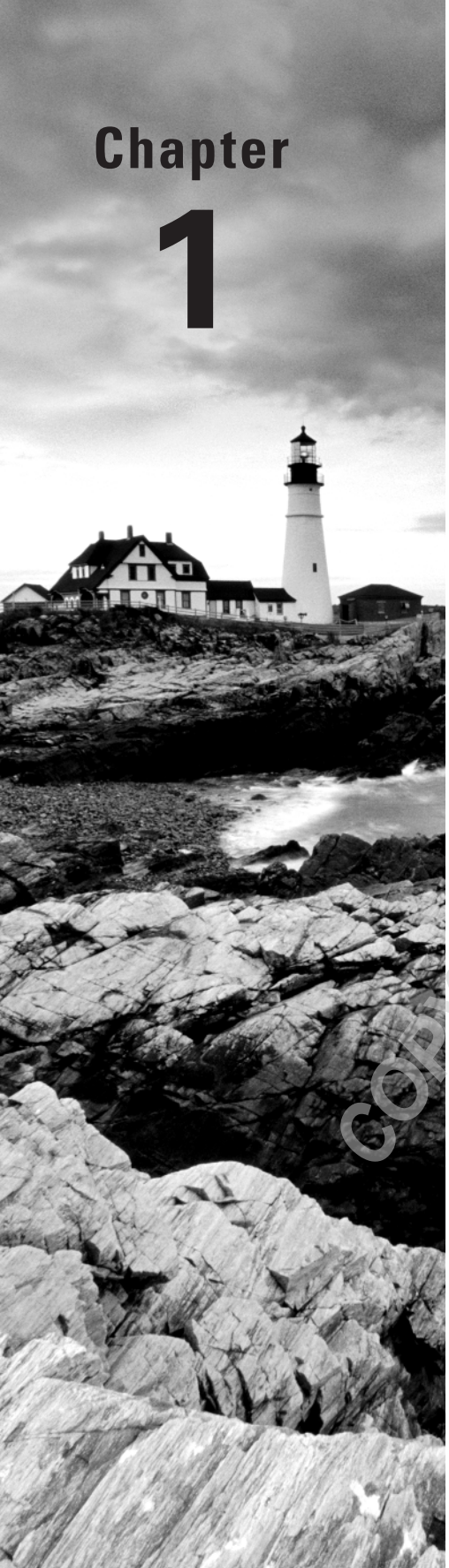


Chapter 1

Introduction to Networks

THE FOLLOWING COMPTIA NETWORK+ EXAM OBJECTIVES ARE COVERED IN THIS CHAPTER:

- ✓ **3.5 Describe different network topologies.**
  - MPLS
    - Point-to-point
    - Point-to-multipoint
  - Ring
  - Star
  - Mesh
  - Bus
  - Peer-to-peer
  - Client-server
  - Hybrid

You'd have to work pretty hard these days to find someone who would argue when we say that our computers have become invaluable to us personally and professionally. Our society has become highly dependent on the resources they offer and on sharing them with each other. The ability to communicate with others—whether they're in the same building or in some faraway land—completely hinges on our capacity to create and maintain solid, dependable networks.

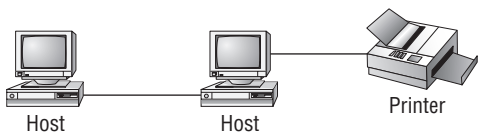
And those vitally important networks come in all shapes and sizes—ranging from small and simple to humongous and super complicated. But whatever their flavor, they all need to be maintained properly, and to do that well, you have to understand networking basics. The various types of devices and technologies that are used to create networks, as well as how they work together, is what this book is about, and I'll go through this critical information one step at a time with you. Understanding all of this will not only equip you with a rock-solid base to build on as you gain IT knowledge and grow in your career, it will also arm you with what you'll need to ace the Network+ certification exam!

To find up-to-the-minute updates for this chapter, please see `www.lammle.com/forum` or the book's web site at `www.sybex.com/go/netplus2e`.

First Things First: What's a Network?

The dictionary defines the word *network* as "a group or system of interconnected people or things." Similarly, in the computer world, the term *network* means two or more connected computers that can share resources such as data and applications, office machines, an Internet connection, or some combination of these, as shown in Figure 1.1.

FIGURE 1.1 A basic network



Okay—Figure 1.1 shows a really basic network made up of only two host computers connected; they share resources such as files and even a printer hooked up to one of the hosts. These two hosts "talk" to each other using a computer language called *binary code*, which consists of lots of 1s and 0s in a specific order that describes exactly what they want to "say."

Next, I'm going to tell you about local area networks (LANs), how they work, and even how we can connect LANs together. Then, later in this chapter, I'll describe how to connect remote LANs together through something known as a wide area network (WAN).

The Local Area Network (LAN)

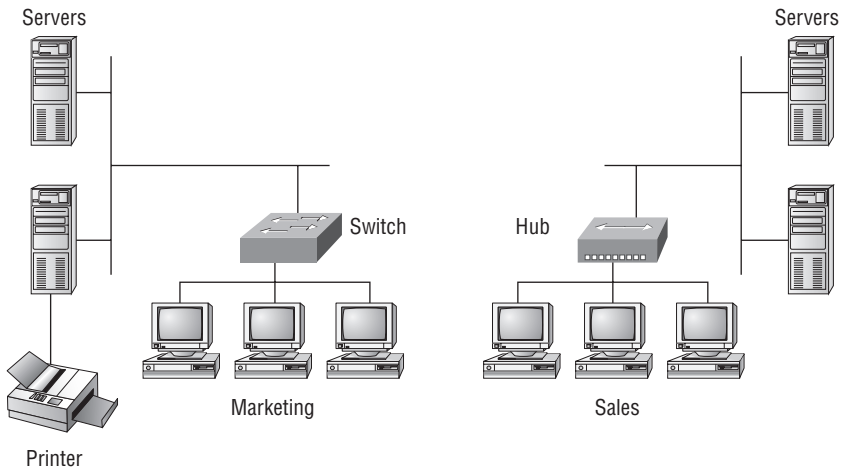
Just as the name implies, a *local area network (LAN)* is usually restricted to spanning a particular geographic location such as an office building, a single department within a corporate office, or even a home office.

Back in the day, you couldn't put more than 30 workstations on a LAN, and you had to cope with strict limitations on how far those machines could actually be from each other. Because of technological advances, all that's changed now, and we're not nearly as restricted in regard to both a LAN's size and the distance a LAN can span. Even so, it's still best to split a big LAN into smaller logical zones known as *workgroups* to make administration easier.

The meaning of the term workgroup in this context is slightly different than when the term is used in contrast to domains. In that context a workgroup is a set of devices with no security association with one another (whereas in a domain they do have that association). In this context, we simply mean they physically are in the same network segment.

In a typical business environment, it's a good idea to arrange your LAN's workgroups along department divisions; for instance, you would create a workgroup for Accounting, another one for Sales, and maybe another for Marketing—you get the idea. Figure 1.2 shows two separate LANS, each as its own workgroup.

FIGURE 1.2 Two separate LANs (workgroups)



First, don't stress about the devices labeled *hub* and *switch*—these are just connectivity devices that allow hosts to physically connect to resources on a LAN. Trust me; I'll describe them to you in much more detail in Chapter 5, "Networking Devices."

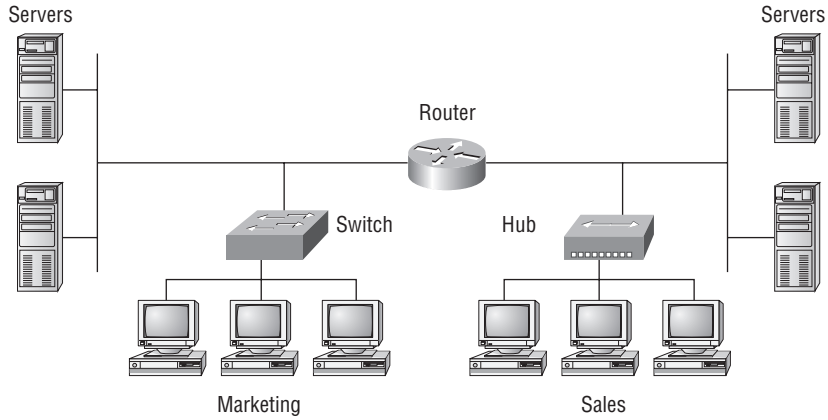
Anyway, back to the figure…. Notice that there's a Marketing workgroup and a Sales workgroup. These are LANs in their most basic form. Any device that connects to the Marketing LAN can access the resources of the Marketing LAN—in this case, the servers and printer. You must be physically connected (via cable or wireless) to a workgroup's LAN to get the resources from it.

There are two problems with this:

- You must be physically connected to a workgroup's LAN to get the resources from it.
- You can't get from one LAN to the other LAN and use its server data and printing resources remotely.

This is a typical network issue that's easily resolved by using a cool device called a *router* to connect the two LANs, as shown in Figure 1.3.

FIGURE 1.3 A router connects LANs.



Nice—problem solved! Even though you can use routers for more than just connecting LANs, the router shown in Figure 1.3 is a great solution because the host computers from the Sales LAN can get to the resources (server data and printers) of the Marketing LAN, and vice versa.

Now, you might be thinking that we really don't need the router—that we could just physically connect the two workgroups with a type of cable that would allow the Marketing and Sales workgroups to hook up somehow. True, we could do that, but if we did, we would have only one big, cumbersome workgroup instead of separate workgroups for Marketing and Sales. And that kind of arrangement isn't practical for today's networks.

This is because with smaller, individual, yet connected groups, the users on each LAN enjoy much faster response times when accessing resources, and administrative tasks are a lot easier, too. Larger workgroups run more slowly because in them, a legion of hosts are all trying to get to the same resources simultaneously. So the router shown in Figure 1.3,

which separates the workgroups while still allowing access between them, is a really great solution after all.

As I said, don't worry about the network connectivity devices I've mentioned so far in this chapter, such as hubs, switches, and routers. I promise to cover them all in detail in Chapter 5. At this stage, I really want you to focus on understanding the concepts that I'm presenting. For now, all you need to know is that hubs and switches are devices that connect other devices together, and routers connect networks together.

So now, let me define those other terms I've used so far: *workstations*, *servers*, and *hosts*.

Common Network Components

There are a lot of different machines, devices, and media that make up our networks. Right now, I'm going to tell you about three of the most common:

- Workstations
- Servers
- Hosts

Workstations

Workstations are often seriously powerful computers that run more than one central processing unit (CPU) and whose resources are available to other users on the network to access when needed. Something this powerful is typically referred to only as a server today. Don't confuse workstations with client machines, which can be workstations but aren't always. A *client machine* is any device on the network that can ask for access to resources from a server or powerful workstation—that for instance, hosts a printer.

The terms *workstation* and *host* can sometimes be used interchangeably. Computers have become more and more powerful and the terms have become somewhat fuzzy because hosts can be clients, workstations, servers, and more! The term *host* is used to describe pretty much anything that takes an IP address.

Servers

Servers are also powerful computers. They get their name because they truly are "at the service" of the network and run specialized software for the network's maintenance and control known as the *network operating system*.

In a good design that optimizes the network's performance, servers are highly specialized and are there to handle one important labor-intensive job. This is not to say that a single

server can't do many jobs, but more often than not, you'll get better performance if you dedicate a server to a single task. Here's a list of common dedicated servers:

File server Stores and dispenses files

Mail server The network's post office; handles email functions

Print server Manages printers on the network

Web server Manages web-based activities by running Hypertext Transfer Protocol (HTTP) for storing web content and accessing web pages

Fax server The "memo maker" that sends and receives paperless faxes over the network

Application server Manages network applications

Telephony server Handles the call center and call routing and can be thought of as a sophisticated network answering machine

Remote access server Provides remote users with access to the network through modems and an IP connection

Proxy server Handles tasks in the place of other machines on the network

See how the name of each kind of server indicates what it actually does—how it serves the network? This is an excellent way to remember them.

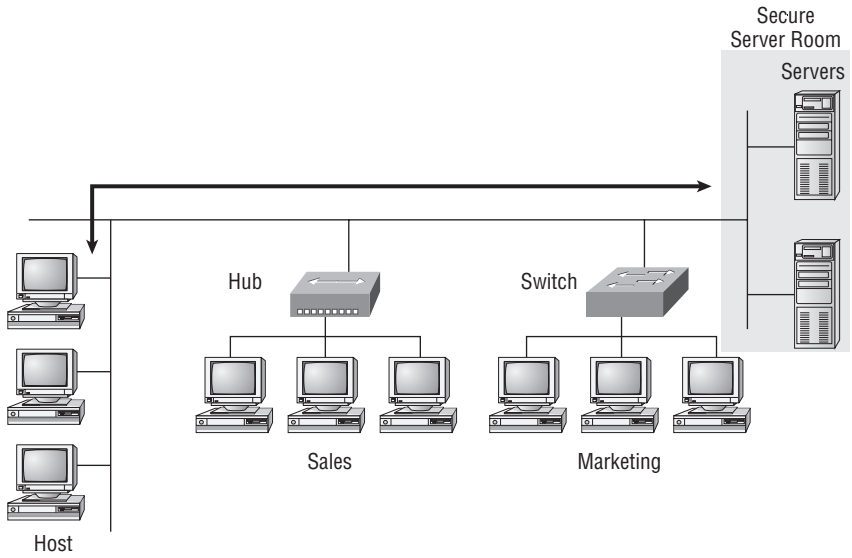
Okay, as I said, servers are usually dedicated to doing one specific important thing within the network. But not always—sometimes they have more than one job. But whether servers are designated for one job or are network multitaskers, they can maintain the network's data integrity by backing up the network's software and hardware. And no matter what, they all serve a number of client machines.

Back in Figure 1.2, I showed you an example of two really simple LAN networks. I want to make sure you know that servers must have considerably superior CPUs, hard-drive space, and memory—a lot more than a simple workstation's capacity—because they serve many client machines and provide any resources they require. Because they're so important, you should always put your servers in a very secure area. My company's servers are in a locked server room because not only are they really pricey workhorses, they also store huge amounts of important and sensitive company data, so, they need to be kept safe from any unauthorized access.

In Figure 1.4, you can see a network populated with both workstations and servers. You also see that the hosts can access the servers across the network—pretty much the general idea of having a network.

You probably noticed that there are more workstations here than servers, right? Think of why that is…. If you answered that it's because one server can provide resources to what can sometimes be a huge number of individual users at the same time, but workstations don't, you have it!

FIGURE 1.4 A network populated with servers and workstations



Hosts

It can be kind of confusing because when people refer to hosts, they really can be referring to almost any type of networking devices—including workstations and servers. But if you dig a bit deeper, you'll find that usually this term comes up when people are talking about resources and jobs that have to do with Transmission Control Protocol/Internet Protocol (TCP/IP). The scope of possible machines and devices is so broad because, in TCP/IP-speak, *host* means any network device with an IP address. Yes, you'll hear IT professionals throw this term around pretty loosely; for the Network+ exam, stick to the definition being network devices, including workstations and servers, with IP addresses.

Here's a bit of background: The name *host* harkens back to the Jurassic period of networking when those dinosaurs known as *mainframes* were the only intelligent devices to roam the network. These were called *hosts* whether they had TCP/IP functionality or not. In that bygone age, everything else in the network-scape was referred to as *dumb terminals* because only mainframes—hosts—were given IP addresses. Another fossilized term from way back then is *gateways* when it was used to talk about any Layer 3 machines like routers. We still use these terms today, but they've evolved a bit to refer to the many intelligent devices populating our present-day networks, each of which has an IP address. This is exactly the reason why you hear *host* used so broadly.

Wide Area Network (WAN)

There are legions of people who, if asked to define a *wide area network (WAN)*, couldn't do it. Yet most of them use the Big Dog of all WANs—the Internet—every day! With that

in mind, you can imagine that WAN networks are what we use to span large geographic areas and truly go the distance. Like the Internet, WANs usually employ both routers and public links, so that's generally the criteria used to define them.

WANs are so important that I have dedicated an entire chapter to them: Chapter 16, "Wide Area Networks."

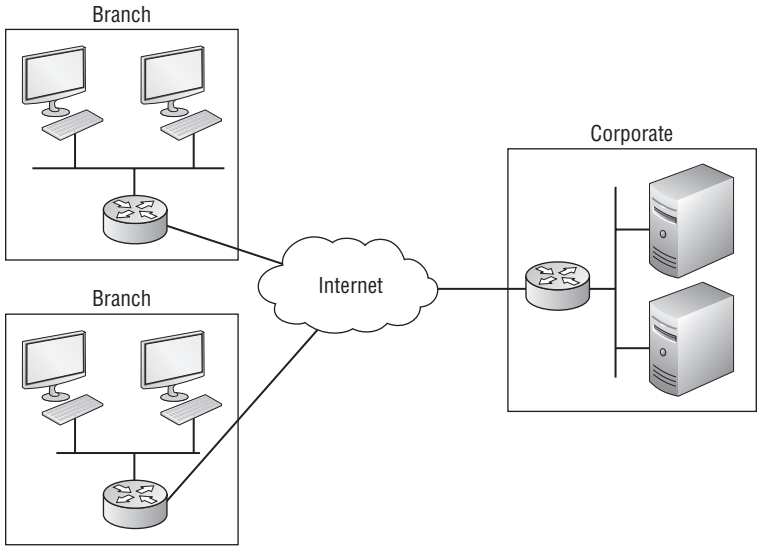
Here's a list of some of the important ways that WANs are different from LANs:

- WANs usually need a router port or ports.
- WANs span larger geographic areas and/or can link disparate locations.
- WANs are usually slower.
- We can choose when and how long we connect to a WAN. A LAN is all or nothing—our workstation is either connected permanently to it or not at all, although most of us have dedicated WAN links now.
- WANs can utilize either private or public data transport media such as phone lines.

We get the word *Internet* from the term *internetwork*. An internetwork is a type of LAN and/or WAN that connects a bunch of networks, or *intranets*. In an internetwork, hosts still use hardware addresses to communicate with other hosts on the LAN. However, in an internetwork, hosts use logical addresses (IP addresses) to communicate with hosts on a different LAN (other side of the router).

And *routers* are the devices that make this possible. Each connection into a router is a different logical network. Figure 1.5 demonstrates how routers are employed to create an internetwork and how they enable our LANs to access WAN resources.

FIGURE 1.5 An internetwork



The Internet is a prime example of what's known as a *distributed WAN*—an internetwork that's made up of a lot of interconnected computers located in a lot of different places. There's another kind of WAN, referred to as *centralized*, that's composed of a main, centrally located computer or location that remote computers and devices can connect to. A good example is remote offices that connect to a main corporate office, as shown in Figure 1.5.

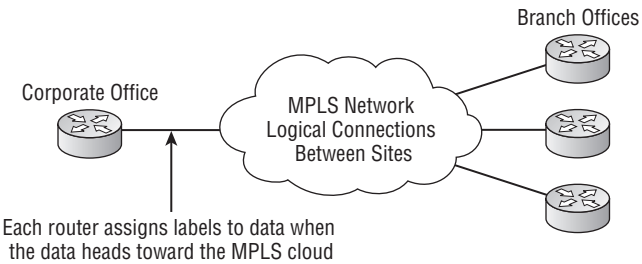
MPLS

MultiProtocol Label Switching (MPLS) will be defined clearly in Chapter 16, but for the objectives of the CompTIA Network+ exam, this chapter will define the actual layout of what is one of the most popular WAN protocols in use today. MPLS has become one of the most innovative and flexible networking technologies on the market and has several advantages compared to other WAN technologies:

- Physical layout flexibility
- Prioritizing of data
- Redundancy in case of link failure
- One-to-many connection

MPLS is a switching mechanism that imposes labels (numbers) to data and then uses those labels to forward data when the data arrives at the MPLS network, as shown in Figure 1.6.

FIGURE 1.6 MultiProtocol Label Switching layout



The labels are assigned on the edge of the MPLS network, and forwarding inside the MPLS network (cloud) is done solely based on labels through virtual links instead of physical links. Prioritizing data is a huge advantage, and voice data could have priority over basic data based on the labels, for example. And since there are multiple paths for the data to be forwarded through the MPLS cloud, there is some redundancy provided.

Network Architecture: Peer-to-Peer or Client-Server?

So, we've developed networking as a way to share resources and information, and how that's achieved directly maps to the particular architecture of the network operating system software. There are two main network types you need to know about: peer-to-peer

and client-server. And by the way, it's really tough to tell the differences just by looking at a diagram or even by checking out live video of the network humming along, but the differences between peer-to-peer and client-server architectures are major. They're not just physical; they're logical differences. You'll see what I mean in a bit.

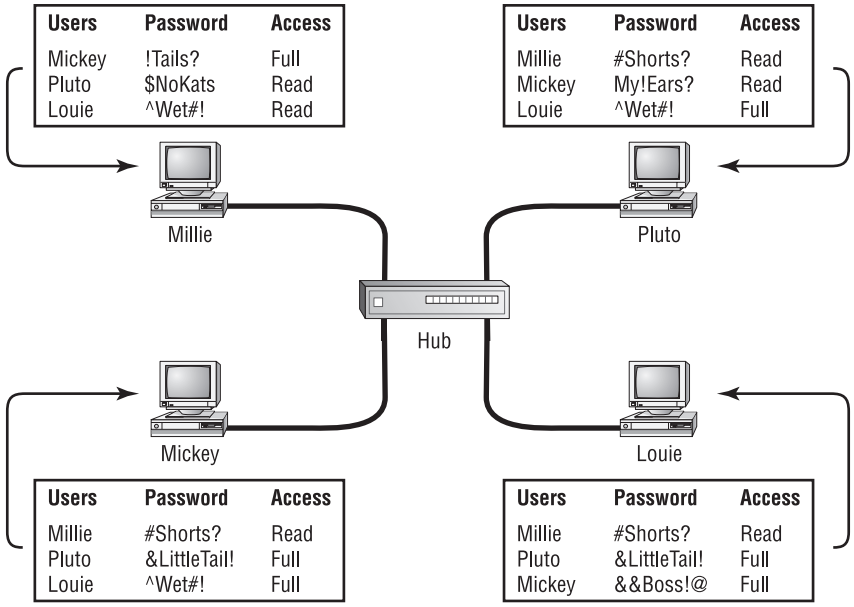
Peer-to-Peer Networks

Computers connected together in *peer-to-peer networks* do not have any central, or special, authority—they're all *peers*, meaning that when it comes to authority, they're all equals. This means it's up to the computer that has the resource being requested to perform a security check for access rights to its resources.

It also means that the computers existing in a peer-to-peer network can be client machines that access resources and server machines that provide them to other computers. This works really well if there's not a huge number of users on the network, if each user backs things up locally, and if your network doesn't require a lot of security.

If your network is running Windows, Mac, or Unix in a local LAN workgroup, you have a peer-to-peer network. Figure 1.7 gives you a snapshot of a typical peer-to-peer network. Peer-to-peer networks present some challenges. For example, backing up company data becomes an iffy proposition.

FIGURE 1.7 A peer-to-peer network



Since it should be clear by now that peer-to-peer networks are all sunshine, backing up all that super-important data is not only vital, it can be really challenging. What if you forget

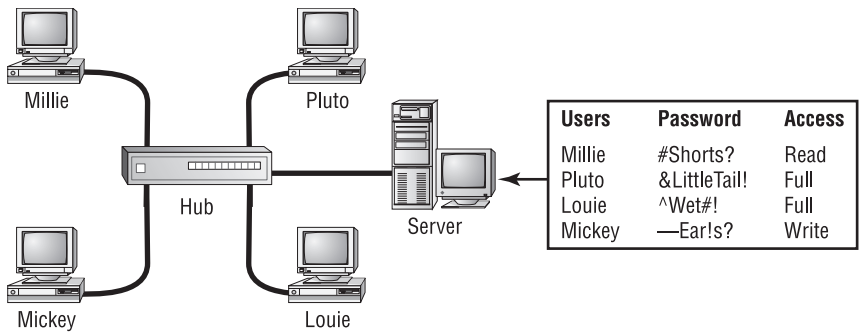
where you put a badly needed file? (Haven't we all done that?) And then there's that security issue to tangle with. Because security is not centrally governed, each and every user has to remember and maintain a list of users and passwords on each and every machine. Worse, some of those all-important passwords for the same users change on different machines—even for accessing different resources. Yikes!

Client-Server Networks

Client-server networks are pretty much the polar opposite of peer-to-peer networks because in them, a single server uses a network operating system for managing the whole network. So a client machine's request for a resource goes to the main server, which responds by handling security and directing the client to the resource it wants, instead of the request going directly to the machine with the desired resource. This arrangement definitely has its benefits. First, because the network is much better organized and doesn't depend on users remembering where needed resources are, it's a whole lot easier to find the files you need because everything is stored in one spot—on that special server. Your security also gets a lot tighter because all usernames and passwords are on that server (which, by the way, isn't ever used as a workstation). You even gain scalability—client-server networks can have legions of workstations on them. And even with all those demands, their performance is actually optimized.

Check out Figure 1.8, which shows a client-server network with a server that has a database of access rights, user accounts, and passwords.

FIGURE 1.8 A client-server network



Many of today's networks are a healthy (we hope) combination of the peer-to-peer and client-server architectures with carefully specified servers that permit the simultaneous sharing of resources from devices running workstation operating systems. Even though the supporting machines can't handle as many inbound connections at a time, they still run the server service reasonably well. If this type of mixed environment is designed well, most networks benefit greatly by having the capacity to take advantage of the positive aspects of both worlds.

Physical Network Topologies

Just as a topographical map is a type of map that shows the shape of the terrain, the *physical topology* of a network is also a type of map. It defines the specific characteristics of a network, such as where all the workstations and other devices are located and the precise arrangement of all the physical media such as cables. On the other hand, *logical topologies*, which were covered earlier, delineate exactly how data moves through the network. And though these two topologies are usually a lot alike, a particular network can have physical and logical topologies that are very different. But basically, what you want to remember is that a network's physical topology gives you the lay of the land and the logical topology shows how digital signal or data navigates through that layout.

The following are the topologies you're most likely to run in to these days:

- Bus
- Star
- Ring
- Mesh
- Point-to-point
- Point-to-multipoint
- Hybrid

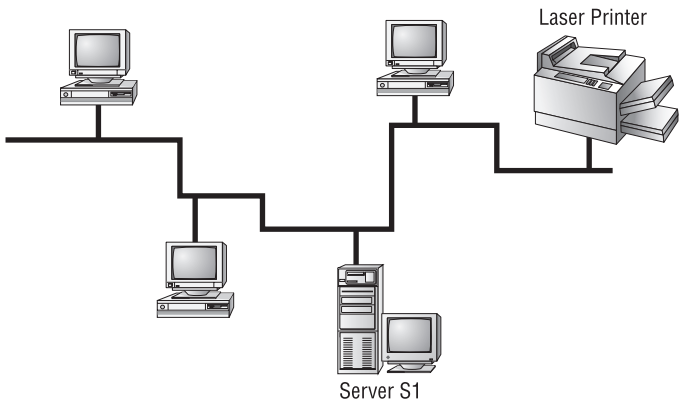
Bus Topology

This type of topology is the most basic one of the bunch, and it really does sort of resemble a bus. (Well, okay—actually, it looks more like a bus that's been in a pretty nasty wreck!) Anyway, the *bus topology* consists of two distinct and terminated ends, with each of its computers connecting to one unbroken cable running its entire length. Back in the day, we used to attach computers to that main cable with wire taps, but this didn't work all that well so we began using drop cables in their place (unless you're dealing with 10Base2 Ethernet, in which case you would slip a "T" into the main cable anywhere you wanted to connect a device to it instead of using drop cables).

Figure 1.9 depicts what a typical bus network's physical topology looks like.

Even though all the computers on this kind of network see all the data flowing through the cable, only the one computer that the data is specifically addressed to actually *gets* the data. Some of the benefits in favor of using a bus topology are that it's easy to install and it's not very expensive, in part because it doesn't require as much cable as the other types of physical topologies. But it also has some drawbacks: For instance, it's hard to troubleshoot, change, or move, and it really doesn't offer much in the way of fault tolerance because everything is connected to that single cable.

FIGURE 1.9 A typical bus network's physical topology



By the way, *fault tolerance* is the capability of a computer or a network system to respond to a condition automatically, often resolving it, which reduces the impact on the system. If fault-tolerance measures have been implemented correctly on a network, it's highly unlikely that any of that network's users will know that a problem even existed.

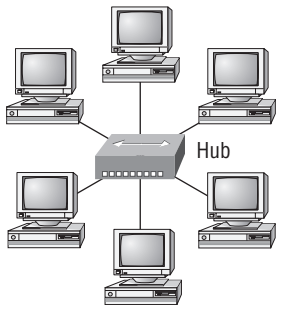
Star Topology

A star topology's computers are connected to a central point with their own individual cables or wireless connections. You'll often find that central spot inhabited by a device like a hub, a switch, or an access point.

Star topology offers a lot of advantages over bus topology, making it more widely used even though it obviously requires more physical media. One of its best features is that because each computer or network segment is connected to the central device individually, if the cable fails, it brings down only that particular machine or network segment. That's truly a great benefit because it makes the network much more fault tolerant as well as a lot easier to troubleshoot. Another great thing about a star topology is that it's a lot more scalable—all you have to do if you want to add to it is run a new cable and connect to the machine at the core of the star. In Figure 1.10, you'll find a great example of a typical star topology.

Okay, although it is called a *star* topology, it really looks a lot more like the imaginary pictures people draw of the sun. (Yes, the sun is a star—but it definitely doesn't look like how we usually depict it, does it?) You could also get away with saying it looks like a bike wheel with spokes connecting to the hub in the middle of the wheel and extending outward to connect to the rim. And just like that bike wheel, it's the hub device at the center of a star topology network that can give you the most grief if something goes wrong with it. If that hub in the middle of it all happens to fail, down comes the whole network, so it's a very good thing hubs don't fail often!

FIGURE 1.10 Typical star topology with a hub



Just as it is with pretty much everything, a star topology has its pros and cons. But the good news far outweighs the bad, which is why people are choosing to go with a star topology more and more. Here's a list of benefits gained by opting for a star topology:

- New stations can be added easily and quickly.
- A single cable failure won't bring down the entire network.
- It is relatively easy to troubleshoot.

Here are the disadvantages of a star topology:

- The total installation cost can be higher because of the larger number of cables (but prices are constantly becoming more competitive).
- It has a single point of failure (the hub or other central device).

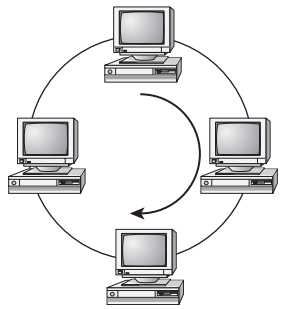
There are two more sophisticated implementations of a star topology. The first is called *point-to-point link*, where you have not only the device in the center of the spoke acting as a hub, but also the one on the other end. This is still a star-wired topology, but as I'm sure you can imagine, it gives you a huge amount of scalability!

Another refined version is the wireless flavor, but to understand this version well, you've really got to have a solid grasp of the capabilities and features of all the devices populating the wireless star topology. No worries, though—I'll be covering wireless access points later on in Chapter 12, "Wireless Networking." For now, it's good enough for you to know that access points are pretty much just wireless hubs or switches that behave like their wired counterparts. Basically, they create a point-by-point connection to endpoints and other wireless access points.

Ring Topology

In this type of topology, you'll find that each computer is directly connected to other computers within the same network. Looking at Figure 1.11, you can see that the network's data flows from computer to computer back to the source, with the network's primary cable forming a ring. The problem is, the *ring topology* has a lot in common with the bus topology because if you want to add to the network, you have no choice but to break the cable ring—something that is probably going to bring down the entire network.

FIGURE 1.11 A typical ring topology



This is one big reason this topology isn't all that popular—you just won't run into it a lot as I did in the 1980s and early 1990s. It's also pricey because you need several cables to connect each computer, it's really hard to reconfigure, and as you've probably guessed, it's not fault tolerant.

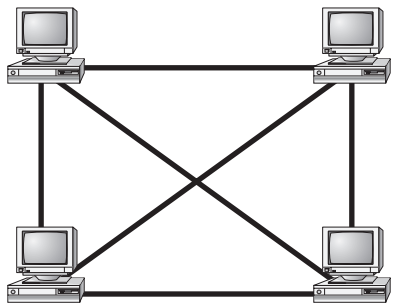
However, with all that being said, if you work at an ISP, you may find a physical ring topology in use for a technology called SONET or possibly some other WAN technology. You just won't find any LANs in physical rings anymore.

Mesh Topology

In this type of topology, you'll find that there's a path from every machine to every other one in the network. That's a lot of connections—in fact, the *mesh topology* wins the prize for "most physical connections per device"! You won't find it used in LANs very often, if ever, these days, but you will find a modified version of it known as a *hybrid mesh* used in a restrained manner on WANs, including the Internet.

Often, hybrid mesh topology networks will have quite a few connections between certain places to create redundancy (backup). And other types of topologies can sometimes be found in the mix too, which is also why it's dubbed *hybrid*. At any rate, it isn't a full-on full mesh topology if there isn't a connection between all devices in the network. But it's still respectably complicated—Figure 1.12 shows just how much only four connections can complicate things.

FIGURE 1.12 A typical mesh topology



As shown in the figure, things just get more and more complex as both the wiring and the connections multiply. For each *n* locations or hosts, you end up with *n*(*n*–1)/2 connections. This means that in a network consisting of only four computers, you have 4(4–1)/2, or 6 connections. And if that little network grows to, say, a population of 10 computers, you'll have a whopping 45 connections to cope with—yikes! That's a huge amount of overhead, so only small networks can really use this topology and manage it well. On the bright side, you get a very respectable level of fault tolerance. But it is nice that we don't use these in corporate LANs any longer because they were very complicated to manage.

A full mesh physical topology has the absolute least likelihood of having a collision.

This is the reason you will usually find the hybrid version in today's WANs. In fact, the mesh topology is actually pretty rare these days. It's mainly used because of the robust fault tolerance it offers—because you have a multitude of connections, if one goes on the blink, computers and other network devices can simply switch to one of the many redundant connections that are up and running. But as you can imagine, all that cabling in the mesh topology makes it really costly. Plus, you can make your network management much less insane by using what's known as a *partial mesh topology* solution instead, so why not go that way? You may lose a little fault tolerance, but if you go the partial mesh route, you still get to use the same technology between all the network's devices. Just remember that with partial mesh, not all devices will be interconnected, so it's very important to choose wisely the ones that are.

Point-to-Point Topology

As its name implies, in a *point-to-point* topology you have a direct connection between two routers, giving you one communication path. The routers in a point-to-point topology can either be linked by a serial cable, making it a physical network, or be far apart and connected only by a circuit within a Frame Relay or MPLS network, making it a logical network.

Figure 1.13 gives you a prime specimen of a T1, or WAN point-to-point connection.

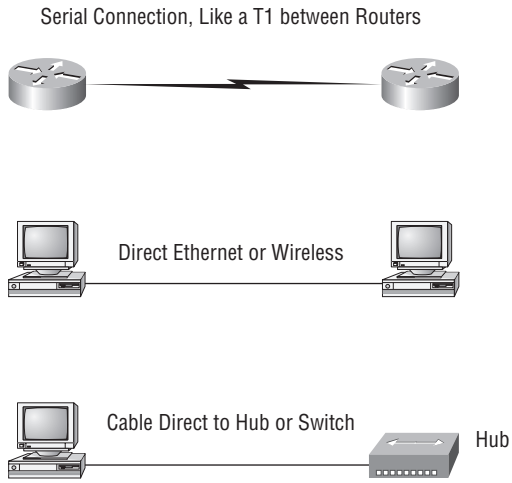
What you see here is a lightning bolt and a couple of round things with a bunch of arrows projecting from them, right? Well, the two round things radiating arrows represent our network's two routers, and that lightning bolt represents a WAN link. (These symbols are industry standard, and I'll be using them throughout this book, so it would a good idea to get used to them.)

Part two of the diagram shows two computers connected by a cable—a point-to-point link. By the way, this should remind you of something we just went over…remember our talk about peer-to-peer networks? Good! I hope you also happen to remember that a big drawback related to peer-to-peer network sharing is that it is not very scalable. With this in mind, you probably won't be all that surprised that even if both machines have a wireless point-to-point connection, the network won't be very scalable.

You'll usually find point-to-point networks within many of today's WANs, and as you can see in part three of Figure 1.13, a link from a computer to a hub or switch is also a valid point-to-point connection. A common version of this setup consists of a direct

wireless link between two wireless bridges that's used to connect computers in two different buildings together.

FIGURE 1.13 Three point-to-point connections



Point-to-Multipoint Topology

Again as the name suggests, a *point-to-multipoint* topology consists of a succession of connections between an interface on one router and multiple destination routers—one point of connection to multiple points of connection. Each of the routers and every one of their interfaces involved in the point-to-multipoint connection are part of the same network.

Figure 1.14 shows a WAN to best demonstrate a point-to-multipoint network that depicts a single corporate router connecting to multiple branches.

Figure 1.15 shows another prime example of a point-to-multipoint network: a college or corporate campus.

FIGURE 1.14 A point-to-multipoint network, example 1

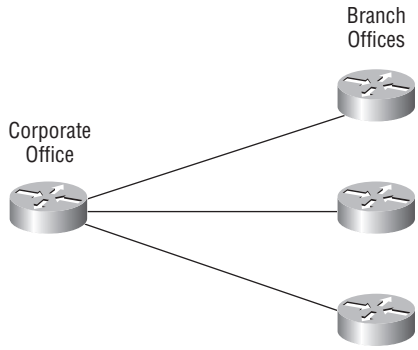
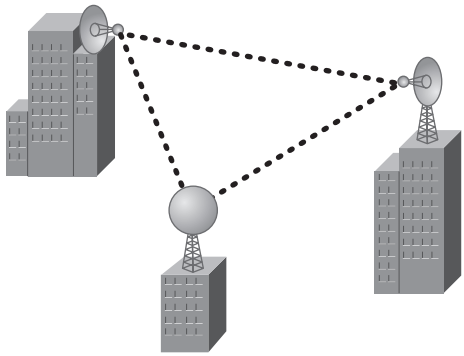


FIGURE 1.15 A point-to-multipoint network, example 2

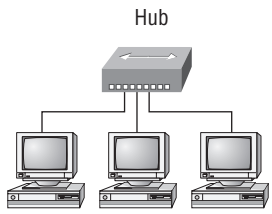


Hybrid Topology

I know I talked about hybrid network topology back in the section about mesh topology, but I didn't give you a picture of it in the form of a figure. I also want to point out that *hybrid topology* means just that—a combination of two or more types of physical or logical network topologies working together within the same network.

Figure 1.16 depicts a simple hybrid network topology; it shows a LAN switch or hub in a star topology configuration that connects to its hosts via bus topology.

FIGURE 1.16 A simple hybrid network



Topology Selection, Backbones, and Segments

Now that you're familiar with many different types of network topologies, you're ready for some tips on selecting the right one for your particular network. You also need to know about backbones and segments—the very last part of this chapter.

Real World Scenario

They're Just Cables, Right?

Wrong! Regardless of the type of network you build, you need to start thinking about quality at the bottom and work up.

Think of it as if you were at an electronics store buying the cables for your sweet new home theater system. You've already spent a bunch of time and money getting the right components to meet your needs. In fact, you've probably parted with a respectable chunk of change, so why would you stop there and connect all these great devices together with the cable equivalent of twine? No, you're smarter than that. You know that picking out the exact cables that will maximize the sound and picture quality of your specific components can also protect them.

It's the same thing when you're faced with selecting the physical media for a certain network (such as your new client-server network). You just don't want to cut corners here. Because if it's the backbone of the network, you absolutely don't want to be faced with having to dig up everything that's already been installed after the fact. Doing this costs a lot more than taking the time to wisely choose the right cables and spending the money it takes to get them in the first place. The network downtime alone can cost a company a bundle (pun intended). Another reason for choosing the network's physical media correctly is that it's going to be there for a good 5 to 10 years. This means two things: It better be solid quality, and it better be scalable because that network is going to grow and change over the years.

Selecting the Right Topology

As you now know, not only do you have a buffet of network topologies to choose from, but each one also has pros and cons to implementing it. But it really comes down to that well-known adage "Ask the right questions." First, how much cash do you have? And how much fault tolerance do you really need? Also, is this network likely to grow like a weed—is it probably going to need to be quickly and easily reconfigured often? In other words, how scalable does your network need to be?

For instance, if your challenge is to design a nice, cost-effective solution that involves only a few computers in a room, getting a wireless access point and some wireless network cards is definitely your best way to go because you won't need to pony up for a bunch of cabling and it's super simple to set up. Alternatively, if you're faced with coming up with a solid design for a growing company's already-large network, you're probably good to go with using a wired star topology because it will nicely allow for future changes. Remember, a star topology really shines when it comes to making additions to the network, moving things around, and making any kind of changes happen quickly, efficiently, and cost effectively.

If, say, you're hired to design a network for an ISP that needs to be up and running 99.9 percent of the time with no more than 8 hours a year allowed downtime, well, you need Godzilla-strength fault tolerance. Do you remember which topology gives that up the best? (Hint: Internet.) Your primo solution is to go with either a hybrid or a partial mesh topology. Remember that partial mesh leaves you with a subset of $n(n–1)/2$ connections to maintain—a number that could very well blow a big hole in your maintenance budget!

Here's a list of things to keep in mind when you're faced with coming up with the right topology for the right network:

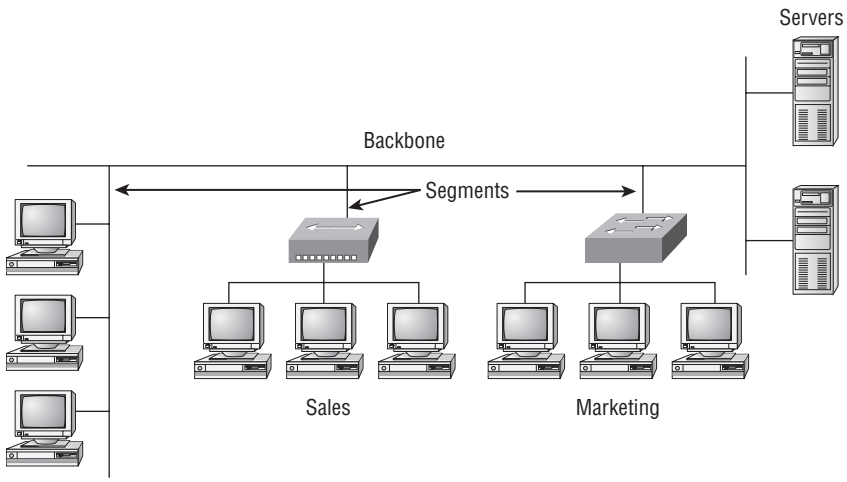
- Cost
- Ease of installation
- Ease of maintenance
- Fault-tolerance requirement

The Network Backbone

Today's networks can get pretty complicated, so we need to have a standard way of communicating with each other intelligibly about exactly which part of the network we're referencing. This is the reason we divide networks into different parts called *backbones* and *segments*.

Figure 1.17 illustrates a network and shows which part is the backbone and which parts are segments.

FIGURE 1.17 Backbone and segments on a network



You can see that the network backbone is actually kind of like our own. It's what all the networks segments and servers connect to and what gives the network its structure. As you

can imagine, being such an important nerve center, the backbone must use some kind of seriously fast, robust technology—often that's Gigabit Ethernet. And to optimize network performance (that is, speed and efficiency), it follows that you would want to connect all of the network's servers and segments directly to the network's backbone.

Network Segments

When we refer to a segment, we can mean any small section of the network that may be connected to, but isn't actually a piece of, the backbone. The network's workstations and servers organized into segments connect to the network backbone, which is the common connecting point for all segments; you can see this by taking another look at Figure 1.17, which displays three segments.

Summary

This chapter created a solid foundation for you to build your networking knowledge on as you go through this book.

In it, you learned what, exactly, a network is, and you got an introduction to some of the components involved in building one—routers, switches, and hubs—as well as the jobs they do in a network.

You also learned that having the components required to build a network isn't all you need. Understanding the various types of network connection methods like peer-to-peer and client-server is also vital.

Further, you learned about the various types of logical and physical network topologies and the features and drawbacks of each. I wrapped up the chapter with a short discussion about network backbones and segments and equipped you with the right questions to ask yourself to ensure that you come up with the right network topology for your networking needs.

Exam Essentials

Know your network topologies. Know the names and descriptions of the topologies. Be aware of the difference between physical networks (what humans see) and logical networks (what the equipment "sees").

Know the advantages and disadvantages of the topologies. It is important to know what each topology brings to the table. Knowing the various characteristics of each topology comes in handy during troubleshooting.

Understand the terms *LAN* and *WAN*. You need to understand when you would use a LAN and when you would use a WAN. A LAN is used to connect a group of hosts together, and a WAN is used to connect various LANs together.

Written Labs

Provide the answers to the following questions:

1. What are the three primary LAN topologies?
2. What common WAN topology often results in multiple connections to a single site (leading to a high degree of fault tolerance) and has one-to-many connections?
3. What is the term for a device that shares its resources with other network devices?
4. What network model draws a clear distinction between devices that share their resources and devices that do not?
5. Which network topology or connection type can be implemented with only two endpoints?
6. What device is an example of an Ethernet technology implemented as a star topology?
7. What does MPLS stand for?
8. What does WAN stand for?
9. Will a computer that shares no resources most likely be connected to the backbone or to a segment?
10. Which LAN topology is characterized by all devices being daisy-chained together with the devices at each end being connected to only one other device?

You can find the answers to the written labs in Appendix B.

Review Questions

You can find the answers in Appendix A.

1. You need a network that provides centralized authentication for your users. Which of the following logical topologies should you use?
   - **A.** VLANs
   - **B.** Peer-to-peer
   - **C.** Client-server
   - **D.** Mesh
2. You need a topology that is scalable to use in your network. Which of the following will you install?
   - **A.** Bus
   - **B.** Ring
   - **C.** Star
   - **D.** Mesh
3. Which of the following physical topologies has the most connections and is the least popular for LANs?
   - **A.** Bus
   - **B.** Start
   - **C.** Ring
   - **D.** Mesh
4. In a physical star topology, what happens when a workstation loses its physical connection to another device?
   - **A.** The ring is broken, so no devices can communicate.
   - **B.** Only that workstation loses its ability to communicate.
   - **C.** That workstation and the device it's connected to lose communication with the rest of the network.
   - **D.** No devices can communicate because there are now two unterminated network segments.

5. Which type of WAN technology uses labels, which enables priority of voice though the network?
   - **A.** VPN
   - **B.** T1
   - **C.** MPLS
   - **D.** LAN
   - **E.** Bus
6. What is a logical grouping of network users and resources called?
   - **A.** WAN
   - **B.** LAN
   - **C.** MPLS
   - **D.** Host
7. Which of the following is a concern when using peer-to-peer networks?
   - **A.** Where to place the server
   - **B.** Whose computer is least busy and can act as the server
   - **C.** The security associated with such a network
   - **D.** Having enough peers to support creating such a network
8. Which of the following is an example of when a point-to-multipoint network is called for?
   - **A.** When a centralized office needs to communicate with many branch offices
   - **B.** When a full mesh of WAN links is in place
   - **C.** When multiple offices are daisy-chained to one another in a line
   - **D.** When there are only two nodes in the network to be connected
9. Which of the following is an example of a LAN?
   - **A.** Ten buildings interconnected by Ethernet connections over fiber-optic cabling
   - **B.** Ten routers interconnected by Frame Relay circuits
   - **C.** Two routers interconnected with a T1 circuit
   - **D.** A computer connected to another computer so they can share resources
10. Which of the following is a disadvantage of the star topology?
    - **A.** When a port on the central concentrating device fails, the attached end device and entire network loses connectivity to the rest of the network.
    - **B.** When the central concentrating device experiences a complete failure, all attached devices lose connectivity to the rest of the network.
    - **C.** In a star topology, a more expensive type of host must be used compared to the host used when implementing a physical bus.
    - **D.** It is more difficult to add stations and troubleshoot than with other topologies.

11. What is a difference between a LAN and a WAN?

A. WANs need a special type of router port.

B. WANs cover larger geographical areas.

C. WANs can utilize either private or public data transport.

D. All of the above.

12. Which of the following provides the most physical layout flexibility in a very large, geographically dispersed enterprise network?

A. Bus topology

B. LAN switch

C. Star topology

D. MPLS cloud network

13. In what type of network are all computers considered equals and they do not share any central authority?

A. Peer-to-peer

B. Client-server

C. Physical topology

D. None of the above

14. What advantage does the client-server architecture have over peer-to-peer?

A. Easier maintenance

B. Greater organization

C. Tighter security

D. All of the above

15. Which of the following is an example of a hybrid network?

A. Ethernet switch

B. Ring topology

C. Bus topology

D. Star topology

16. You have a network with multiple devices and need to have a smaller broadcast domain while working with a single device. Which of the following is the best solution?

A. Use static IP addresses.

B. Add more hubs.

C. Implement more switches.

D. Install a router.

17. Which type of topology has the greatest number of physical connections?

- **A.** Point-to-multipoint
- **B.** Star
- **C.** Point-to-point
- **D.** Mesh

18. What type of topology gives you a direct connection between two routers so that there is one communication path?

- **A.** Point-to-point
- **B.** Star
- **C.** Bus
- **D.** Straight

19. Which network topology is a combination of two or more types of physical or two or more types of logical topologies?

- **A.** Point-to-multipoint
- **B.** Hybrid
- **C.** Bus
- **D.** Star

20. When designing a network and deciding which type of network topology to use, which item(s) should be considered? (Select all that apply.)

- **A.** Cost
- **B.** Ease of installation
- **C.** Ease of maintenance
- **D.** Fault-tolerance requirements