a network is functional. When you ping a remote host, your computer sends a sequence of packets to the target host. The target host receives the packets and replies with its own sequence of packets. For this reason, the ping tool is useful for diagnosing IP network or router failures.

ICMP messages are contained within IP datagrams. This ensures that the ICMP message will be able to find its way to the appropriate host on the internet. The most common ICMP messages are "echo request," "echo reply," "redirect," "source quench," and "destination unreachable." ICMP source quench is used to ask a source host to slow down. This is useful where a source host is sending packets to a remote destination host at a rate that's saturating routers.

Summary

The following table summarizes the various protocols in the TCP/IP suite.

Protocol	Explanation
ARP	Resolution of IP-to-physical address
IP	Routing functions
ICMP	Reporting errors
TCP	Connection-orientated delivery service
UDP	Connectionless datagram service

Devices, protocols, and applications

As previously discussed, the OSI reference model layers are:

- Application
- Presentation
- Session
- Transport
- Network
- Data-link
- Physical

Not all network protocol stacks implement all layers, but they broadly map to the model. But which components operate in these seven layers? Figure 5 compares the OSI model with TCP/IP.

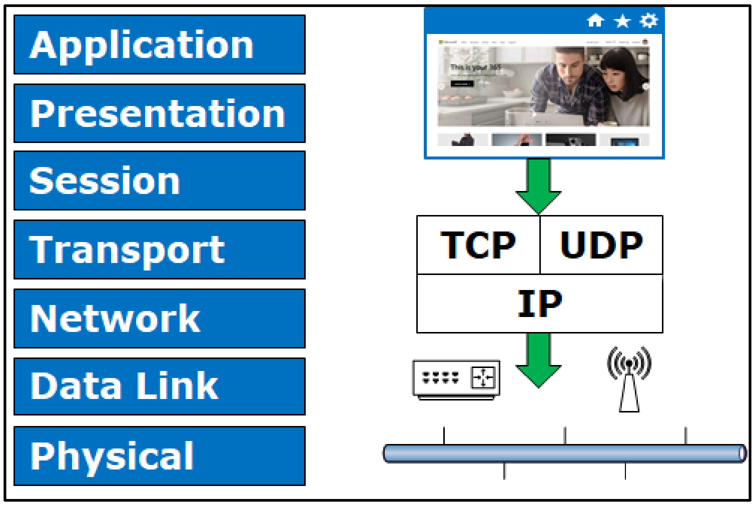


Figure 5. The OSI model compared with TCP/IP

Devices

Let’s start at the beginning—physical devices operate at the lower levels of the model. Specifically, cabling and infrastructure components such as hubs operate at the physical layer. Components such as network adapters (installed in computers and servers), bridges, and layer 2 switches operate at the data-link layer. Routers and layer 3 switches operate at the network layer but can still be said to be physical devices because they’re tangible components that provide a specific function.

Note

A *bridge* is a device that operates at layer 2. Bridges operate in *promiscuous mode* on attached networks, meaning they receive all network frames on all connected interfaces. They then forward all those frames to all other interfaces.

Bridges provide error checking on the data-link and enable you to extend a network over larger distances. However, they typically don’t provide traffic management capabilities.

Hosts don’t specifically address bridges, which means that a bridge isn’t visible to a host. Layer 2 switches are devices that you can configure to behave like a bridge between configured interfaces.

Protocols

Next, protocols or network protocol stacks operate at the network and transport layers. These components are implemented as software in hosts and devices such as layer 3 switches and routers. All networked devices implement network protocols. Today, that’s typically going to be TCP/IP. Unlike devices, you can’t observe or touch network protocols.

Note

Unlike bridges and layer 2 switches, hosts specifically address routers and layer 3 switches. When a host wants to communicate with a device in another network or subnet, it must have a route to that subnet; failing that, it sends its packets to its configured router, sometimes called a *default gateway*. The router is responsible for onward forwarding. Unlike bridges, routers only propagate network packets that are specifically addressed to a remote subnet. This enables routers to help manage network traffic by localizing subnet traffic.

Going back to our package analogy, let's say that you want to send a package to an address in New York City. You have the address, so you could determine a way to deliver the package yourself by calculating a route to the package's destination. However, if you've never been to New York City, you'd be better off mailing the package and letting the mail service provider deliver it for you. In this scenario, you could think of the mail service provider as being your default gateway. If you must send several packages, you'd likely consult a specialist who knows the routes to every destination, rather than spending your time inefficiently, learning every route. That's what a router/layer 3 switch does.

Applications

Finally, at the upper layers of the protocol stack are applications. These might be server-side components such as a database server, a web server, or an email server. At the client end, these might be web browsers, email clients, and office productivity software.

Client applications communicate with server applications by opening a protocol port at the server end. In TCP/IP, when this port is associated with a transport protocol (TCP or UDP) and an IP address (IPv4 or IPv6), it creates a socket.

Frames, packets, and datagrams

As each layer of the network protocol stack communicates with the layer under it, a process known as *encapsulation* occurs. The element from the upper layer is placed into a container, and a header and footer (checksum) are added to ensure correct delivery to the corresponding layer of the remote host.

When an element from an upper layer is too big to fit into the designated container structure at a lower level, fragmentation occurs. In this process, the structure from the upper layer divides into smaller pieces. Each small piece is inserted into a container, prepended with a header, and appended with a checksum. The container then passes down the protocol stack in sequence, as Figure 6 depicts.

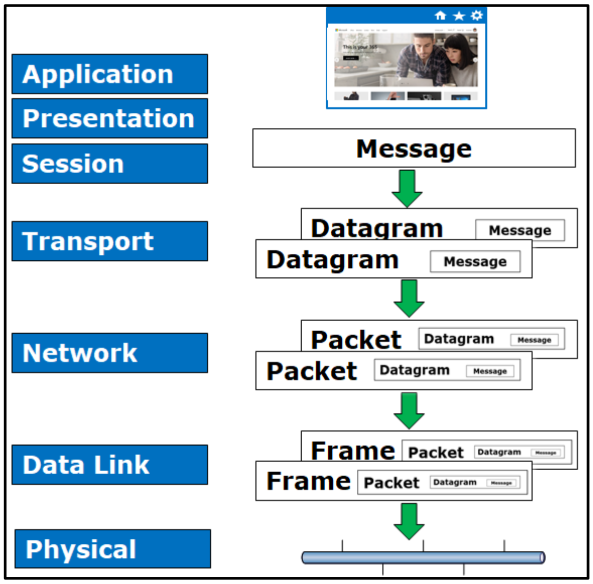


Figure 6. Encapsulation and fragmentation

At the remote end, the lowest-level container is received on the physical layer. The host computer’s network adapter examines the header, and based on addressing, determines whether the received data is relevant. If it is, the header and checksum are removed, and the resulting structure is passed back up the stack for the process to be repeated. This process ensures that for each layer, data passes to the correct component above.

Note

To better understand fragmentation and reassembly, let’s use another example. Suppose that you have a model airplane that you want to send to your friend. You could send it as a single package with an address written on the envelope, thereby avoiding having to fragment your package. You’ll need to put a header on both the address and the envelope.

Now imagine that we want to deliver a bicycle to the same friend. Fully assembled, the bike won’t fit inside the biggest box you have. So, you dismantle the bike, place the pieces into a series of boxes, and provide instructions for reconstructing the bike. You also stick an address label on each box, and then you send the boxes to your friend. Upon receipt, your friend opens each box, and using the instructions that you provided, rebuilds the bike. When done with data packets, this is fragmentation and reassembly.

In a TCP/IP network, applications use messages. At the transport layer (TCP and UDP), datagrams are used, although, these are often still referred to as *packets*. At the network layer (IPv4 and IPv6), packets are the container used to pass data up and down the stack. At the data-link layer, frames are used. Finally, frames are transmitted onto the wire in streams of binary bits.

TCP datagram structure

The following table describes the contents of a TCP datagram.

Field	Explanation
Source port	TCP port of the sending host
Destination port	TCP port of the destination host
Sequence number	Ensures that all bytes have been received
ACK number	Sequence number of the next byte
Data length	Length of the TCP segment

Field	Explanation
Data	Datagram’s payload—the message, or part thereof, from the upper layer application
Reserved	Reserved
Flags	Describes the content in the segment
Window	How much space is in the TCP windows
Checksum	Ensures validity of the header
Urgent pointer	For urgent in-transit data, this specifies the end of that data in the segment

UDP datagram structure

The following table describes the contents of a UDP datagram.

Field	Explanation
Source port	UDP port of the sending host
Destination port	UDP port of the destination host
Message length	The size of the UDP message
Data	The payload of the datagram—the message, or part thereof, from the upper layer application
Checksum	Verifies the header

IP packet structure

The following table describes the contents of an IP packet.

Field	Explanation
Source IP address	Identifies the sender of the datagram by IP address
Destination IP address	Identifies the destination of the datagram by IP address
Protocol	Indicates whether data should pass to UDP or TCP at the destination host
Data	The payload of the packet—the datagram, or part thereof, from the transport protocol
Checksum	Field used to verify the packet's integrity upon arrival at the destination
Time to Live	Number of seconds a datagram can stay on the network before being discarded. Otherwise, packets could endlessly loop around a network. A router will decrease the Time to Live (TTL) by at least 1 second when it handles the packet, and it's required to decrement the TTL by at least the time spent on the router.

Ethernet frame structure

Although several network topologies exist, by far the most common is Ethernet. Ethernet is a widely used physical network topology, despite being based on a 1970s technology that remains largely unchanged.

Ethernet operates at the OSI model's data-link and physical layers. Data at this layer is encapsulated, transmitted, and received in frames.

The following table describes the contents of an Ethernet frame.

Field	Explanation
Preamble	Indicates that a frame is following
MAC destination	The MAC address of a target host. If the target host is in the same subnet as the sender, this is the target host MAC address. If the target host is in a different subnet, this is the MAC address of the router (default gateway).
MAC source	The sender’s MAC address
Ethertype	Specifies the type of Ethernet
Payload	The data stored in the frame—the packet, or part thereof, from the protocol layer above
Frame check sequence	A field that’s used to verify the frame’s integrity upon arrival at the destination
Interpacket gap	Marks the end of the frame

Well-known ports and their uses

To ensure that communication can be established reliably, port numbers are assigned to processes that are running on hosts. When a client needs to communicate with a service that’s running on a remote host, it creates a UDP or TCP datagram, and within the header for the datagram, it identifies what application it wishes to communicate with by declaring the remote port number. The header also contains a source port number so that the remote host has a return application.

Note

You can think of ports as being a little like calling a hotel and asking for a room number to speak with your friend. The room number ensures that you speak to the correct person. Similarly, the port number ensures that the correct app is selected.

Ports are identified with a number between 0 and 65,535. Each sockets-based application identifies itself with a unique protocol port number. The operating system on the client side dynamically assigns port numbers when a service is requested. However, port numbers for well-known server-side applications are pre-assigned by the Internet Assigned Numbers Authority (IANA) and don't usually change. Figure 7 illustrates well-known port numbers.

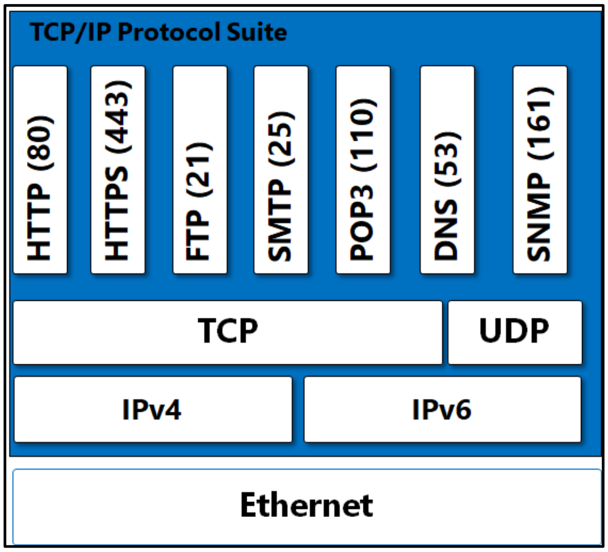


Figure 7. Well-known port numbers

A *socket* is the endpoint for a network communication. You create a socket by designating the IP address of a destination host, the type of service (TCP or UDP), and the port to use.

The following table identifies common, well-known TCP port numbers and their associated apps.

TCP port	Explanation
21	File Transfer Protocol (FTP)
25	Simple Mail Transfer Protocol (SMTP), used by mail servers to relay email
53	Domain Name System (DNS)
80	Web server using HTTP
88	Kerberos protocol, used to sign in to an Active Directory Domain Services (AD DS) domain
110	Post Office Protocol version 3 (POP3), used for basic email client retrieval
143	Internet Message Access Protocol 4 (IMAP4), also used for basic email client retrieval
139	NetBIOS name service
389	Lightweight Directory Access Protocol (LDAP), also used by AD DS for the domain-based directory service
443	HTTPS, web service over Secure Sockets Layer (SSL)
993	POP3 over SSL
995	IMAP4 over SSL
3268	Global catalog, used by AD DS for the forest-wide directory service

The following table lists common, well-known UDP port numbers and their associated apps.

UDP port	Explanation
15	NETSTAT, used to identify the current network status
53	DNS
67	DHCP server-side port, used to offer an IP address
68	DHCP client-side port, used to obtain an IP address
69	Trivial File Transfer Protocol (TFTP)
137	NETBIOS-NS, NetBIOS name service
138	NETBIOS-DGM, NetBIOS datagram service
161	Simple Network Management Protocol (SNMP), network monitor

Note

When two hosts communicate through a firewall, the firewall might prohibit the passing of certain ports for security purposes. As a network administrator, you must be aware of which ports should be open—that is, what traffic the firewall should let pass—so that your network applications function without compromising security. We'll be learning more about firewalls later in this module.

How network connections are established

Network communication typically begins with a connection request to another host, using its computer name. However, to communicate, the requesting host must know the MAC address of the receiving host's network interface. Conversely, the receiving host also needs to know the requesting host's MAC address, as Figure 8 depicts.

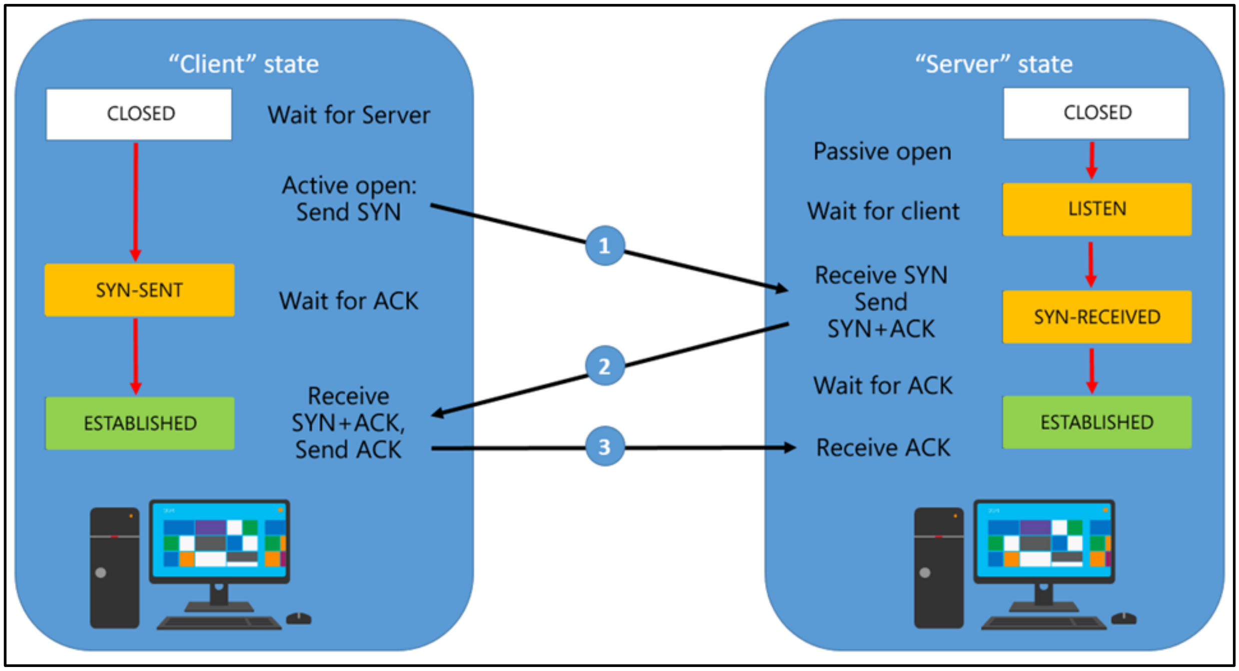


Figure 8. Network communication

After the requesting host discovers the MAC information, it caches it locally. As you know, a MAC address is often a hard-coded unique identifier that's assigned to network interfaces by manufacturers of network adapters.

Several steps occur before the requesting host can find the receiving host's MAC address. The following steps are a high-level overview of this process:

1. A host sends a request to connect to Server1. The name Server1 must be resolved to an IPv4 address.

2. After the sender knows the recipient's IPv4 address, it uses the subnet mask to determine whether the IPv4 address is remote or on the local subnet.

3. If the address is local, the sender broadcasts an ARP request on the local subnet. If it’s remote, the sender sends an ARP request to the default gateway, and the host routes it to the correct subnet.

4. The host that owns that IPv4 address responds with its MAC address and a request for the sender’s MAC address.

5. After the exchange of MAC addresses, IPv4 communication negotiation and the exchange of IP data packets takes place.

Establish a TCP session, or how to do a three-way handshake

When a host starts a TCP session with another host, a three-way handshake occurs. The TCP session starts with both the initiating host (*client*) and responding host (*server*) in a closed state. The client starts a TCP session, and a three-way handshake proceeds as described in the following high-level steps:

1. The client sends a sync sequence numbers message—known as a synchronization (SYN)—to the server. The client then enters a waiting state.

2. The server, in the passive listening state, receives the SYN from the client and responds by sending an acknowledgment (ACK) together with its own SYN. The server now enters a waiting state.

Note

The SYN is a request to the server to sync the sequence numbers that the client sent. To initialize a connection, the client and server must sync each other's sequence numbers.

3. The client receives the SYN and the ACK from the server, and it responds with an ACK. The server receives the ACK, and the session now begins.

This three-way handshake helps set up and maintain a reliable TCP session between communicating hosts. UDP datagrams don’t require a three-way handshake because UDP is a best-effort delivery protocol.

Lesson 2: What is the internet, an intranet, and an extranet?

Consider a startup company—perhaps a little coffee house named Fourth Coffee. Initially, its IT needs are simple, such as wireless customer access to the internet and a separate network for employees to conduct business.

But business is booming for Fourth Coffee, and the owners decide to open new branches. This means they'll need to manage their inventory more efficiently, and they'll need to implement digital payment solutions. They'll also need to interconnect their head office with their branch offices, distribution centers, and coffee shops.

By connecting the company's private networks together and by defining the type of traffic that can traverse these networks, Fourth Coffee is beginning to create intranets. By linking its business with its suppliers and corporate customers, Fourth Coffee is creating extranets. It's an exciting time for Fourth Coffee, but it's a stressful time for the IT staff that's responsible for putting it all together. In this lesson, you'll learn about the internet and how groups of interconnected networks form intranets and extranets.

Objectives

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

- Describe the internet.
- Explain intranets.
- Explain extranets.

The internet

It’s sometimes surprising to learn that the internet has been with us for many decades. Originating during the Cold War in the late 1960s as the Advanced Research Projects Agency Network (ARPANET), it evolved significantly from its military origins. The early ARPANET was based on the network control program (NCP) protocol. By the early 1980s, however, NCP was replaced by the now ubiquitous TCP/IP protocol, laying the foundations for the modern internet.

It wasn’t until the mid-1990s that commercial organizations began to take advantage of the internet and the services it could provide, such as the World Wide Web, which is 30 years old at the time of writing (March 2019). Today, the internet provides individuals and organizations with many services, including transport for organizational email, web servers, remote access for home-based workers, and much more.

As a network administrator, you have no control over the internet, because it’s a public network. However, you do have control over how your organization’s users and devices can access and use the internet. You must also devise strategies for connecting to and from the internet from your organization’s network infrastructure.

Intranets

An *intranet* is most accurately described as a collection of TCP/IP networks that are interconnected by router devices. These routers might interconnect networks in the same building or campus, or they might define interconnections between networks that are distributed across the globe. The key characteristic of an intranet is that it belongs to a specific organization. Therefore, that organization has control over the devices and services that connect to its infrastructure. Intranets are also known as *private networks*.

Organizations that connect their intranets to the internet potentially expose themselves to a myriad of threats, such as malicious hackers and data loss. To protect themselves from these and other threats, organizations usually create perimeter networks to help provide security.

Extranets

Most organizations must interact with other organizations outside of their intranets in an environment known as *business to business*. While this interaction can be on an ad hoc or informal basis, it's increasingly more common for organizations to link their network infrastructures together. For example, if you wanted to sell your products through a third party's storefront, you might need to make your inventory accessible to the organization and platform that hosts the storefront. Networks that are linked between organizations are known as *extranets*.

Allowing another organization to have unrestricted access to your organizational infrastructure is not optimal. Therefore, you must identify and clearly define the level of allowable interaction between your two organizations. Then you must implement appropriate technology to facilitate and control the required access. This might involve defining a method of connecting to the other organization's network and determining a means of authentication and authorization to access appropriate resources.

Lesson 3: Firewalls and security zones

Protecting your organization's data from malicious attacks is of utmost importance for any network administrator. By implementing firewalls, you can help prevent unauthorized network traffic from entering or existing in a device. By configuring a series of firewall and router devices within your organization's network infrastructure, you can create security zones. For example, a college creates a public zone on its network for students, parents, researchers, and others. At the same time, the college also restricts access to its internal, organizational, and administrative resources.

In this lesson, you'll learn about firewalls and how to implement them to create and manage security zones.

Objectives

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

- Describe common network-security threats and mitigations.
- Describe firewalls.
- Explain how to create security zones by using firewalls.

Common network-security threats and mitigations

Network security can be compromised in many ways. However, we can group these security threats into categories.

Common threats

Common network-based security threats include:

- Eavesdropping. An eavesdropping attack, also known as *network sniffing*, occurs when an unauthorized user captures data (or *network packets*) that workstations connected to your network are sending and receiving. Eavesdropping attacks can compromise your organization's sensitive data, such as usernames and passwords, which can lead to other, more damaging attacks.

- Denial of service (DoS) attack. This type of attack limits the function of a network app or renders an app or network resource unavailable. Malicious hackers can initiate a DoS attack in several ways. Often, they're aware of vulnerabilities in the target app, which they then exploit to render it unavailable. Malicious hackers typically perform DoS attacks by overloading a service that replies to network requests to shut down the service or server that hosts the service. A distributed denial of service (DDoS) attack is a version of a DoS attack.

- Port scanning. Apps that run on a computer using the TCP/IP protocol use TCP or UDP ports to identify themselves. One way that malicious hackers attempt to exploit a network is to query hosts for open ports on which they listen for client requests. After the malicious hackers identify an open port, they can use other techniques to access the services that are running on the computer.

- Man-in-the-middle (MITM) attack. The malicious hacker uses a computer to impersonate a legitimate host on the network your computers are using to communicate. The malicious hacker intercepts all the communications that are intended for a destination host. The malicious hacker might observe the data in transit between the two hosts, but they can also modify that data before forwarding the packets to the destination host.

Mitigations

Malicious hackers try to access your network by using a variety of tools and techniques. After they find a way into your network, they can exploit that success and take their attack further. Therefore, it's important to implement a comprehensive approach to network security to help ensure that one loophole or omission doesn't result in further weaknesses and exposure. You can use any or all the following defense mechanisms to help protect your network from malicious hackers:

- Internet Protocol security (IPsec). This authenticates IP-based communications between two hosts, and where desirable, encrypts that network traffic.

- Firewalls. These allow or block network traffic based on the type of traffic.

- Security zones. These are isolated areas on your network to and from which you can define network traffic flow. When you must make network services available on the internet, as a best practice, don't connect hosting servers directly to the internet. Instead, place these servers in a perimeter network, thereby making them available to internet users without allowing those users access to your organization's intranet.

- VPNs. These authenticate and encrypt connections between remote users and your organization's intranet. It's important that users can connect to their organization's intranets from the internet as securely as possible. The internet is a public network, and data in transit across the internet is susceptible to eavesdropping or MITM attacks. You can help mitigate this risk by using VPNs.
- Intrusion detection. You can proactively monitor your network and search for symptoms of intrusion. It's important to implement the preceding techniques to secure your network and continue to monitor your network regularly for signs of attack. You can use intrusion-detection systems to do this by implementing them on perimeter devices, such as internet-facing routers.

What are firewalls?

A *firewall* is a security solution that establishes a barrier between two or more networks. Firewalls block or allow network traffic between these networks based on the traffic's properties. You can use hardware-based firewalls or software-based firewalls that run on a device.

Depending on your firewall's sophistication, you can configure it to block or allow traffic based on the following traffic properties:

- Source address
- Destination address
- Source port
- Destination port
- Protocol
- Packet contents

For example, a sophisticated firewall analyzes network traffic and filters out harmful traffic such as attempted DoS or SQL injection attacks. Administrators often place firewalls at a network perimeter between an organization's screened subnet and the internet and between the screened subnet and the internal network. We'll get to this a little later.

Today, it's also common for each computer to have its own additional firewall, which is known as a *host firewall*.

What are security zones?

Security zones help network administrators control the flow of network traffic between devices within an organization's network infrastructure. For example, you might decide to restrict the flow of traffic to and from the internet. Between the sites that make up your internal network infrastructure, you decide to be less restrictive. You know that certain services must be available to your users even if their devices aren't connected to the organizational network. Finally, you decide to offer wireless network access to the internet for visitors to your offices.

You can address these needs by deploying multiple firewalls to segment your network into security zones. These firewalls control the flow of designated traffic between the zones in your network. Typically, an organization might implement an internal network zone, a perimeter network zone, and the internet zone, as Figure 9 depicts:

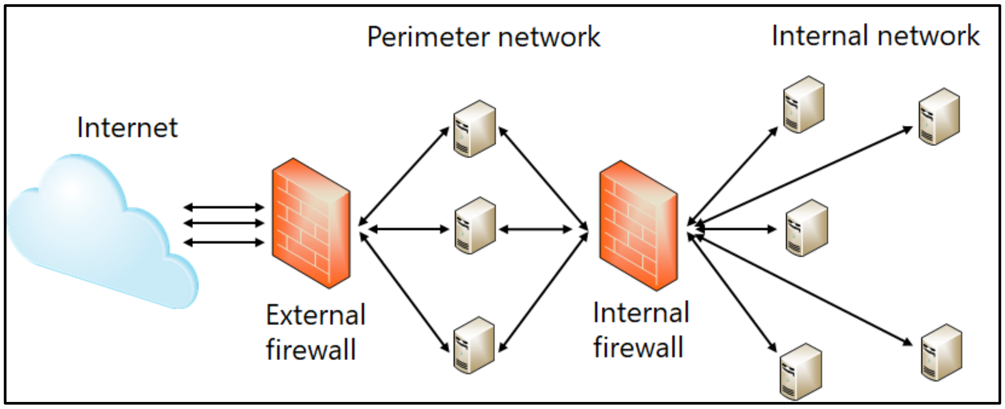


Figure 9. Security zones

A *perimeter network* is the network between an external and an internal firewall. No traffic can pass directly from the internet to the protected internal network, and no traffic can pass directly from the protected internal network to hosts on the internet. Instead, all traffic must traverse a host on the perimeter network.

You should configure an external firewall so that traffic can only pass to hosts on the perimeter network using specific ports. For example, you should configure incoming traffic on TCP port 443 to route to a secure web server on the perimeter network (in this case, using HTTPS).

Similarly, you should configure an internal firewall so that traffic that traverses the internal firewall can only pass if it uses specific ports. For example, traffic on TCP port 443 from the internal network to the secure web server on the perimeter network should be allowed, but traffic on port 443 from the secure web server on the perimeter network to the internal network need not be.

The network perimeter design model has the following benefits:

- If a host on the perimeter network, such as a web server, is compromised by a malicious hacker, a firewall still blocks the malicious hacker from accessing hosts on the internal network.
- It allows specific services to be made available to the internet in a protected manner without exposing hosts on the internal network.
- It blocks direct communication between hosts on the internet and hosts on the internal network, and it blocks direct communication from hosts on the internal network and hosts on the internet. This makes it difficult for a malicious hacker to access hosts on the internal network because the traffic flow is being restricted.

You typically deploy the following server roles on perimeter networks:

- External web server. The external web server should only contain content that your organization must make available to the public. Because sensitive information must not be stored here, ensure that only public information is published. Intranet servers with sensitive, restricted information should be hosted on trusted internal networks.
- Web proxy server. Clients on the internal network use this server to access web-related content on the internet. It also stores cached copies of commonly accessed sites. You can also configure this server to check web content for malware and to block clients on the internal network from accessing certain sites and content.
- Reverse web proxy. Use this server role to enable remote, internet-based access to specific services on your internal network without placing the servers that are hosting those services in the perimeter network. You can publish those services by using a reverse web proxy. For example, you can publish access to users' email by using a reverse proxy, making workplace email available on users' own phones through a specific URL. The Web Application Proxy feature in Windows Server can be used in this way.

- SMTP relay. This server routes mail traffic into and out of the organization. You can configure this server to block unsolicited commercial email, filter malware, and block messages with confidential information from being sent by people on the internal network.
- DNS forwarder. This server forwards DNS (name resolution) requests from DNS servers on the internal network to DNS servers on the internet.
- VPN server. To enable your users' devices to connect from the internet to your organization over a VPN, you must place VPN servers in the perimeter network. Note that although a device in the perimeter might establish the VPN connection, a device on the internal network can authenticate the remote access attempt if you open the required ports on the internal firewall. For the external firewall, the ports you must open vary depending on the type of VPN you decide to implement.

Lesson 4: Virtual private networks

More often, employees are working while away from their offices. Sometimes they work from home and sometimes their work necessitates travel. VPNs provide a secure way of accessing internal data and applications from user devices that are connected to the internet. To support a VPN environment, you must understand tunneling protocols, VPN authentication, and other configuration options. This lesson describes these technologies.

Objectives

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

- Explain the need for remote access.
- List and describe remote access solutions.
- Provide an overview of VPNs.
- Describe the available VPN tunnel types.
- Describe the available VPN authentication methods.

The need for remote access

Even if you're working at home, sitting on a subway, or waiting at an airport departure gate, chances are you might have to work. This might involve submitting a paper, responding to an email, or contributing to a team project. Remote access technologies provide more secure access to your organization's infrastructure from different locations. While organizations usually own and protect LANs entirely by themselves, remote connections to servers, shares, and apps must often travel across unprotected and unmanaged networking infrastructures such as the internet. Any method of using public networks for the transit of organizational data must include a way to protect the integrity and confidentiality of that data.

The type of remote access technology that an organization implements depends on its business requirements. Some organizations might deploy several remote access technologies on different servers, and some might deploy them on the same server.

Remote access solutions

Several remote access technologies are available, including:

- VPN. VPNs enable users that are working offsite—for example, from home, a customer site, or a public wireless access point—to access a server on their organization's private network by using the infrastructure that a public network such as the internet provides. From the user's perspective, the VPN is a point-to-point connection between a computer, the VPN client, and their organization's server. The exact infrastructure of the shared or public network is irrelevant because it appears as if the data is sent over a dedicated private link.

- DirectAccess. DirectAccess is a Windows feature that enables remote users to securely access organizational resources such as email servers, shared folders, and internal websites without connecting to a VPN. DirectAccess is based on IPv6 and related tunneling technologies. However, it only supports AD DS–joined devices that are running Windows 10 Enterprise edition. Note that DirectAccess is being replaced with a newer generation of VPNs that support Always On access.

- Reverse web proxy. Reverse web proxy provides access for users who must connect to their organization's internal web applications from the internet. The reverse web proxy, such as the Web Application Proxy server role, deploys in the perimeter network. Publishing rules define what type of network applications are available from the internal network to users who are connected to the internet.

Overview of VPNs

A VPN provides a connection between components of a private network through a public network such as the internet, and tunneling protocols allow a VPN client to make and maintain a connection to a virtual port "listening" on a VPN server, as Figure 10 depicts.

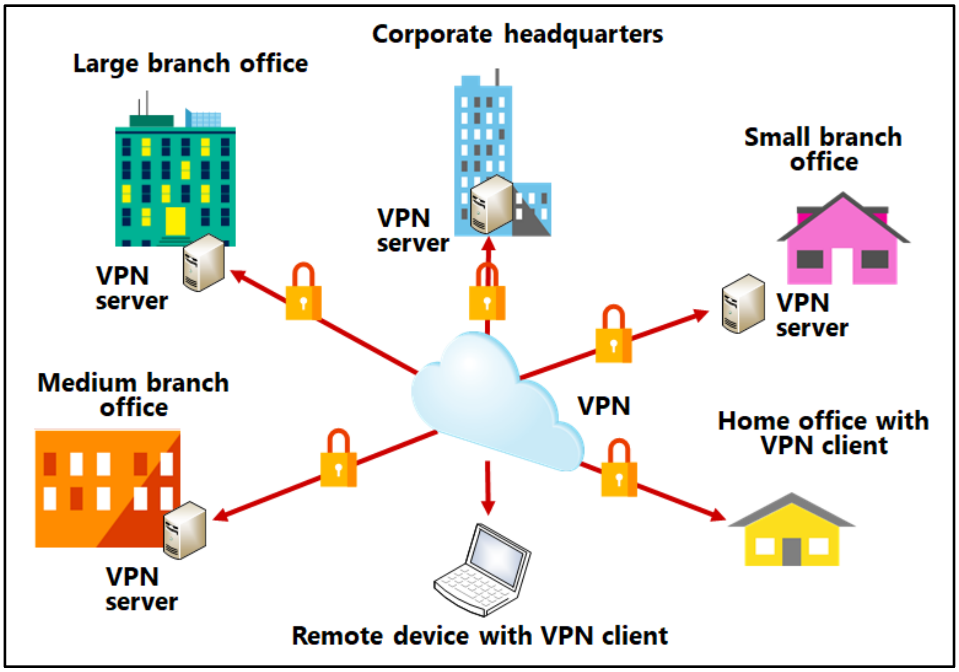


Figure 10. Overview of VPNs

Tunneling protocols are referred to as such because they create a pathway, or *tunnel*, through one environment to connect two parts of another environment. Think of an actual tunnel. It might enable a rail service to connect two separate countries/regions, passing beneath a sea to do so. The tunnel passes through the sea to connect two land masses together.

To emulate the point-to-point link, the VPN client encapsulates the data and prefixes it with a header. The header provides routing information that enables the data to traverse the shared or public network to reach its endpoint.

To emulate a private link, the VPN client encrypts data, which helps ensure confidentiality. Without encryption keys, packets that are intercepted on a shared or public network are indecipherable. The VPN client encapsulates and encrypts private data on the private link or on the VPN connection. The two types of VPN connections are remote access VPN and site-to-site VPN.

Remote access VPN

Users who work from home, at a customer site, or from a public wireless-access point can use remote access VPN connections to access a server on their organization's private network. The remote access VPN connection uses the infrastructure that a public network provides, such as the internet.

Note

From a user's perspective, the exact infrastructure of the shared or public network is irrelevant because it appears logically as if it's sending the data over a dedicated private link. The user doesn't know or care that they're connected by using a VPN. To them, it acts just the same as a local connection.

Site-to-site VPN

Site-to-site VPN connections, also known as *router-to-router*, enable organizations to have routed connections. These connections could be between separate offices or between one office and another organization over a public network. This VPN connection type helps maintain secure communications.

A routed VPN connection across the internet operates logically as a dedicated WAN link. When networks connect over the internet, a router forwards packets across a VPN connection to another router.

A site-to-site VPN connection connects two portions of a private network. The VPN server provides a routed connection to the network to which the VPN server is attached. The calling router authenticates itself to the answering router.

In a site-to site VPN connection, the packets sent from either router across the VPN connection typically don't originate at the routers. In other words, the site-to-site connection isn't visible to the computers that use the link.

Properties of VPN connections

VPN connections have the following properties:

- Encapsulation. When you use VPN technology, it encapsulates private data with a header that contains routing information. This information allows the data to traverse the transit (public) network.

- Authentication. *Authentication* is the process of identifying oneself. In real life, you might do this with a passport or driver's license. In computing, authentication is used to identify computers to one another and to identify users to servers.

- Data encryption. To help ensure data confidentiality as it traverses the shared or public transit network, the sender encrypts the data, and the receiver decrypts it. The encryption and decryption processes depend on the sender and the receiver both using a common encryption key. Data packets that are intercepted in the transit network are unintelligible to anyone who doesn't have the common encryption key.

  The encryption key's length is an important security parameter. You can use computational techniques to determine the encryption key. However, such techniques require more computing power and computational time as the encryption keys get longer. Still, it's important to use the largest possible key size to ensure data confidentiality, even if this incurs a performance cost.

VPN tunnels

You can choose from several different tunneling protocols. Windows 10 supports the following VPN tunneling protocols:

- Point-to-Point Tunneling Protocol (PPTP). You can use PPTP for both remote access and site-to-site VPN connections. When using the internet as the VPN public network, the PPTP server is a PPTP-enabled VPN server with one interface on the internet and a second interface on the intranet. PPTP enables you to encrypt and encapsulate data in an IP header multiprotocol traffic that's then sent across an IP network or a public IP network, such as the internet.

- Layer Two Tunneling Protocol (L2TP). L2TP enables you to encrypt multiprotocol traffic, which you can then send over any medium that supports point-to-point datagram delivery, such as IP or asynchronous transfer mode (ATM). L2TP is a combination of PPTP and Layer Two Forwarding (L2F).

- Secure Socket Tunneling Protocol (SSTP). SSTP is a tunneling protocol that uses the HTTPS protocol over TCP port 443 to pass traffic through firewalls and web proxies that otherwise might block PPTP and L2TP traffic. SSTP provides a mechanism to encapsulate PPP traffic over the SSL channel of the HTTPS protocol. Using PPP allows support for strong authentication methods, such as Extensible Authentication Protocol-Transport Layer Security (EAP-TLS). SSL provides transport-level security with enhanced key negotiation, encryption, and integrity checking.

- Internet Key Exchange version 2 (IKEv2). IKEv2 uses the IPsec tunnel mode protocol over UDP port 500. IKEv2 supports mobility, meaning that it can automatically reconnect when needed. Consequently, IKEv2-based VPNs enable users to more easily move between wireless hotspots or between wireless and wired connections.

VPN authentication

Authentication of access clients is an important part of security. Authentication methods typically use an authentication protocol that’s negotiated during the process of establishing a connection. The following methods are available, depending on the VPN technology you’re using:

- Password Authentication Protocol (PAP). This method is the least secure authentication protocol because it uses plaintext passwords. It typically is negotiated if the remote access client and remote access server can’t negotiate a more secure form of validation. Windows Server 2016 and Windows 10 include PAP to support older client operating systems that don’t support other authentication methods.

- Challenge Handshake Authentication Protocol (CHAP). This method is a challenge-response authentication protocol that uses the industry-standard MD5 hashing scheme. Various vendors of network access servers and clients use CHAP. However, CHAP requires the use of a reversibly encrypted password, so you should consider using another authentication protocol, such as Microsoft CHAP version 2 (MS-CHAPv2).

- MS-CHAPv2. This Microsoft authentication is a more secure implementation of CHAP because it doesn't require storing a reversibly encrypted password.
- Extensible Authentication Protocol (EAP). With this method, an arbitrary authentication mechanism authenticates a remote access connection. The remote access client and the authenticator (typically the VPN server) negotiate the exact authentication scheme to use.

The following table summarizes these authentication options.

Method	Explanation
PAP	Uses plaintext passwords. Typically used when the remote access client and remote access server can't negotiate a more secure form of validation.
CHAP	A challenge-response authentication protocol that uses the industry-standard MD5 hashing scheme to encrypt the response.
MS-CHAPv2	An upgrade of MS-CHAP; two-way authentication, also known as *mutual authentication*, is provided. The remote access client receives verification that the remote access server that it's dialing in to has access to the user's password.
EAP	Allows for arbitrary authentication of a remote access connection by using authentication schemes, known as *EAP types*.

Learning in action: Planning a network

Scenario

You have just taken a new job in IT with Lucerne Publishing in London. This small publishing company is about to take off in a big way, having signed some great new authors and secured the movie rights for several its books. The IT infrastructure at Lucerne Publishing is a little dated, and your first job is to modernize its infrastructure.

Answer the following questions.

1. Some of the editors work from home, and occasionally, from restaurants where they're meeting with authors, agents, and producers. Management would like you to enable secure access to the Lucerne Publishing intranet from the editors' phones while they're away from the office. Specifically, they need access to their email, which is currently hosted on a Microsoft Exchange server in the London office. It's possible to access mailboxes by using the HTTPS protocol from editors' phones. What must you do to facilitate this? Choose all that apply and put them in the most appropriate order.

   A. Configure an account on the editors' phones that points to the published URL for their email server.

   B. Deploy a firewall to create an internal network.

   C. Install a reverse web proxy to publish users' mailboxes through HTTPS.

   D. Connect the Lucerne Publishing network to the internet.

2. Lucerne Publishing is working with a movie production company in South Wales. It's important that the two organizations can share data easily between their two networks. What kind of network would this be?

   A. Internet

   B. Intranet

   C. Extranet

3. Authors want to be able to submit manuscripts to Lucerne Publishing editors securely by checking them in and out of a document management system that was recently installed at Lucerne Publishing. What should you install at Lucerne Publishing to facilitate remote access from authors' houses?

   A. An extranet

   B. A VPN server

   C. A firewall

4. Because authors must securely check their manuscripts in and out of the document management system, what should you configure on the authors' laptops?

   A. A firewall

   B. A perimeter network

   C. A VPN client

5. You want to keep the firewall configuration at Lucerne Publishing as simple as possible, allowing for the fewest open external ports. What port supports all the previous requirements?

   A. UDP 500

   B. TCP 443

   C. TCP 80

Test your knowledge

1. **What does the OSI model contain?**

   A. Three layers

   B. Four layers

   C. Five layers

   D. Seven layers

2. **In the OSI model, data is placed on the network medium at which layer?**

   A. Presentation layer

   B. Physical layer

   C. Data-link layer

   D. Application layer

3. **In TCP/IP, which protocol provides reliable end-to-end data transfer at the transport layer?**

   A. ARP

   B. UDP

   C. TCP

   D. IP

Fill in the blanks for the following statements:

4. **In a TCP/IP network, a datagram at the transport layer becomes a ( ) at the network layer.**
5. **A ( ) is an internetwork device that operates at the network layer, forwarding packets between interconnected subnets.**
6. **George wants to make his web server available through his local host firewall. He's configured the web server to use SSL. George should open the ( ) port on his local host firewall to enable others to connect to his web server over SSL.**
7. **True or false: A web proxy and reverse web proxy can often be found in the internal network security zone.**

   True

   False

8. **True or false: Using PAP is preferable to using CHAP when considering authentication protocols for a VPN.**

   True

   False

Study the scenario and answer the questions:

9. **Before going to class, Sidney stops by Fourth Coffee for a drink. John, the owner of Fourth Coffee, knows that Sidney enjoys solving computer problems, so he asks her if she can help him manage Fourth Coffee's server and wireless network. In particular, he wants to access Fourth Coffee's server from his home and mobile phones, and he also wants to help keep his customers' devices safe from viruses and malicious hackers when they use Fourth Coffee's wireless network. Lastly, Josh wants to use his wireless network to securely supply his employees with internal company information, such as schedules and company policies.**

   A. What must Sidney install for Josh to be able to access Fourth Coffee's server from his home and mobile phones?

   B. How can Sidney help secure the Fourth Coffee network for its customers and employees?

Glossary

Term	Definition
Bridge	A device that's running at the data-link layer of the network, which is used to extend a LAN. It interconnects LAN segments and forwards all frames that it receives. In the past, bridges were used to interconnect LANs across distances to create WANs, but a bridge isn't ideally suited to this task.
Extranet	The extension of an organization's network to other organizations
Firewall	A device or software component that filters network traffic based on its characteristics and determines whether to allow or block that traffic
Internet	The public network that individuals and organizations widely use to distribute and share information and to support networked services such as email, databases, web browsing
Intranet	A private network that typically consists of multiple subnets
Local area network (LAN)	A collection of networked devices that are relatively adjacent to one another. Typically, a LAN can span devices within a building, or several buildings in proximity, such as a university campus.
Media access control (MAC) address	A unique 48-bit binary address (usually expressed in hexadecimal) that identifies a network adapter. Typically, the MAC address is the serial number of the network adapter.
Network adapter	A device found in network hosts that connects the host to the network infrastructure via wiring or wireless protocols
Open Systems Interconnection (OSI) model	A reference model that you can use to understand how specific network protocols communicate
Port	Applications on a TCP/IP network use a port to listen for network communications. There are 65,536 ports, and the lower 1,024 are

	well-known and assigned to specific applications. For example, port 443 is used over TCP for web servers that are secured by SSL.
Protocol stack	The protocol stack is responsible for taking messages from applications and packaging and addressing them for transmission to remote hosts. At the remote end, the protocol stack handles passing the received data up the stack to the appropriate application.
Router	An internetwork device that propagates and receives network packets at layer 3 of the OSI reference model. Routers enable network administrators to separate networks into distinct subnets to help manage network traffic. You can also use routers to join remote LANs to create WANs. A router is network transport–specific—that's to say, it runs a specific network protocol, such as TCP/IP.
Switch	A wiring concentrator with advanced software that enables you to change the way frames and packets are handled between devices that are connected to ports on the switch. Layer 2 switches behave like bridges on configured ports. Layer 3 switches emulate router functionality.
Security zone	An area of an organization's network that's bounded by router and firewall devices that control the type of traffic allowed to enter or leave the zone. Typical zones are private and perimeter networks.
Virtual private network (VPN)	A tunnel created by authentication and networking protocols that enables an organization to use the public internet as a transport for private communications.
Wide area networks (WAN)	Uses network devices and protocols to interconnect devices that potentially span the globe.