Because security requires control over who can access specific resources, more intelligent devices can provide a higher level of protection because they can make more detail-oriented decisions regarding who can access resources. When devices can look deeper into the packets, they have access to more information to make access decisions, which provides more granular access control.

As previously stated, switching makes it more difficult for intruders to sniff and monitor network traffic because no broadcast and collision information is continually traveling throughout the network. Switches provide a security service that other devices cannot provide. VLANs (described in depth in Chapter 13) are an important part of switching networks, because they enable administrators to have more control over their environment and they can isolate users and groups into logical and manageable entities.

Routers

We are going up the chain of the OSI layers while discussing various network devices. Repeaters work at the physical layer, bridges and switches work at the data link layer, and routers work at the network layer. As we go up each layer, each corresponding device has more intelligence and functionality because it can look deeper into the frame. A repeater looks at the electrical signal. The switch can look at the MAC address within the header. The router can peel back the first header information and look farther into the frame and find out the IP address and other routing information. The farther a device can look into a frame, the more decisions it can make based on the information within the frame.

Routers are layer 3, or network layer, devices that are used to connect similar or different networks. (For example, they can connect two Ethernet LANs or an Ethernet LAN to a Frame Relay link.) A router is a device that has two or more interfaces and a routing table, so it knows how to get packets to their destinations. It can filter traffic based on an access control list (ACL), and it fragments packets when necessary. Because routers have more network-level knowledge, they can perform higher-level functions, such as calculating the shortest and most economical path between the sending and receiving hosts.

A router discovers information about routes and changes that take place in a network through its routing protocols (RIP, BGP, OSPF, and others, as discussed in Chapter 11). These protocols tell routers if a link has gone down, if a route is congested, and if another route is more economical. They also update routing tables and indicate if a router is having problems or has gone down.

The router may be a dedicated appliance or a computer running a networking operating system that is dual-homed. When packets arrive at one of the interfaces, the router compares those packets to its ACL. This list indicates what packets are allowed in and what packets are denied. Access decisions are based on source and destination IP addresses, protocol type, and source and destination ports. An administrator may block all packets coming from the 10.10.12.0 network, any FTP requests, or any packets headed toward a specific port on a specific host, for example. This type of control is provided by the ACL, which the administrator must program and update as necessary.

What actually happens inside the router when it receives a packet? Let's follow the steps:

1. A packet is received on one of the interfaces of a router. The router views the routing data.
2. The router retrieves the destination IP network address from the packet.
3. The router looks at its routing table to see which port matches the requested destination IP network address.
4. If the router does not have information in its table about the destination address, it sends out an ICMP error message to the sending computer indicating that the message could not reach its destination.
5. If the router does have a route in its routing table for this destination, it decrements the TTL value and sees whether the maximum transmission unit (MTU) is different for the destination network. If the destination network requires a smaller MTU, the router fragments the packet.
6. The router changes header information in the packet so that the packet can go to the next correct router, or if the destination computer is on a connecting network, the changes made enable the packet to go directly to the destination computer.
7. The router sends the packet to its output queue for the necessary interface.

Table 14-3 provides a quick review of how routers differ from bridges and switches.

When is it best to use a repeater, bridge, or router? A repeater is used if an administrator needs to expand a network and amplify signals so they do not weaken on longer cables. However, a repeater also extends collision and broadcast domains.

Bridges and switches work at the data link layer and have a bit more intelligence than a repeater. Bridges can do simple filtering and separate collision domains, but not broadcast domains. A switch should be used when an administrator wants to connect multiple computers in a way that reduces traffic congestion and excessive collisions.

A router splits up a network into collision domains and broadcast domains. A router gives more of a clear-cut division between network segments than repeaters or bridges.

Bridge/Switch	Router
Reads header information but does not alter it	Creates a new header for each packet
Builds forwarding tables based on MAC addresses	Builds routing tables based on IP addresses
Has no concept of network addresses	Assigns a different network address per port
Filters traffic based on MAC addresses	Filters traffic based on IP addresses
Forwards broadcast traffic	Does not forward broadcast traffic
Forwards traffic if a destination address is unknown to the bridge	Does not forward traffic that contains a destination address unknown to the router

Table 14-3 Main Differences Between Bridges/Switches and Routers

A router should be used if an administrator wants to have more defined control of where the traffic goes, because more sophisticated filtering is available with routers, and when a router is used to segment a network, the result is more controllable sections.

Gateways

Gateway is a general term for software running on a device that connects two different environments and that many times acts as a translator for them or somehow restricts their interactions. Usually a gateway is needed when one environment speaks a different language, meaning it uses a certain protocol that the other environment does not understand. The gateway can translate mail from one type of mail server and format it so that another type of mail server can accept and understand it, or it can connect and translate different data link technologies such as Fiber Distributed Data Interface (FDDI) to Ethernet (both of which are discussed in Chapter 11).

Gateways perform much more complex tasks than connection devices such as routers and bridges. However, some people refer to routers as gateways when they connect two unlike networks (Token Ring and Ethernet) because the router has to translate between the data link technologies. Figure 14-7 shows how a network access server (NAS) functions as a gateway between telecommunications and network connections.

When networks connect to a backbone, a gateway can translate the different technologies and frame formats used on the backbone network versus the connecting LAN protocol frame formats. If a bridge were set up between an FDDI backbone and an Ethernet LAN, the computers on the LAN would not understand the FDDI protocols and frame formats. In this case, a LAN gateway would be needed to translate the protocols used between the different networks.

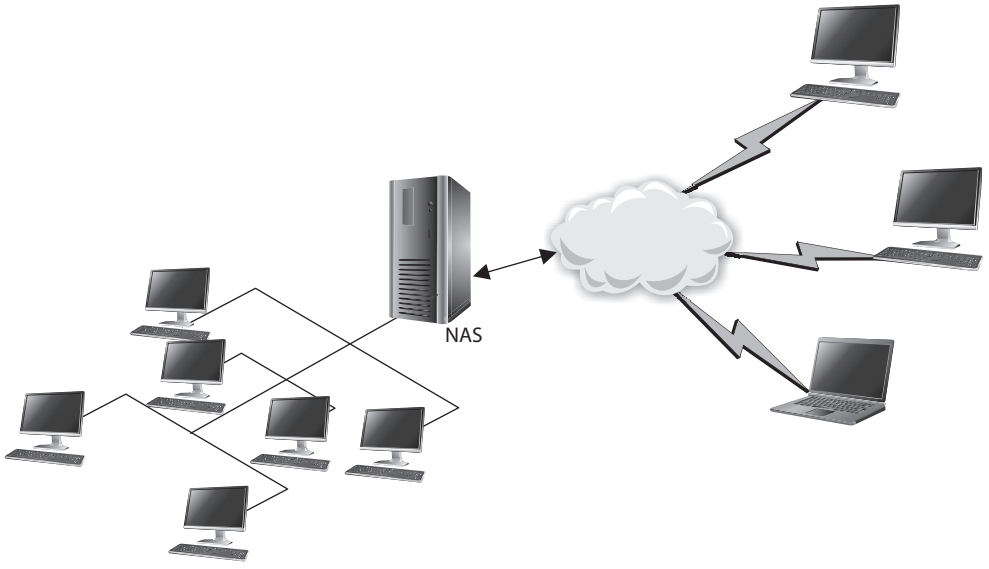


Figure 14-7 Several types of gateways can be used in a network. A NAS is one example.

A popular type of gateway is an *e-mail* gateway. Because several e-mail vendors have their own syntax, message format, and way of dealing with message transmission, e-mail gateways are needed to convert messages between e-mail server software. For example, suppose that David, whose corporate network uses Sendmail, writes an e-mail message to Dan, whose corporate network uses Microsoft Exchange. The e-mail gateway converts the message into a standard that all mail servers understand—usually X.400—and passes it on to Dan's mail server.

Proxy Servers

Proxy servers act as an intermediary between the clients that want access to certain services and the servers that provide those services. As a security professional, you do not want internal systems to directly connect to external servers without some type of control taking place. For example, if users on your network could connect directly to websites without some type of filtering and rules in place, the users could allow malicious traffic into the network or could surf websites your organization deems inappropriate. To prevent this situation, all internal web browsers should be configured to send their web requests to a web proxy server. The proxy server validates that the request is safe and then sends an independent request to the website on behalf of the user. A very basic proxy server architecture is shown in Figure 14-8.

The proxy server may cache the response it receives from the server so that when other clients make the same request, the proxy server doesn't have to make a connection out to the actual web server again but rather can serve up the necessary data directly. This drastically reduces latency and allows the clients to get the data they need much more quickly.

There are different types of proxies that provide specific services. A *forwarding proxy* is one that allows the client to specify the server it wants to communicate with, as in our scenario earlier. An *open proxy* is a forwarding proxy that is open for anyone to use. An anonymous open proxy allows users to conceal their IP address while browsing websites

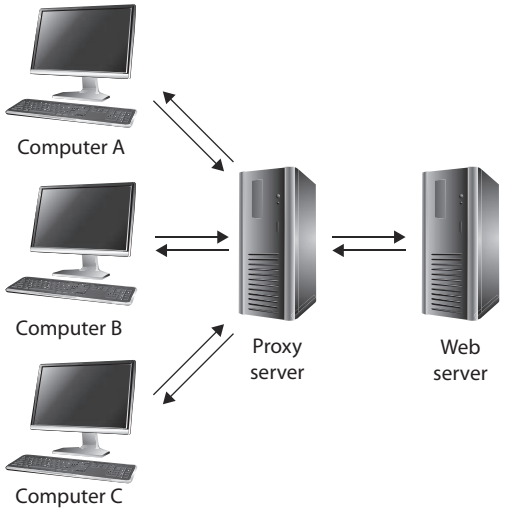


Figure 14-8 Proxy servers control traffic between clients and servers.

or using other Internet services. A *reverse proxy* appears to the clients as the original server. The client sends a request to what it thinks is the original server, but in reality this reverse proxy makes a request to the actual server and provides the client with the response. The forwarding and reverse proxy functionality seems similar, but as Figure 14-9 illustrates, a forwarding proxy server is commonly on an internal network controlling traffic that is exiting the network. A reverse proxy server is commonly on the network that fulfills clients' requests; thus, it is handling traffic that is entering its network. The reverse proxy can carry out load balancing, encryption acceleration, security, and caching.

Web proxy servers are commonly used to carry out content filtering to ensure that Internet use conforms to the organization's acceptable use policy (AUP). These types of proxies can block unacceptable web traffic, provide logs with detailed information pertaining to the websites specific users visited, monitor bandwidth usage statistics, block restricted website usage, and screen traffic for specific keywords (e.g., porn, confidential, Social Security numbers). The proxy servers can be configured to act mainly as caching servers, which keep local copies of frequently requested resources, allowing organizations to significantly reduce their upstream bandwidth usage and costs while significantly increasing performance.

While the most common use of proxy servers is for web-based traffic, they can be used for other network functionality and capabilities, as in DNS proxy servers. Proxy servers are a critical component of almost every network today. They need to be properly placed, configured, and monitored.

NOTE The use of proxy servers to allow for online anonymity has increased over the years. Some people use a proxy server to protect their browsing behaviors from others, with the goal of providing personal freedom and privacy. Attackers use the same functionality to help ensure their activities cannot be tracked back to their local systems.

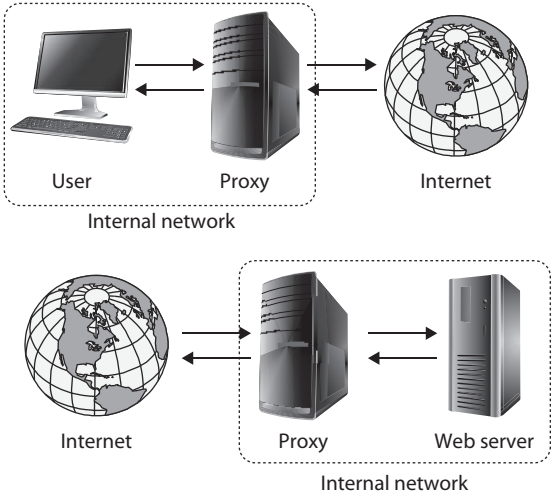


Figure 14-9 Forward vs. reverse proxy services

The Tor Network

Tor (originally known as The Onion Router) is a volunteer-operated network of computers around the world that work together to route encrypted web traffic. The goal of Tor is to keep your identity private online, or at least as close to private as is possible. (Misconfigurations or exploitable software on your local machine can still reveal your identity.) Every computer (or node) in Tor receives data from another node and passes it on to the next. Each node only knows where the encrypted data came from and where it's going next. After several hops, someone at the destination has no way of knowing who initiated the connection when you pop back up in the open Internet.

Tor can also provide access to so-called "hidden services" in the deep web that run only inside Tor. The infamous drug marketplace The Silk Road was an example of this. Tor is very popular among privacy advocates and people who live in countries that have strong censorship laws. However, Tor also is commonly used by criminal and even nation-state actors who want to protect their source location. Therefore, you should be extremely suspicious if you see Tor traffic in any enterprise network.

PBXs

Telephone companies use switching technologies to transmit phone calls to their destinations. A telephone company's central office houses the switches that connect towns, cities, and metropolitan areas through the use of optical fiber rings. So, for example, when

Putting It All Together: Network Devices

The network devices we've covered so far are the building blocks of almost any network architecture. Table 14-4 lists them and points out their important characteristics.

Device	OSI Layer	Functionality
Repeater	Physical	Amplifies the signal and extends networks
Bridge	Data link	Forwards packets and filters based on MAC addresses; forwards broadcast traffic, but not collision traffic
Switch	Data link	Provides a private virtual link between communicating devices; allows for VLANs; reduces collisions; impedes network sniffing
Router	Network	Separates and connects LANs creating internetworks; filters based on IP addresses
Gateway	Application	Connects different types of networks; performs protocol and format translations
Web proxy	Application	Acts as an intermediary between clients and servers, typically to improve security and/or performance

Table 14-4 Main Differences Between Network Devices

Dusty makes a landline phone call from his house, the call first hits the local central office of the telephone company that provides service to Dusty, and then the switch within that office decides whether it is a local or long-distance call and where it needs to go from there. A *Private Branch Exchange (PBX)* is a private telephone switch that is located on an organization's property. This switch performs some of the same switching tasks that take place at the telephone company's central office. The PBX has a dedicated connection to its local telephone company's central office, where more intelligent switching takes place.

A PBX can interface with several types of devices and provides a number of telephone services. The voice data is multiplexed onto a dedicated line connected to the telephone company's central office. Figure 14-10 shows how data from different data sources can be placed on one line at the PBX and sent to the telephone company's switching facility.

PBXs use digital switching devices that can control analog and digital signals. While these modern exchanges are more secure than their analog predecessors, that in no way means PBX systems are free from vulnerabilities. Many PBX systems have system administrator passwords that are hardly ever changed. These passwords are set by default; therefore, if 100 companies purchase and implement 100 PBX systems from the PBX vendor ABC and they do not reset the password, a *phreaker* (a phone hacker) who knows this default password now has access to 100 PBX systems. Once a phreaker breaks into a PBX system, she can cause mayhem by rerouting calls, reconfiguring switches, or configuring the system to provide her and her friends with free long-distance calls. This type of fraud happens more often than most organizations realize because many of them do not closely audit their phone bills. Though the term is not used as much nowadays, phreakers are very much an issue to our telecommunications systems. Toll fraud (as most of their activities are called) associated with PBX systems are estimated to cost over $3 billion in annual losses worldwide, according to the Communications Fraud Control Association's (CFCA) 2019 Fraud Loss Survey.

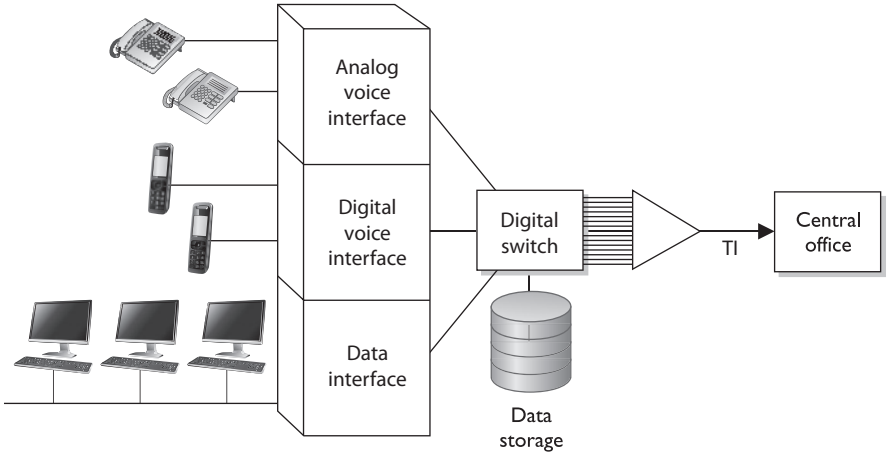


Figure 14-10 A PBX combines different types of data on the same lines.

PBX systems are also vulnerable to brute force and other types of attacks, in which phreakers use scripts and dictionaries to guess the necessary credentials to gain access to the system. In some cases, phreakers have listened to and changed people's voice messages. So, for example, when people call Bob and reach his voicemail, they might hear not his usual boring message but a new message that is screaming obscenities and insults.

Unfortunately, many security people do not even think about a PBX when they are assessing a network's vulnerabilities and security level. This is because telecommunication devices have historically been managed by service providers and/or by someone on the staff who understands telephony. The network administrator is usually not the person who manages the PBX, so the PBX system commonly does not even get assessed. The PBX is just a type of switch and it is directly connected to the organization's infrastructure; thus, it is a doorway for the bad guys to exploit and enter. These systems need to be assessed and monitored just like any other network device.

So, what should we do to secure PBX systems? Since many of these systems nowadays ride on IP networks, some of the basic security measures will sound familiar. Start by ensuring you know all accounts on the system and that their passwords are strong. Then, ensure that your PBX is updated regularly and that it sits behind your firewall with the appropriate ACLs in place. Other security measures are more specific to a PBX. For example, consider separating your voice and data traffic through these systems by placing them on different VLANs. If one of the VLANs is penetrated, the other could remain secure. Also, limiting the rate of traffic to IP telephony VLANs can slow down an outside attack.

Network Access Control Devices

Network access control (NAC) is any set of policies and controls that we use to, well, control access to our networks. The term implies that we will verify that a device satisfies certain requirements before we let it in. At its simplest level, this could just be user authentication, which was the theme of our discussion of the IEEE 802.1X standard when we were covering wireless network security in Chapter 12. The 802.1X protocol allows devices to connect in a very limited manner (i.e., only to the network authenticator) until we can verify the user credentials it presents.

To fully leverage the power of NAC, however, we should do much more. For starters, we can (and should) authenticate a device. Endpoint/device authentication should be familiar to you because you already use it whenever you establish an HTTPS connection to a web server. When a client requests a secure connection, the server responds with its certificate, which contains its public key issued by a trusted certificate authority (CA). The client then encrypts a secret session key using the server's public key, so only the server can decrypt it and then establish a symmetrically encrypted secure link. It is possible to configure a NAC device to authenticate itself in a similar manner, but also require the client device to do the same. Obviously, we'd need a certificate (and matching private key) installed on the client device for this to work. An alternative approach to using certificates is to use a hardware Trusted Platform Module (TPM) if the endpoint has one. We discussed TPMs in Chapter 9.

A common use of NAC is to ensure the endpoint is properly configured prior to it being allowed to connect to the network. For example, it is pretty common to check the version of the OS as well as the signatures for the antimalware software. If either of these is not current, the device may be placed in an untrusted LAN segment from which it can download and install the required updates. Once the device meets the access policy requirements, it is allowed to connect to the protected network.

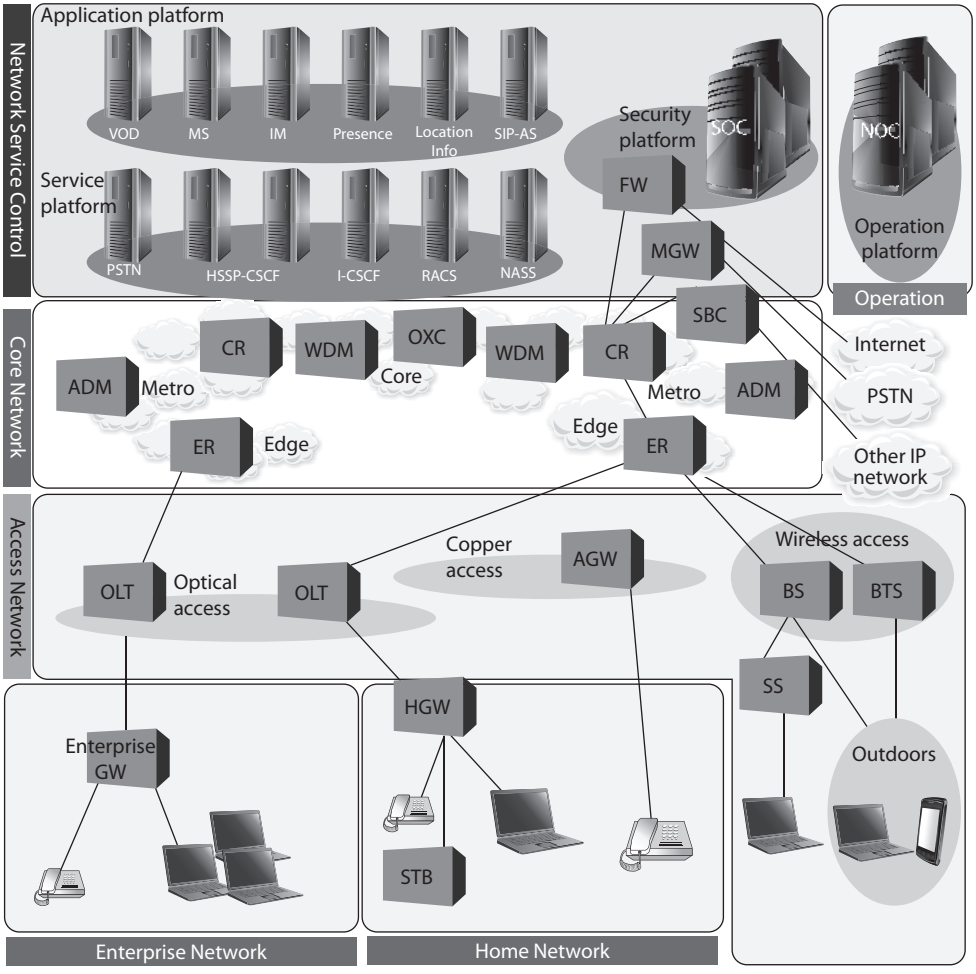
Network Diagramming

In many cases, you cannot capture a full network in a diagram because of the complexity of most organizations' networks. Sometimes we have a false sense of security when we have a pretty network diagram that we can all look at and be proud of, but let's dig deeper into why this can be deceiving. From what perspective should you look at a network? Many possibilities exist:

- A cabling diagram that shows how everything is physically connected (coaxial, UTP, fiber) and a wireless portion that describes the WLAN structure
- A network diagram that illustrates the network in infrastructure layers of access, aggregation, edge, and core
- A diagram that illustrates how the various networking routing takes place (VLANs, MPLS connections, OSPF, IGRP, and BGP links)
- A diagram that shows how different data flows take place (FTP, IPSec, HTTP, TLS, L2TP, PPP, Ethernet, FDDI, ATM, etc.)
- A diagram that separates workstations and the core server types that almost every network uses (DNS, DHCP, web farm, storage, print, SQL, PKI, mail, domain controllers, RADIUS, etc.)
- A view of a network based upon trust zones, which are enforced by filtering routers, firewalls, and DMZ structures
- A view of a network based upon its IP subnet structure

But what if you look at a network diagram from a Microsoft perspective, which illustrates many of these things but in forest, tree, domain, and OU containers? Then you need to show remote access connections, VPN concentrators, extranets, and the various MAN and WAN connections. How do we illustrate our IP telephony structure? How do we integrate our mobile device administration servers into the diagram? How do we document our new cloud computing infrastructure? How do we show the layers of virtualization within our database? How are redundant lines and fault-tolerance solutions marked? How does this network correlate and interact with our offsite location that carries out parallel processing? And we have not even gotten to our security components (firewalls, IDS, IPS, DLP, antimalware, content filters, etc.). And in the real world,

whatever network diagrams an organization does have are usually out of date because they take a lot of effort to create and maintain.



The point is that a network is a complex beast that cannot really be captured on one piece of paper. Compare it to a human body. When you go into the doctor's office, you see posters on the wall. One poster shows the circulatory system, one shows the muscles, one shows bones, another shows organs, and another shows tendons and ligaments; a dentist's office has a bunch of posters on teeth; if you are at an acupuncture clinic, there will be a poster on acupuncture and reflexology points. And then there is a ton of stuff no one makes posters for—hair follicles, skin, toenails, eyebrows—but these are all part of one system.

So what does this mean to the security professional? You have to understand a network from many different aspects if you are actually going to secure it. You start by learning all this network stuff in a modular fashion, but you need to quickly understand how it all works together under the covers. You can be a complete genius on how everything works within your current environment but not fully understand that when an employee connects her iPhone to her company laptop that is connected to the corporate network and uses it as a modem, this is an unmonitored WAN connection that can be used as a doorway by an attacker. Security is complex and demanding, so do not ever get too cocky, and always remember that a diagram is just showing a perspective of a network, not the whole network.

Operation of Hardware

Once you have your network designed and implemented, you need to ensure it remains operational. Keep in mind that one of the aspects of security is availability, which can be compromised not only by adversaries but also by power outages, equipment defects, and human error. Remember that all risks, not just the ones that come from human actors, should be addressed by your risk management program. This ensures that you can select cost-effective controls to mitigate those risks. In the sections that follow, we discuss three specific types of controls that protect the availability of your network components. These control types are redundant electrical power, equipment warranties, and support agreements on the operation of our network components.

Electrical Power

Electrical power is essential to operating IT hardware, which, in turn, runs the software that provides IT services to our organizations. We already discussed this topic generally in Chapter 10, but we now return to it in terms of ensuring our critical systems have redundant power. To understand these power requirements, we need to first become familiar with three key terms that describe electricity:

- **Voltage** Measured in volts, this tells us what the *potential* electric force between two points in a circuit could be. You can think of volts as the water pressure inside a pipe.
- **Current** Measured in amps, this is the *actual* electric flow through the circuit. If you think of volts as the pressure inside a water pipe, you can think of current as the diameter of a valve attached to it; the bigger the valve, the faster the water can come out.
- **Power** There are two ways to measure power. We measure electrical power in watts, which we calculate by multiplying voltage by amperage. In other words, if your server rack is running on 240 volts and drawing 9 amps of current, it is consuming 2,160 watts or 2.16 kilowatts (kW). Another related term is kilowatt-hours (kWh), which is simply the amount of power consumed during a 1-hour period. So, that same server rack would draw 2.16 kWh in one hour, or 51.84 kWh in a day (assuming the current draw is constant).

What we actually care about is whether or not we have enough electric power to run our equipment. There are two ways to measure power: apparent and real. You can think of *apparent power* as the maximum amount of electricity that could get through a circuit in a perfect case. This value is simply the product of the voltage and current of a system, and is measured in volt-amps (VA). So, if you have a 120-volt computer that can draw up to 3 amps, its apparent power would be 360 VA.

Typically, however, the real power drawn by a system is less than its apparent power. This is because of certain complexities of alternating current (AC) circuits that we won't dive into. Suffice it to say that AC, which is the type of current produced from virtually every power outlet, is constantly changing. This variance means that the *real power* drawn by a server will be some value, measured in watts, equal to or (much more frequently) lower than the apparent power. Thankfully, we don't have to calculate this value; most computing equipment is labeled with the real power value in watts (or kilowatts).

Why should you care? Because real power (watts) determines the actual power you purchase from the utility company, the size of any backup generators you might need, and the heat generated by the equipment. Apparent power (VA) is used for sizing wiring and circuit breakers, so the former don't melt (or worse, catch fire) and the latter don't trip. The ratio of real power to apparent power is called the *work factor*, which can never be greater than one (since the denominator is the ideal apparent power).

With all this discussion under our belts, we can now (finally) talk about redundant power, which typically comes in the two forms presented in Chapter 10: uninterruptable power supplies (UPSs) and backup power sources. Suppose one of your organization's facilities has (what will eventually turn out to be) an extended power outage lasting multiple days. Your business continuity plan (BCP; covered in Chapter 2) should identify your mission-critical systems and determine how long they can remain unavailable before your organizational losses are intolerable. You would have addressed this in your facility planning (Chapter 10) by implementing a backup power source. Typically, there is a period between the start of a power outage and when the backup power source comes online and is usable. This is the amount of time during which your UPS systems will have to keep your critical assets running.

To determine how much power you need from your backup power source, you simply add up the power consumption of your critical assets (in kW), keeping in mind the need for cooling and any other supporting systems. Let's say this comes out to be 6 kW and your backup source is a generator. Since generators run optimally at 75 percent to 80 percent of their rated loads, you'd need an 8-kW generator or greater. You also want to factor in room for growth, which should be no less than 25 percent, so you end up getting a 10-kW generator. Now, suppose you also get an automatic transfer switch that will start the generator and transfer the load from critical circuits 60 seconds after the outage is detected. How much UPS capacity do you need?

Whereas the real power consumption that you used to estimate your generator needs probably came from actual readings of how many kilowatts your critical servers drew, your apparent power needs are probably higher because they capture peaks in consumption that are averaged out by real power readings. Remember that apparent power is at least as much as (and usually higher than) your real power. If you look at your equipment's

technical descriptions (or labels) you may see a value measured in volt-ampere (VA or kVA), and all you have to do is add up these values and get a UPS that is rated for that value. Alternatively, a good rule of thumb is to multiply your real power by 1.4 kWA (kilowatt-ampere) per kVA. The resulting number of kVAs should give you sufficient UPS capacity until the generator kicks in.

Equipment Warranty

Of course, many other things can go wrong with our assets with or without power outages. Equipment failures due to manufacturing defects are, unfortunately, unavoidable in the long run. The good news is that most original equipment manufacturers (OEMs) provide a three-year warranty against such defects. However, you have to read the fine print and may want to upgrade the protections. Suppose that you have a critical server fail and you can only afford to have it down for 24 hours. The standard warranty includes next-day replacement delivery, so you're covered, right? Well, not if you factor in the time it'll take you to reconfigure the server, load up all the data it needs, and put it back into production. Since it is difficult and expensive to get better than next-day support, you may want to build in the cost of having a spare server (or two) in addition to the warranty to ensure you meet your maximum tolerable downtime (MTD).

Most OEMs also offer extended warranties at an additional cost. Depending on your hardware refresh cycle (i.e., how long you will operate equipment before replacing it with new systems), you may want to add one, two, or three more years to the base three-year warranty. This is usually cheaper to purchase when you buy the hardware, as opposed to purchasing it a year or two later. Seven to eight years after the initial purchase, however, warranty offers tend to expire, as the hardware will be too old for the OEM to continue supporting it.

Support Agreements

Even if your hardware doesn't fail, it could become unavailable (or insufficiently available) with regard to supporting your organizational processes. For example, suppose that a server slows down to the point where your users sit around for several seconds (or even minutes) waiting for a response. This would not only be frustrating but also lead to a loss of productivity that could add up to significant financial losses. If you have a large and well-staffed organization, you probably have a resident expert who can troubleshoot the server and get it back to peak performance. If you don't have such an expert, what do you do?

Many organizations use support agreements with third parties to deal with issues that are outside the expertise of their IT or security staff. Sometimes this support can be provided by the OEM as part of the purchase of a system. Other times, organizations hire a managed services provider (MSP), who not only responds when things go badly but continuously monitors the systems' performance to detect and fix problems as early as possible. Most MSPs charge flat monthly fees per device and include 24/7 remote monitoring, maintenance, and, when needed, onsite support. Think of this as an insurance policy against loss of availability.

Endpoint Security

An *endpoint* is any computing device that communicates through a network and whose principal function is not to mediate communications for other devices on that network. In other words, if a device is connected to a network but is not part of the routing, relaying, or managing of traffic on that network, then it is an endpoint. That definition leaves out all of the network devices we've discussed in the preceding sections. Endpoints include devices that you would expect, such as desktops, laptops, servers, smartphones, and tablets. However, they also include other devices that many of us don't normally think of, such as point of sale (POS) terminals at retail stores, building automation devices like smart thermostats and other Internet of Things (IoT) devices, and sensors and actuators in industrial control systems (ICS).

One of the greatest challenges in dealing with (and securing) endpoints is knowing they are present in the first place. While it would be extremely unusual (not to say frightening) for your routers and switches to unexpectedly drop in and out of the network, this is what mobile devices do by their very nature. The intermittent connectivity of mobile devices is also a problem when it comes to ensuring that they are properly configured and running the correct firmware, OS, and software versions. An approach to dealing with some of these issues is to use network access control (NAC), as discussed earlier in this chapter.

But mobile devices are not the only problem. Our increasing reliance on embedded systems like IoT and ICS devices poses additional challenges. For starters, embedded devices normally have lesser computing capabilities than other endpoints. You usually can't install security software on them, which means that many organizations simply

Securing Endpoints

Endpoint security really boils down to a handful of best practices. Sure, you should thoroughly analyze risks to your endpoints and implement cost-effective controls as part of a broader risk management program, but if you don't take care of the basic "tackling and blocking," then whatever else you do won't really make much of a difference. Here's a short list to get you started:

- Know what every single endpoint is, where it is, who uses it, and what it should (and should not) be doing.
- Strictly enforce least privilege (i.e., no regular users with local admin rights).
- Keep everything updated (ideally, do this automatically).
- Use endpoint protection and response (EDR) solutions.
- Back up everything (ideally in a way that is difficult for an attacker to compromise).
- Export endpoint logs to a security information and event management (SIEM) solution.

create security perimeters or bubbles around them and hope for the best. Just to make things even more interesting, IoT and ICS devices oftentimes control physical processes like heating, ventilation, and air conditioning (HVAC) that can have effects on the health and safety of the people in our organizations.

Content Distribution Networks

So far, our discussion of networking has sort of implied that there is *a* (singular) web server, a (singular) database server, and so on. While this simplifies our discussion of network foundations, protocols, and services, we all know that this is a very rare scenario in all but the smallest networks. Instead, we tend to implement multiples of each service, whether to segment systems, provide redundancy, or both. We may have a couple of web servers connected by a load balancer and interfacing with multiple backend database servers. This sort of redundant deployment can improve performance, but all clients still have to reach the same physical location regardless of where in the world they may be. Wouldn't it be nice if users in Europe did not have to ride transatlantic cables or satellite links to reach a server in the United States and instead could use one closer to them?

A *content distribution network (CDN)* consists of multiple servers distributed across a large region, each of which provides content that is optimized for users closest to it. This optimization can come in many flavors. For example, if you were a large streaming video distribution entity like Netflix, you would want to keep your movie files from having to traverse multiple links between routers, since each hop would incur a delay and potential loss of packets (which could cause jitter in the video). Reducing the number of network hops for your video packets would also usually mean having a server geographically closer to the other node, offering you the opportunity to tailor the content for users in that part of the world. Building on our video example, you could keep movies dubbed in Chinese on servers that are in or closer to Asia and those dubbed in French closer to Europe. So when we talk about optimizing content, we can mean many things.

Another benefit of using CDNs is that they make your Internet presence more resistant to distributed denial-of-service (DDoS) attacks. These attacks rely on having a large number of computers flood a server until it becomes unresponsive to legitimate requests. If an attacker can muster a DDoS attack that can send a million packets per second (admittedly fairly small by today's standards) and aim it at a single server, then it could very well be effective. However, if the attacker tries that against a server that is part of a CDN, the clients will simply start sending their requests to other servers in the network. If the attacker then directs a portion of his attack stream to each server on the CDN in hopes of bringing the whole thing down, the attack will obviously be diffused and would likely require many times more packets. Unsurprisingly, using CDNs is how many organizations protect themselves against DDoS attacks.

Chapter Review

The physical components that make up our networks are foundational to our information systems. Without these cables and switches and routers, nothing else would work. This may seem obvious, but when was the last time you inspected any of them to ensure

that they are secure, in good condition, properly configured, and well supported by appropriate third parties? The two classes of threat actors with which we should concern ourselves in this context are attackers and nature. We take care of the first by applying the principles of secure design we've discussed throughout the book and, particularly, by physically securing these cables and devices as discussed in Chapter 10. As far as natural threats, we need to be on the lookout for the wear and tear that is natural over time and that can exacerbate small product defects that may not have been apparent during our initial inspections of new products. This boils down to having qualified staff that is augmented, as necessary, by third parties that provide warranty and support services.

Quick Review

- Analog signals represent data as continuously changing wave values, while digital signals encode data in discrete voltage values.
- Digital signals are more reliable than analog signals over a long distance and provide a clear-cut and efficient signaling method because the voltage is either on (1) or not on (0), compared to interpreting the waves of an analog signal.
- Synchronous communications require a timing component but ensure reliability and higher speeds; asynchronous communications require no timing component and are simpler to implement.
- A baseband technology uses the entire communication channel for its transmission, whereas a broadband technology divides the communication channel into individual and independent subchannels so that different types of data can be transmitted simultaneously.
- Coaxial cable has a copper core that is surrounded by a shielding layer and grounding wire, which makes it more resistant to electromagnetic interference (EMI), provides a higher bandwidth, and supports the use of longer cable lengths.
- With twisted-pair cable, the twisting of the wires, the type of insulation used, the quality of the conductive material, and the shielding of the wire determine the rate at which data can be transmitted.
- Fiber-optic cabling carries data as light waves, is expensive, can transmit data at high speeds, is difficult to tap into, and is resistant to EMI and RFI. If security is extremely important, fiber-optic cabling should be used.
- Because it uses glass, fiber-optic cabling has higher transmission speeds that allow signals to travel over longer distances.
- Depending on the material used, network cables may be susceptible to noise, attenuation, and crosstalk.
- Line noise refers to random fluctuations in electrical-magnetic impulses that are carried along a physical medium.
- Attenuation is the loss of signal strength as it travels.
- Crosstalk is a phenomenon that occurs when electrical signals of one wire spill over to the signals of another wire.

- Bandwidth is the amount of information that can theoretically be transmitted over a link within a second.
- Data throughput is the actual amount of data that can actually be carried over a real link.
- A repeater provides the simplest type of connectivity because it only repeats electrical signals between cable segments, which enables it to extend a network.
- A bridge is a LAN device used to connect LAN segments (or VLAN segments) and thus extends the range of a LAN.
- A transparent bridge starts to learn about the network's environment as soon as it is powered on and continues to learn as the network changes by examining frames and making entries in its forwarding tables.
- Spanning Tree Protocol (STP) ensures that forwarded frames do not circle networks forever, provides redundant paths in case a bridge goes down, assigns unique identifiers to each bridge, assigns priority values to these bridges, and calculates path costs.
- The Shortest Path Bridging (SPB) protocol is defined in IEEE 802.1aq and is more efficient and scalable than STP; it is used in newer bridges.
- Switches are multiport bridges that typically have additional management features.
- Routers are layer 3, or network layer, devices that are used to connect similar or different networks.
- Routers link two or more network segments, where each segment can function as an independent network. A router works at the network layer, works with IP addresses, and has more network knowledge than bridges, switches, or repeaters.
- Gateway is a general term for software running on a device that connects two different environments and that many times acts as a translator for them or somehow restricts their interactions.
- A Private Branch Exchange (PBX) is a private telephone switch that is located on an organization's property and performs some of the same switching tasks that take place at the telephone company's central office.
- Proxy servers act as an intermediary between the clients that want access to certain services and the servers that provide those services.
- Network access control (NAC) is any set of policies and controls that restrict access to our networks.
- An endpoint is any computing device that communicates through a network and whose principal function is not to mediate communications for other devices on that network.
- A content distribution network (CDN) consists of multiple servers distributed across a large region, each of which provides content that is optimized for users closest to it.

Questions

Please remember that these questions are formatted and asked in a certain way for a reason. Keep in mind that the CISSP exam is asking questions at a conceptual level. Questions may not always have the perfect answer, and the candidate is advised against always looking for the perfect answer. Instead, the candidate should look for the best answer in the list.

1. Which of the following is true of asynchronous transmission signals?
   - **A.** Used for high-speed, high-volume transmissions
   - **B.** Robust error checking
   - **C.** Used for irregular transmission patterns
   - **D.** More complex, costly implementation
2. Which of the following technologies divides a communication channel into individual and independent subchannels?
   - **A.** Baseband
   - **B.** Broadband
   - **C.** Circuit-switched
   - **D.** Crosstalk
3. What type of cabling would you use if you needed inexpensive networking in an environment prone to electromagnetic interference?
   - **A.** Fiber-optic
   - **B.** Unshielded twisted pair (UTP)
   - **C.** Plenum
   - **D.** Coaxial
4. Which of the following issues would be likeliest to cause problems in a cable tray where large numbers of cables run in parallel and close proximity?
   - **A.** Thermal noise
   - **B.** Line noise
   - **C.** Crosstalk
   - **D.** Attenuation
5. What problem is inevitable as the length of a cable run increases?
   - **A.** Thermal noise
   - **B.** Line noise
   - **C.** Crosstalk
   - **D.** Attenuation

6. What is the term for the maximum amount of data that actually traverses a given network link?
   - **A.** Latency
   - **B.** Bandwidth
   - **C.** Throughput
   - **D.** Maximum transmission unit (MTU)
7. Which protocol ensures that frames being forwarded by switches do not circle networks forever?
   - **A.** Open Shortest Path First (OSPF)
   - **B.** Border Gateway Protocol (BGP)
   - **C.** Intermediate System-to-Intermediate System (IS-IS)
   - **D.** Spanning Tree Protocol (STP)
8. Which standard specifically addresses issues in network access control?
   - **A.** IEEE 802.1Q
   - **B.** IEEE 802.1aq
   - **C.** IEEE 802.AE
   - **D.** IEEE 802.1X
9. Which of the following would not be considered an endpoint?
   - **A.** Point of sale (POS) terminal
   - **B.** Industrial control system (ICS)
   - **C.** Internet of Things (IoT) device
   - **D.** Multiprotocol Label Switching (MPLS) system
10. All of the following are good reasons to implement a content distribution network except for which one?
    - **A.** Reduced latency
    - **B.** Reduced total cost of ownership (TCO)
    - **C.** Protection against distributed denial-of-service (DDoS) attacks
    - **D.** Tailoring content to users around the world

Answers

1. **C.** Asynchronous communications are typically used when data transfers happen at lower volumes and with unpredictable intervals. All other answers describe synchronous signaling, which is best suited for regular, high-volume traffic.

2. **B.** A broadband technology divides the communication channel into individual and independent subchannels so that different types of data can be transmitted simultaneously. A baseband technology, on the other hand, uses the entire communication channel for its transmission.
3. **D.** Coaxial cable has a copper core that is surrounded by a shielding layer and grounding wire, which makes it more resistant to electromagnetic interference (EMI). It is significantly cheaper than fiber-optic cable, which is the other EMI-resistant answer listed, while still allowing higher bandwidths.
4. **C.** Crosstalk is a phenomenon that occurs when electrical signals of one wire spill over to the signals of another wire. The more cables you have in close proximity, the worse this issue can be unless you use shielded cables.
5. **D.** Attenuation is the loss of signal strength as it travels. Regardless of which type of cabling is used, attenuation is inevitable given a long enough distance, which is why repeaters were invented.
6. **C.** Data throughput is the actual amount of data that can be carried over a real link. Bandwidth, on the other hand, is the amount of information that can theoretically be transmitted over a link within a second.
7. **D.** Spanning Tree Protocol (STP) ensures that forwarded frames do not circle networks forever, provides redundant paths in case a bridge goes down, assigns unique identifiers to each bridge, assigns priority values to these bridges, and calculates path costs. The other answers are all routing (layer 3) protocols.
8. **D.** The 802.1X protocol allows devices to connect in a very limited manner (i.e., only to the network authenticator) until the device and/or user can be authenticated. The other standards listed all pertain to layer 2 bridging and security.
9. **D.** An endpoint is any computing device that communicates through a network and whose principal function is not to mediate communications for other devices on that network. MPLS functionality is built into networking devices to help them move packets between endpoints more efficiently.
10. **B.** A content distribution network (CDN) consists of multiple servers distributed across a large region, each of which provides content that is optimized for users closest to it. This improves latency and localization. The very distributed nature of the CDN also provides DDoS protections. It all comes at significant costs and increases the complexity of deploying systems and content, which may require additional organizational resources apart from the service itself.

This page intentionally left blank

CHAPTER 15

Secure Communications Channels

This chapter presents the following:

- Voice communications
- Multimedia collaboration
- Remote access
- Data communications
- Virtualized networks
- Third-party connectivity

Mr. Watson—come here—I want to see you.

—Alexander Graham Bell

Up to this point, we've treated all the data as if it were equal. While it is true that a packet is a packet regardless of its contents, there are a number of common cases in which the purpose of a communication matters a lot. If we're downloading a file from a server, we normally don't care (or even know about) the variation in delay times between consecutive packets. This variation, known as *packet jitter*, could mean that some packets follow each other closely (no variance) while others take a lot longer (or shorter) time to arrive. While packet jitter is largely inconsequential to our file download, it could be very problematic for voice, video, or interactive collaboration communications channels.

Implementing secure communications channels has always been important to most organizations. However, the sudden shift to remote working brought on by COVID-19 has made the security of these channels critical due to the convergence of increased demand by legitimate users and increased targeting by threat actors. In this chapter, we look at some of the most prevalent communications channels that ride on our networks. These include voice, multimedia collaboration, remote access, and third-party channels. Let's start with the one we're most accustomed to: voice communications.

Voice Communications

Voice communications have come a long way since Alexander Graham Bell made that first call in 1876. It is estimated that 95 percent of the global population has access to telephone service, with most of those being cellular systems. What ties global voice networks together is a collection of technologies, some of which we've discussed before (e.g., ATM in Chapter 11 and LTE in Chapter 12), and some to which we now turn our attention.

Public Switched Telephone Network

The traditional telephone system is based on a circuit-switched, voice-centric network called the *public switched telephone network (PSTN)*. The PSTN uses circuit switching instead of packet switching. When a phone call is made, the call is placed at the PSTN interface, which is the user's telephone. This telephone is connected to the telephone company's local loop via electric wires, optical fibers, or a radio channel. Once the signals for this phone call reach the telephone company's central office (the end of the local loop), they are part of the telephone company's circuit-switching world. A connection is made between the source and the destination, and as long as the call is in session, the data flows through the same switches.

When a phone call is made, the phone numbers have to be translated, the connection has to be set up, signaling has to be controlled, and the session has to be torn down. This takes place through the Signaling System 7 (SS7) protocol. Figure 15-1 illustrates how calls are made in the PSTN using SS7. Suppose Meeta calls Carlos. Meeta's phone is directly connected to a signal switching point (SSP) belonging to the telephone company (telco) that provides her service. Her telco's SSP finds the SSP of the telco providing Carlos's phone service and they negotiate the call setup. The call itself is routed over

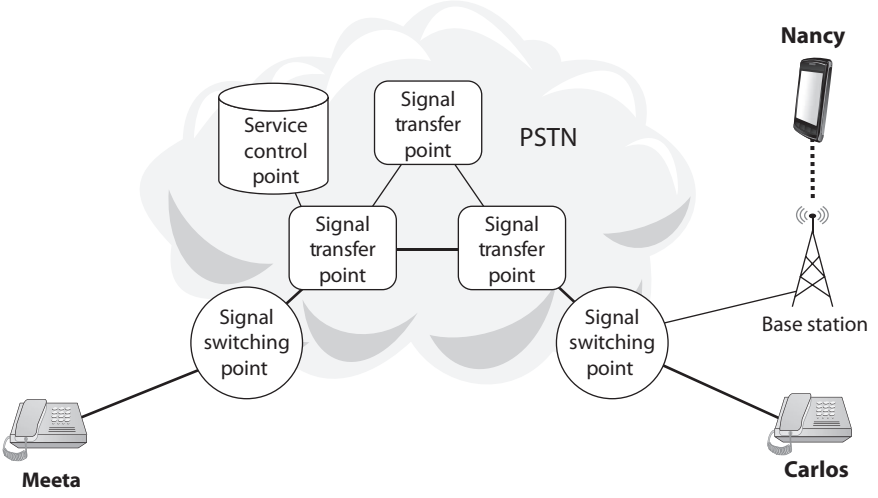


Figure 15-1 Major components of a public switched telephone network

the two signal transfer points (STPs) that interconnect the two SSPs. STPs perform a similar function in a circuit-switched network as routers do in an IP network. If Meeta wanted to call (or conference in) Nancy on her mobile phone, her SSP could query a service control point (SCP), which controls advanced features such as finding mobile subscribers' SSPs and enabling conference calls involving multiple networks.

NOTE PSTNs are being replaced with IP telephony. In the UK, for example, the service provider BT announced that it will switch off its PSTN in 2025.

DSL

It turns out that PSTN local loops (i.e., the telephone wires that go into our homes and offices) are able to support much more bandwidth than the small amount required for voice communications. In the 1980s, telcos figured out that they could transmit digital data at frequencies above those used for voice calls without interference. This was the birth of *digital subscriber line (DSL)*, which is a high-speed communications technology that simultaneously transmits analog voice and digital data between a home or business and the service provider's central office.

Figure 15-2 shows a typical DSL network. In the subscriber's home, a DSL modem creates a LAN to which computers and wireless access points can be connected. This modem, in turn, is connected to a DSL splitter if the home also has analog phone service. A bunch of DSL subscribers in the same neighborhood are then connected to a DSL access multiplexer (DSLAM) in the central office, where analog signals are sent to a voice switch (and on to the PSTN) and digital signals are routed out to the Internet. The tricky part is that the maximum distance between the DSLAM and the DSL splitter

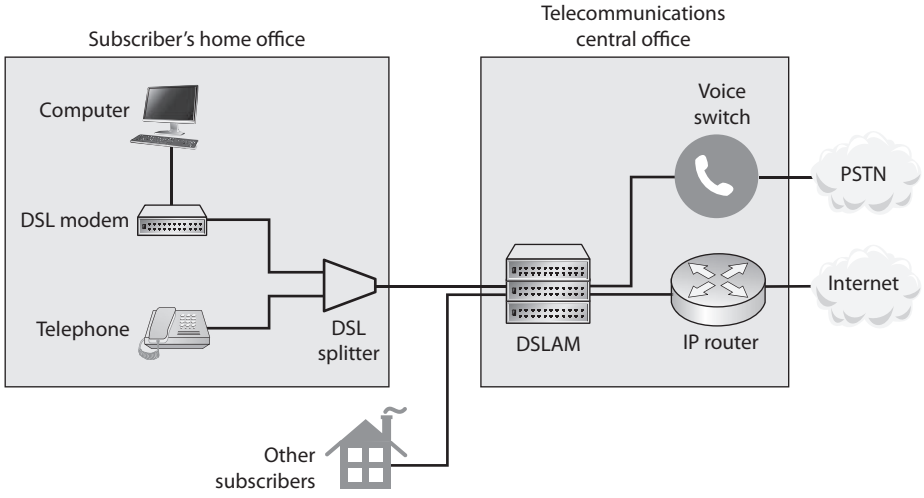


Figure 15-2 DSL network

in the subscriber's home cannot be greater than about 2.5 miles unless you put extenders in place to boost the signal strength.

DSL offers two broad types of services. With *symmetric services*, traffic flows at the same speed upstream and downstream (to and from the Internet or destination). With *asymmetric services*, the downstream speed is much higher than the upstream speed. The vast majority of DSL lines in use today are asymmetric, because most users usually download much more data from the Internet than they upload. The following are some of the most common types of DSL service:

- **Asymmetric DSL (ADSL)** These lines allocate more bandwidth for downstream data than for upstream. The technology has gone through multiple upgrades, with ADSL2+ (ITU standard G.992.5) being the latest and fastest. It has data rates of up to 24 Mbps downstream and 1.4 Mbps upstream, but can only support distances of about a mile from the central office. ADSL is generally used by residential users.
- **Very high-data-rate DSL (VDSL)** VDSL is basically ADSL at much higher data rates (up to 300 Mbps downstream and 100 Mbps upstream). It is capable of supporting high-bandwidth applications such as HDTV, telephone services (Voice over IP), and general Internet access over a single connection.
- **G.fast** Since the biggest challenge with DSL is the length of the subscriber loop, why not run fiber-optic cable from the central office to a distribution point near the home and then finish the last few hundred feet using the copper wires that are already in place? This is what G.fast (ITU standards G.9700 and G.9701) does. It can deliver data rates of up to 1 Gbps.

Dial-up Connections

Dial-up modems using PSTN were the dominant form of remote access in the early days of the Internet. Antiquated as they may seem, some organizations still have modems enabled, sometimes without the network staff being aware of them. For example, we once discovered that the facilities manager at a large school district installed a dial-up modem so he could control the HVAC systems remotely during inclement weather. Therefore, it is important to search for these systems and ensure no unauthorized modems are attached and operational.

If you find yourself using modems, some of the security measures that you should put in place for dial-up connections include

- Disable and remove nonessential modems.
- Configure the remote access server to call back the initiating phone number to ensure it is valid and authorized.
- Consolidate all modems into one location and manage them centrally, if possible.
- Whenever possible, implement use of two-factor authentication, VPNs, and NAC for remote access connections.

NOTE Despite being in wide use, DSL is an obsolescent technology. Major telecommunications companies around the world have announced plans to phase out DSL by 2025.

ISDN

Integrated Services Digital Network (ISDN) is another technology that leverages legacy telephone lines to enable data, voice, and signaling traffic to travel over a medium in a digital manner previously used only for analog voice transmission. ISDN uses the same wires and transmission medium used by analog dial-up technologies, but it works in a digital fashion. If a computer uses a modem to communicate with an ISP, the modem converts the data from digital to analog to be transmitted over the phone line. If that same computer was configured to use ISDN and had the necessary equipment, it would not need to convert the data from digital to analog, but would keep it in a digital form. This, of course, means the receiving end would also require the necessary equipment to receive and interpret this type of communication properly. Communicating in a purely digital form provides higher bit rates that can be sent more economically.

ISDN is a set of telecommunications services that can be used over public and private telecommunications networks. It provides a digital, point-to-point, circuit-switched medium and establishes a circuit between the two communicating devices. An ISDN connection can be used for anything a modem can be used for, but it provides more functionality and higher bandwidth. This digital service can provide bandwidth on an

ISDN Examined

ISDN breaks the telephone line into different channels and transmits data in a digital form rather than the old analog form. Three ISDN implementations are in use:

- **Basic Rate Interface (BRI) ISDN** This implementation operates over existing copper lines at the local loop and provides digital voice and data channels. It uses two B channels (at 64 Kbps each) to support user data or voice and one D channel (at 16 Kbps) for signaling, with a combined bandwidth of 144 Kbps. BRI ISDN is generally used for home and small office subscribers.
- **Primary Rate Interface (PRI) ISDN** This implementation has up to 23 B channels and 1 D channel, at 64 Kbps per channel. The total bandwidth is equivalent to a T1, which is 1.544 Mbps. This would be more suitable for an organization that requires a higher amount of bandwidth compared to BRI ISDN.
- **Broadband ISDN (BISDN)** This implementation can handle many different types of services simultaneously and is mainly used within telecommunications carrier backbones. When BISDN is used within a backbone, ATM is commonly employed to encapsulate data at the data link layer into cells, which travel over a SONET network.

as-needed basis and can be used for LAN-to-LAN on-demand connectivity, instead of using an expensive dedicated link.

Analog telecommunication signals use a full channel for communication, but ISDN can break up this channel into multiple channels to move various types of data and provide full-duplex communication and a higher level of control and error handling. ISDN provides two basic services: *Basic Rate Interface (BRI)* and *Primary Rate Interface (PRI)*.

BRI has two B channels that enable data to be transferred and one D channel that provides for call setup, connection management, error control, caller ID, and more. The bandwidth available with BRI is 144 Kbps, and BRI service is aimed at the small office and home office (SOHO) market. The D channel provides for a quicker call setup and process in making a connection compared to dial-up connections. An ISDN connection may require a setup connection time of only 2 to 5 seconds, whereas a modem may require a timeframe of 45 to 90 seconds. This D channel is an out-of-band communication link between the local loop equipment and the user's system. It is considered "out-of-band" because the control data is not mixed in with the user communication data. This makes it more difficult for a would-be defrauder to send bogus instructions back to the service provider's equipment in hopes of causing a denial of service (DoS), obtaining services not paid for, or conducting some other type of destructive behavior.

PRI has 23 B channels and one D channel, and is more commonly used in corporations. The total bandwidth is equivalent to a T1, which is 1.544 Mbps.

ISDN is not usually the primary telecommunications connection for organizations, but it can be used as a backup in case the primary connection goes down. An organization can also choose to implement *dial-on-demand routing (DDR)*, which can work over ISDN. DDR allows an organization to send WAN data over its existing telephone lines and use the PSTN as a temporary type of WAN link. It is usually implemented by organizations that send out only a small amount of WAN traffic and is a much cheaper solution than a real WAN implementation. The connection activates when it is needed and then idles out.

NOTE ISDN has lost popularity over the years and is now a legacy technology that is seldom used. Some organizations still rely on it as a backup for communications.

Cable Modems

The cable television companies have been delivering television services to homes for years, and then they started delivering data transmission services for users who have cable modems and want to connect to the Internet at high speeds. *Cable modems* provide high-speed access to the Internet through existing cable coaxial and fiber lines. The cable modem provides upstream and downstream conversions.

Coaxial and fiber cables are used to deliver hundreds of television stations to users, and one or more of the channels on these lines are dedicated to carrying data. The bandwidth is shared between users in a local area; therefore, it will not always stay at a

static rate. So, for example, if Mike attempts to download a program from the Internet at 5:30 P.M., he most likely will have a much slower connection than if he had attempted it at 10:00 A.M., because many people come home from work and hit the Internet at the same time. As more people access the Internet within his local area, Mike's Internet access performance drops.

Most cable providers comply with *Data-Over-Cable Service Interface Specifications (DOCSIS)*, which is an international telecommunications standard that allows for the addition of high-speed data transfer to an existing cable TV (CATV) system. DOCSIS includes MAC layer security services in its Baseline Privacy Interface/Security (BPI/SEC) specifications. This protects individual user traffic by encrypting the data as it travels over the provider's infrastructure.

IP Telephony

Internet Protocol (IP) telephony is an umbrella term that describes carrying telephone traffic over IP networks. So, if we have all these high-speed digital telecommunications services and the ability to transmit Voice over IP (VoIP) networks, do we even need analog telephones anymore? The answer is a resounding no. PSTN is being replaced by data-centric, packet-oriented networks that can support voice, data, and video. The new IP telephony networks use more efficient and secure switches, protocols, and communication links compared to PSTN but must still coexist (for now) with this older network. This means that VoIP is still going through a tricky transition stage that enables the old systems and infrastructures to communicate with the new systems until the old systems are dead and gone.

This technology gets around some of the barriers present in the PSTN today. The PSTN interface devices (telephones) have limited embedded functions and logic, and the PSTN environment as a whole is inflexible in that new services cannot be easily added. In VoIP, the interface to the network can be a computer, server, PBX, or anything else that runs a telephone application. This provides more flexibility when it comes to adding new services and provides a lot more control and intelligence to the interfacing devices. The traditional PSTN has basically dumb interfaces (telephones without much functionality), and the telecommunication infrastructure has to provide all the functionality. In VoIP, the interfaces are the "smart ones" and the network just moves data from one point to the next.

Because VoIP is a packet-oriented switching technology, the arrival times of different packets may not be regular. You may get a bunch of packets close to each other and then have random delays until the next ones arrive. This irregularity in arrival rates is referred to as *jitter*, which can cause loss of synchronicity in the conversation. It typically means the packets holding the other person's voice message got queued somewhere within the network or took a different route. VoIP includes protocols to help smooth out these issues and provide a more continuous telephone call experience.

EXAM TIP Applications that are time sensitive, such as voice and video signals, need to work over an isochronous network. An isochronous network contains the necessary protocols and devices that guarantee regular packet interarrival times.

Four main components are normally used for VoIP: an IP telephony device, a call-processing manager, a voicemail system, and a voice gateway. The *IP telephony device* is just a phone that has the necessary software that allows it to work as a network device. Traditional phone systems require a "smart network" and a "dumb phone." In VoIP, the phone must be "smart" by having the necessary software to take analog signals, digitize them, break them into packets, and create the necessary headers and trailers for the packets to find their destination. The *voicemail system* is a storage place for messages and provides user directory lookups and call-forwarding functionality. A *voice gateway* carries out packet routing and provides access to legacy voice systems and backup calling processes.

When a user makes a call, his VoIP phone sends a message to the *call-processing manager* to indicate a call needs to be set up. When the person at the call destination takes her phone off the hook, this notifies the call-processing manager that the call has been accepted. The call-processing manager notifies both the sending and receiving phones that the channel is active, and voice data is sent back and forth over a traditional data network line.

Moving voice data through packets is more involved than moving regular data through packets. This is because voice (and video) data must be sent as a steady stream, whereas other types of traffic are more tolerant to burstiness and jitter. A delay in data transmission is not noticed as much as is a delay in voice transmission. VoIP systems have advanced features to provide voice data transmission with increased bandwidth, while reducing variability in delay, round-trip delay, and packet loss issues. These features are covered by two relevant standards: H.323 and the Session Initiation Protocol (SIP).

NOTE A media gateway is the translation unit between disparate telecommunications networks. VoIP media gateways perform the conversion between TDM voice and VoIP, for example.

VoIP vs. IP Telephony

The terms "IP telephony" and "Voice over IP" are used interchangeably, but there is a distinction:

- The term "VoIP" is widely used to refer to the actual services offered: caller ID, QoS, voicemail, and so on.
- IP telephony is an umbrella term for all real-time applications over IP, including voice over instant messaging (IM) and video conferencing.

So, "IP telephony" means that telephone and telecommunications activities are taking place over an IP network instead of the traditional PSTN. "Voice over IP" means voice data is being moved over an IP network instead of the traditional PSTN. They are basically the same thing, but VoIP focuses more on the telephone call services.

H.323

The ITU-T *H.323* recommendation is a standard that deals with audio and video calls over packet-based networks. H.323 defines four types of components: terminals, gateways, multipoint control units, and gatekeepers. The *terminals* can be dedicated VoIP telephone sets, videoconferencing appliances, or software systems running on a traditional computer. *Gateways* interface between H.323 and non-H.323 networks, providing any necessary protocol translation. These gateways are needed, for instance, when using the PSTN to connect H.323 systems. *Multipoint control units (MCUs)* allow three or more terminals to be conferenced together and are sometimes referred to as *conference call bridges*. Finally, the H.323 *gatekeeper* is the central component of the system in that it provides call control services for all registered terminals.

Session Initiation Protocol

An alternative standard for voice and video calls is the *Session Initiation Protocol (SIP)*, which can be used to set up and break down the call sessions, just as SS7 does for PSTN calls. SIP is an application layer protocol that can work over TCP or UDP. It provides the foundation to allow the phone-line features that SS7 provides, such as causing a phone to ring, dialing a phone number, generating busy signals, and so on. SIP is used in applications such as video conferencing, multimedia, instant messaging, and online gaming.

SIP consists of two major components: the *User Agent Client (UAC)* and *User Agent Server (UAS)*. The UAC is the application that creates the SIP requests for initiating a communication session. UACs are generally messaging tools and soft-phone applications that are used to place VoIP calls. The UAS is the SIP server, which is responsible for handling all routing and signaling involved in VoIP calls.

SIP relies on a three-way-handshake process to initiate a session. To illustrate how a SIP-based call kicks off, let's look at an example of two people, Bill and John, trying to communicate using their VoIP phones. Bill's system starts by sending an INVITE message to John's system. Since Bill's system is unaware of John's location, the INVITE message is sent to the SIP server, which looks up John's address in the SIP *registrar* server. Once the location of John's system has been determined, the INVITE message is forwarded to his system. During this entire process, the server keeps the caller (Bill) updated by sending his system a Trying response, indicating the process is underway. Once the INVITE message reaches John's system, it starts ringing. While John's system rings and waits for John to respond, it sends a Ringing response to Bill's system, notifying Bill that the INVITE has been received and John's system is waiting for John to accept the call. As soon as John answers the call, an OK packet is sent to Bill's system (through the server). Bill's system now issues an ACK packet to begin call setup. It is important to note here that SIP itself is not used to stream the conversation because it's just a signaling protocol. The actual voice stream is carried on media protocols such as the *Real-time Transport Protocol (RTP)*. RTP provides a standardized packet format for delivering audio and video over IP networks. Once Bill and John are done communicating, a BYE message is sent from the system terminating the call. The other system responds with an OK, acknowledging the session has ended. This handshake is illustrated in Figure 15-3.

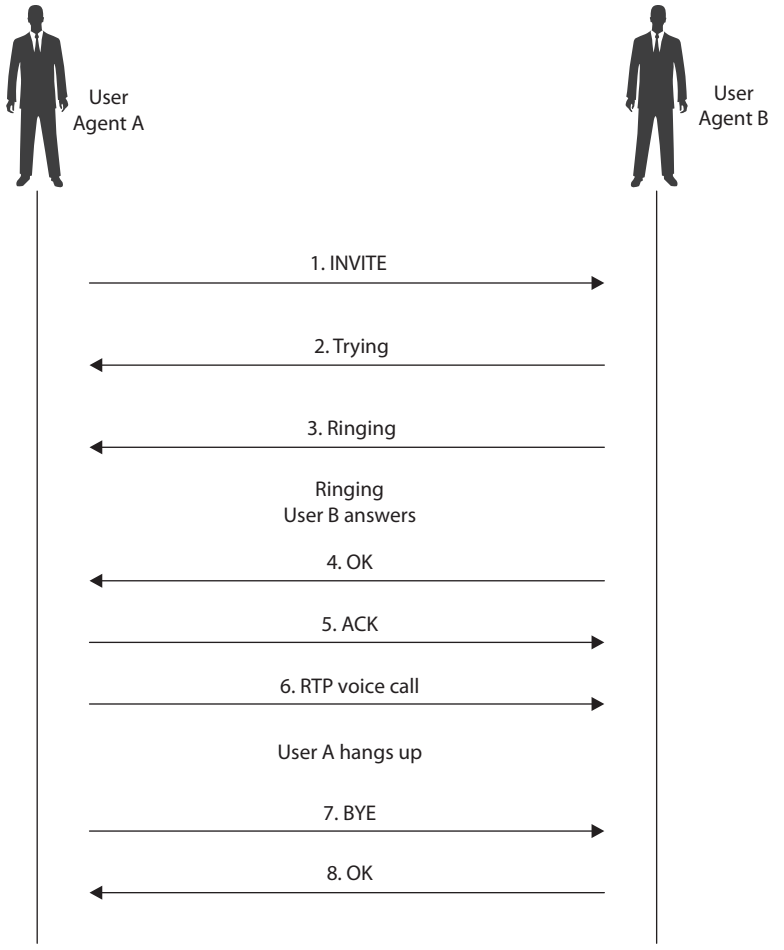
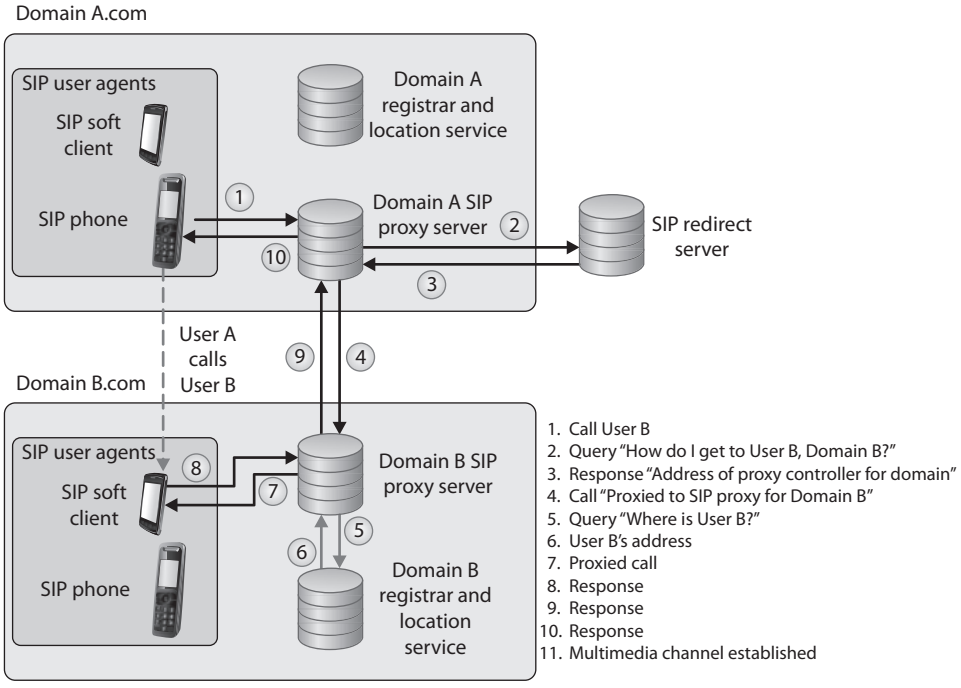


Figure 15-3
SIP handshake

The SIP architecture consists of three different types of servers, which play an integral role in the entire communication process of the VoIP system:

- **Proxy server** Is used to relay packets within a network between the UACs and the UAS. It also forwards requests generated by callers to their respective recipients. Proxy servers are also generally used for name mapping, which allows the proxy server to interlink an external SIP system to an internal SIP client.
- **Registrar server** Keeps a centralized record of the updated locations of all the users on the network. These addresses are stored on a location server.

- **Redirect server** Allows SIP devices to retain their SIP identities despite changes in their geographic location. This allows a device to remain accessible when its location is physically changed and hence while it moves through different networks. The use of redirect servers allows clients to remain within reach while they move through numerous network coverage zones. This configuration is generally known as an *intraorganizational* configuration. Intraorganizational routing enables SIP traffic to be routed within a VoIP network without being transmitted over the PSTN or external network.



Streaming Protocols

The Real Time Protocol (RTP) is a session layer protocol that carries data in media stream format, as in audio and video, and is used extensively in VoIP, telephony, video conferencing, and other multimedia streaming technologies. It provides end-to-end delivery services and is commonly run over the transport layer protocol UDP. *RTP Control Protocol (RTCP)* is used in conjunction with RTP and is also considered a session layer protocol. It provides out-of-band statistics and control information to provide feedback on QoS levels of individual streaming multimedia sessions.

IP Telephony Issues

VoIP's integration with the TCP/IP protocol has brought about some security challenges because it allows threat actors to leverage their TCP/IP experience to probe for flaws in both the architecture and the implementation of VoIP systems. Also involved are the traditional security issues associated with networks, such as unauthorized access, exploitation of communication protocols, and the spreading of malware. The promise of financial benefit derived from stolen call time is a strong incentive for most attackers. In short, the VoIP telephony network faces all the flaws that traditional computer networks have faced, plus the ones from legacy telephone systems too.

SIP-based signaling suffers from the lack of encrypted call channels and authentication of control signals. Attackers can tap into the SIP server and client communication to sniff out login IDs, passwords/PINs, and phone numbers. Once an attacker gets a hold of such information, she can use it to place unauthorized calls on the network. Toll fraud is considered to be the most significant threat that VoIP networks face, but illicit surveillance is also a threat for some organizations. If attackers are able to intercept voice packets, they may eavesdrop on ongoing conversations.

Attackers can also masquerade identities by redirecting SIP control packets from a caller to a forged destination to mislead the caller into communicating with an unintended end system. Like in any networked system, VoIP devices are also vulnerable to DoS attacks. Just as attackers would flood TCP servers with SYN packets on an IP network to exhaust a device's resources, attackers can flood RTP servers with call requests in order to overwhelm its processing capabilities. Attackers have also been known to connect laptops simulating IP phones to the Ethernet interfaces that IP phones use. These systems can then be used to carry out intrusions and DoS attacks. Attackers can also intercept RTP packets containing the media stream of a communication session to inject arbitrary audio/video data that may be a cause of annoyance to the actual participants.

Attackers can also impersonate a server and issue commands such as BYE, CHECKSYNC, and RESET to VoIP clients. The BYE command causes VoIP devices to close down while in a conversation, the CHECKSYNC command can be used to reboot VoIP terminals, and the RESET command causes the server to reset and reestablish the connection, which takes considerable time.

Combating VoIP security threats requires a well-thought-out infrastructure implementation plan. With the convergence of traditional and VoIP networks, balancing security while maintaining unconstrained traffic flow is crucial. VoIP calls can (and probably should) be encrypted over TLS. The use of authorization on the network is also an important step in limiting the possibilities of rogue and unauthorized entities on the network. Authorization of individual IP terminals ensures that only prelisted devices are allowed to access the network. Although not absolutely foolproof, this method can prevent rogue devices from connecting and flooding the network with illicit packets.

The use of secure cryptographic protocols such as TLS ensures that all SIP packets are conveyed within an encrypted and secure tunnel. The use of TLS can provide a secure channel for VoIP client/server communication and prevents the possibility of eavesdropping and packet manipulation.

VoIP Security Measures Broken Down

Hackers can intercept incoming and outgoing calls, carry out DoS attacks, spoof phone calls, and eavesdrop on sensitive conversations. Many of the countermeasures to these types of attacks are the same ones used with traditional data-oriented networks:

- Keep patches updated on each network device involved with VoIP transmissions:
  - The call-processing manager server
  - The voicemail server
  - The gateway server
- Encrypt VoIP traffic whenever possible.
- Identify unidentified or rogue telephony devices:
  - Implement authentication so only authorized telephony devices are working on the network.
- Install and maintain
  - Stateful firewalls
  - VPN for sensitive voice data
  - Intrusion detection
- Disable unnecessary ports and services on routers, switches, PCs, and IP telephones.
- Employ real-time monitoring that looks for attacks, tunneling, and abusive call patterns through IDS/IPS:
  - Employ content monitoring.
  - Use encryption when data (voice, fax, video) crosses an untrusted network.
  - Use a two-factor authentication technology.
  - Limit the number of calls via media gateways.
  - Close the media sessions after completion.

Multimedia Collaboration

The term *multimedia collaboration* is very broad and includes remotely sharing any combination of voice, video, messages, telemetry, and files during an interactive session. The term encompasses conferencing applications like Zoom, WebEx, and Google Meetings but also many other applications in disciplines such as project management, e-learning, science, telemedicine, and military. What distinguishes multimedia collaboration applications

is their need to simultaneously share a variety of data formats, each of which has different loss, latency, jitter, and bandwidth requirements. Of course, as we work to meet these performance requirements and allow maximum participation from authorized users (potentially around the world), we also have to ensure the security of this communication channel.

Meeting Applications

Imagine this scenario: You are hosting an online leadership meeting with your international partners to discuss the year ahead. Suddenly, a participant with a name you don't recognize starts sharing pornographic images and hate speech for all to see. You've just been "Zoom-bombed." (A term that doesn't necessarily mean you were using that particular platform.) This is what happens when access controls to your online meeting are inadequate. Many naïve users of meeting applications simply share a link with their guests, usually via e-mail or some other messaging application. Anyone with that link could then join the call if other precautions aren't taken.

The rise in popularity of meeting applications and their increased importance to the business of our organizations have put them in the crosshairs of a wide range of attackers beyond the Zoom-bombing troll we described. To prevent these attacks, consider the following best practices for securing online meeting applications:

- *Don't use consumer-grade products.* There is much wisdom in the old adage "you get what you pay for." Consumer-grade products are much cheaper than enterprise-grade ones (or even free), but they lack most security controls that we need to secure our organizational meetings.
- *Use AES 256-bit encryption.* It is rare to be able to support true end-to-end encryption for online meetings because most service providers need access to the traffic for things like recording, closed captioning, and echo cancelation. Still, you should ensure all call traffic is encrypted between each participant and the service provider.
- *Control access to every meeting.* Enterprise-grade conferencing services can integrate with your identity and access management service to ensure strong authentication. Failing that, ensure that, at a minimum, each meeting is password-protected.
- *Enable the waiting room feature, particularly for external participants.* Many services place participants in a virtual waiting room when they sign in to the meeting until the host lets them in. This gives you an opportunity to screen each participant prior to allowing them to join. At a minimum, ensure participants cannot connect to the call before the host does.
- *Restrict participants' sharing of their screens or cameras as appropriate.* This is particularly important when the meeting involves external parties such as partners or clients. While cameras may be desirable for a variety of reasons, it is rare for all participants to need unfettered screen sharing. Either way, ensure this is a deliberate decision by the host or organizer and enforceable by the platform.

Telepresence

Sometimes, you and other meeting participants need to do more than just see and hear each other and share slides remotely. *Telepresence* is the application of various technologies to allow people to be virtually present somewhere other than where they physically are. Consider a bomb disposal specialist trying to disarm an explosive device remotely using a robot, or a surgeon performing a delicate operation on a patient who would otherwise be inaccessible. The possibilities are endless and include the far more mundane applications that most of our organizations would consider, such as trade shows, pipeline inspections, and virtual reality (VR) training.

Because telepresence systems are not yet prevalent, there is no consensus yet on how to best secure them as a whole. Still, the secure design principles we've covered in this book (to which we'll return later in this chapter) apply to these systems.

- *Keep your software updated.* Online meeting software is no different than any other in the need for patch and update management. Even if you don't use dedicated clients and use web browsers to connect, you should ensure whatever you use is up to date.
- *Don't record meetings unless necessary.* It is helpful to record meetings, particularly when some participants cannot join in real time and must watch it later. However, the recordings can contain sensitive data that could be stolen or lead to other types of liability. If you do record the meeting, ensure it is for good reasons and that the recorded data is encrypted.
- *Know how to eject unwanted participants.* If you do get Zoom-bombed, that is not the time to figure out how to eject (and lock out) an offending participant. Ensure all hosts know how to do this beforehand and, while they're at it, learn also how to mute their microphones (and cameras) if needed.

Unified Communications

While meeting applications like videoconferencing systems have received a lot of attention recently, there is a broader application of multimedia collaboration services known as *unified communications (UC)*. UC is the integration of real-time and non-real-time communications technologies in one platform. Real-time communications are those that are instantaneous and interactive, such as telephone and video conferencing. Non-real-time communications, on the other hand, don't require our immediate attention and are exemplified by technologies such as e-mail and text messaging. The whole point of UC is that it integrates multiple modes of communication, as shown in Figure 15-4.

One of the key features of UC is the concept of *presence information*, which is an indicator of a subject's availability and willingness to communicate. If you have ever used a platform like Slack or Microsoft Teams, you will have noticed the presence icon next to your teammates. It may show that they are available, sleeping, on a call, or on a meeting. Presence information allows you to choose how to interact with your colleagues. If you

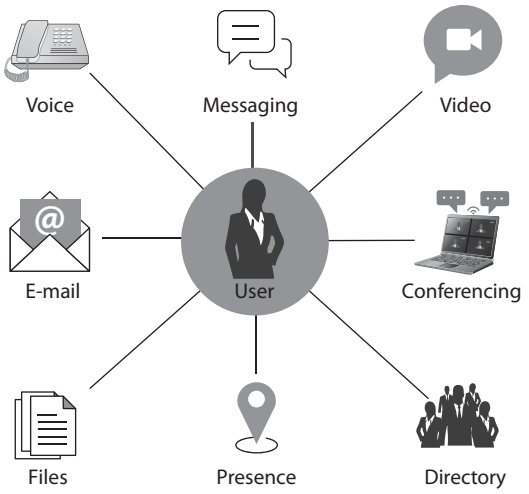


Figure 15-4 Unified communications components

need to get a message to Mohammed, who happens to be in a meeting, you can send him a text message. If, on the other hand, you see that Carmen is available, you may want to reach out to her on a voice or video call. Presence information can also show where in the world your colleagues are. For example, if you want to meet Bob and notice that he happens to be in the same city as you are, you may opt for a face-to-face meeting request.

Securing UC involves similar security controls that we would apply to any other communications platform, but with a couple of important caveats. For starters, UC relies on centralized data and access controls. This means that, whether your organization hosts its services on premises or in the cloud, there is a hub that supports and enables them. You want to ensure that this hub is adequately protected against physical and logical threats. Obviously, you want to protect your data, whether at rest or in motion, with strong encryption, but this will only get you so far if you allow anyone to access it. Consequently, you want to apply strict access controls that still allow the business processes to run efficiently. Finally, you want to ensure that demand spikes don't cause self-inflicted denial-of-service conditions. Instead, ensure that you have enough spare capacity to handle these inevitable (if rare) spikes.

Remote Access

Remote access covers several technologies that enable remote and home users to connect to resources that they need to perform their tasks. Most of the time, these users must first gain access to the Internet through an ISP, which sets up a connection to the destination network. For many organizations, remote access is a necessity because it enables users to access centralized network resources; it reduces networking costs by using the Internet as the access medium instead of expensive dedicated lines; and it extends the workplace for employees to their home computers, laptops, and mobile devices. Remote access can streamline access to resources and information through Internet connections and provides a competitive advantage by letting partners, suppliers, and customers have closely controlled links.

VPN

We discussed VPNs in Chapter 13 as a general concept, but let's circle back and see how to best employ them to provide secure remote connectivity for our staff members. VPNs are typically implemented using a client application that connects to a VPN server (commonly called a concentrator) in our organization. In a perfect world, you would have enough bandwidth and concentrator capacity to ensure all your remote staff members can simultaneously connect over the VPN. Then, you could enforce *always-on VPN*, which is a system configuration that automatically connects the device to the VPN with no user interaction. Obviously, this would only be possible with devices owned by the organization, but it can provide strong access controls if properly implemented. For even better results, you can implement a *VPN kill switch*, which automatically cuts off Internet access unless a VPN session is established.

Alas, things are usually a bit more complicated. Perhaps you don't have enough VPN capacity for your entire workforce, or you allow use of personal devices. If you cannot implement always-on VPN, the next best thing is to ensure you use multifactor authentication (MFA) and network access control (NAC). NAC is particularly important because you want to be able to check that the user device is safe before allowing it to access your corporate network. Since not everyone will be connecting to the VPN, you want to ensure that remote users have access to the resources they need and no others, possibly by putting them on the right VLANs and ensuring you have the right access control lists (ACLs) in your internal routers.

Regardless, you want to ensure your VPN systems (clients and concentrators) are updated and properly configured. Many clients allow you to select the cryptosystem to use, in which case you want to select the strongest option you can. Finally, carefully consider whether you will allow split tunnels.

A *VPN split tunnel* is a configuration that routes certain traffic (e.g., to the corporate data center) through the VPN while allowing other traffic (such as web searches) to access the Internet directly (without going through the VPN tunnel). The advantage of this approach is that users will be less likely to experience latency induced by an overworked concentrator. It also allows them to print to their local printer at home while on VPN. The disadvantage is that, should they pick up malware or otherwise become compromised on the Internet, the adversary will automatically get a free ride into your corporate network through the VPN. To prevent this from happening, you can enforce a *VPN full tunnel*, which routes all traffic through the concentrators.

VPN Authentication Protocols

While we're talking about VPN configuration, let's go over some of the authentication protocols you may come across, so you know what each brings to the table.

PAP The *Password Authentication Protocol (PAP)* is used by remote users to authenticate over Point-to-Point Protocol (PPP) connections such as those used in some VPNs. PAP requires a user to enter a password before being authenticated. The password and the username credentials are sent over the network to the authentication server after a connection has been established via PPP. The authentication server has a database of user credentials that are compared to the supplied credentials to authenticate users. PAP is one

of the least secure authentication methods because the credentials are sent in cleartext, which renders them easy to capture by network sniffers. PAP is also vulnerable to man-in-the-middle attacks. Although this protocol is not recommended for use anywhere, some (improperly configured) systems can revert to PAP if they cannot agree on any other authentication protocol.

EXAM TIP PAP has been considered insecure for decades. If you see it on the exam, consider it a bad choice.

CHAP The *Challenge Handshake Authentication Protocol (CHAP)* addresses some of the vulnerabilities found in PAP. It uses a challenge/response mechanism to authenticate the user instead of having the user send a password over the wire. When a user wants to establish a PPP connection and both ends have agreed that CHAP will be used for authentication purposes, the user's computer sends the authentication server a logon request. The server sends the user a challenge (called a nonce), which is a random value. This challenge is encrypted with the use of a predefined password as an encryption key, and the encrypted challenge value is returned to the server. The authentication server also uses the predefined password as an encryption key and decrypts the challenge value, comparing it to the original value sent. If the two results are the same, the authentication server deduces that the user must have entered the correct password and grants authentication. The steps that take place in CHAP are depicted in Figure 15-5. Unlike PAP, CHAP is not vulnerable to man-in-the-middle attacks because it continues this challenge/response activity throughout the connection to ensure the authentication server is still communicating with a user who holds the necessary credentials.

EXAM TIP MS-CHAP is Microsoft's version of CHAP and provides mutual authentication functionality. It has two versions, which are incompatible with each other.

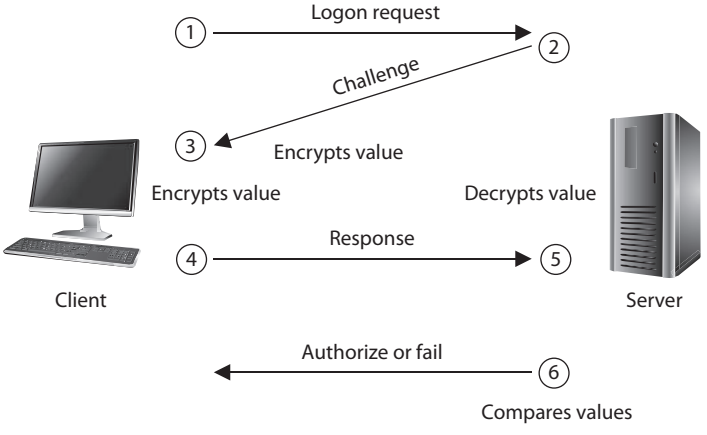


Figure 15-5 CHAP uses a challenge/response mechanism instead of having the user send the password over the wire.

EAP The *Extensible Authentication Protocol (EAP)* is also supported by PPP. Actually, EAP is not a specific authentication protocol as are PAP and CHAP. Instead, it provides a framework to enable many types of authentication techniques to be used when establishing network connections. As the name states, it *extends* the authentication possibilities from the norm (PAP and CHAP) to other methods, such as one-time passwords, token cards, biometrics, Kerberos, digital certificates, and future mechanisms. So when a user connects to an authentication server and both have EAP capabilities, they can negotiate between a longer list of possible authentication methods.

NOTE EAP has been defined for use with a variety of technologies and protocols, including PPP, Point-to-Point Tunneling Protocol (PPTP), Layer 2 Tunneling Protocol (L2TP), IEEE 802 wired networks, and wireless technologies such as 802.11 and 802.16.

There are many different variants of EAP, as shown in Table 15-1, because EAP is an extensible framework that can be morphed for different environments and needs.

Desktop Virtualization

Desktop virtualization technologies allow users to remotely interact with computers as if they were physically using them. In essence, these technologies present a virtual copy of a desktop that is running on some computer (physical or virtual) somewhere else

Protocol	Description
EAP-TLS	Digital certificate–based authentication, considered one of the most secure EAP standards
EAP-PSK	Provides mutual authentication and session key derivation using a preshared key
EAP-TTLS	Tunneled TLS, which requires the server to have a CA-issued certificate, but makes this optional for the client
EAP-IKE2	Internet Key Exchange version 2 (IKE2), which provides mutual authentication and session key establishment using asymmetric or symmetric keys or passwords
PEAPv0/EAP-MSCHAPv2	Similar in design to EAP-TTLS but only requires a server-side digital certificate
PEAPv1/EAP-GTC	Cisco variant based on Generic Token Card (GTC) authentication
EAP-FAST	Cisco-proprietary replacement for Lightweight EAP (LEAP) based on Flexible Authentication via Secure Tunneling (FAST)
EAP-SIM	For Global System for Mobile Communications (GSM), based on Subscriber Identity Module (SIM), a variant of PEAP for GSM
EAP-AKA	For Universal Mobile Telecommunication System (UMTS) Subscriber Identity Module (USIM) and provides Authentication and Key Agreement (AKA)
EAP-GSS	Based on Generic Security Service (GSS), uses Kerberos

Table 15-1 EAP Variants

in the network. IT staff frequently use desktop virtualization to manage rack-mounted servers (without having to attach a monitor, keyboard, and mouse to each), to log into jump boxes, and to manage and troubleshoot user workstations. In some organizations, remote desktop solutions allow staff to work from home and, through their personal devices, securely use an organizational computer. The upside of desktop virtualization is that the asset is protected by the organization's security architecture but still is accessible from almost anywhere. There are two main approaches to desktop virtualization: remote desktops and virtual desktop infrastructure.

NOTE A *jump box* (also called a *jump host* or *jump server*) is a hardened host that acts as a secure entry point or gateway into a sensitive part of a network.

Remote Desktops

Two of the most common approaches to providing remote desktops are Microsoft's *Remote Desktop Protocol (RDP)* and the open-source *Virtual Network Computing (VNC)* system. At a high level, both are very similar. They both require that a special server is running on the computer that will be controlled remotely and that the remote device has a software client installed and connected to the server, by default over port 3389 for RDP and 5900 for VNC. Although there are clients and servers for every major operating system, RDP is more common in Windows environments and VNC is more common in Linux environments.

The most important security consideration when deploying either RDP or VNC is to ensure that the connections are encrypted. Neither of these systems has robust security controls, so you have to tunnel them over a secure channel. If you are providing this service to remote users outside your organizational network, then you should ensure they are connected to the VPN. Having external RDP or VNC servers is a recipe for a security disaster, so their corresponding ports should be blocked at your firewall.

One of the advantages or disadvantages (depending on how you look at it) of RDP and VNC is that they allow a client to remotely control a specific computer. That computer must be provisioned somewhere on the network, specifically configured to allow remote access, and then must remain available. If it is powered off or is otherwise unavailable, there is nothing to remotely control.

Virtual Desktop Infrastructure

By combining virtualization and remote desktop technologies, we can create an environment in which users access the desktops of virtual machines (VMs) that look and behave exactly as the users have configured them, but that can be spun up or down, migrated, wiped, and re-created centrally as needed. *Virtual desktop infrastructure (VDI)* is a technology that hosts multiple virtual desktops in a centralized manner and makes them available to authorized users. Each virtual desktop can be directly tied to a VM (very similarly to the remote desktops described in the previous section) or can be a composite of multiple virtual components, such as a desktop template combined with virtual

applications running on multiple different VMs. This flexibility allows organizations to tailor desktops to specific departments, roles, or even individuals in a scalable and resource-effective manner.

VDI deployments can be either persistent or nonpersistent. In a *persistent VDI*, a given user connects to the same virtual desktop every time and is able to customize it as allowed by whatever organizational policies are in place. In a persistent model, users' desktops look the same at the beginning of one session as they did at the end of the last one, creating continuity that is helpful for long-term use and for complex workflows. By contrast, users of a *nonpersistent VDI* are presented with a standard desktop that is wiped at the end of each session. Nonpersistent infrastructures are useful when providing occasional access for very specific purposes or in extremely secure environments.

VDI is particularly helpful in regulated environments because of the ease with which it supports data retention, configuration management, and incident response. If a user's system is compromised, it can quickly be isolated for remediation or investigation, while a clean desktop is almost instantly spawned and presented to the user, reducing the downtime to seconds. VDI is also attractive when the workforce is highly mobile and may log in from a multitude of physical devices in different locations. Obviously, this approach is highly dependent on network connectivity. For this reason, organizations need to consider carefully their own network speed and latency when deciding how (or whether) to implement it.

Secure Shell

We don't always need a graphical user interface (GUI) to interact with our devices. In fact, there are many advanced use cases in which users, especially experienced and administrative ones, are more productive using a command-line interface (CLI). The tool of choice in many of these cases (particularly in Linux environments) is *Secure Shell (SSH)*, which functions as a type of tunneling mechanism that provides terminal-like access to remote computers. SSH is the equivalent of remote desktops but without the GUI. For example, the program can let Paul, who is on computer A, access computer B's files, run applications on computer B, and retrieve files from computer B without ever physically touching that computer. SSH provides authentication and secure transmission over vulnerable channels like the Internet.

NOTE SSH can also be used for secure channels for file transfer and port redirection.

SSH should be used instead of Telnet, FTP, rlogin, rexec, or rsh, which provide the same type of functionality SSH offers but in a much less secure manner. SSH is a program and a set of protocols that work together to provide a secure tunnel between two computers. The two computers go through a handshaking process and exchange (via Diffie-Hellman) a session key that will be used during the session to encrypt and protect the data sent. The steps of an SSH connection are outlined in Figure 15-6.

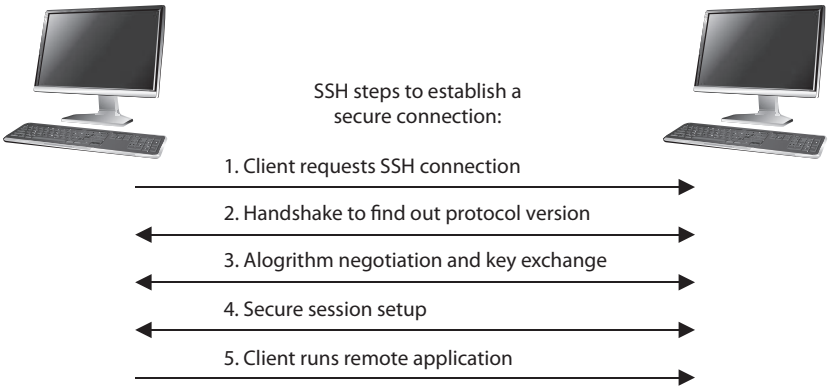
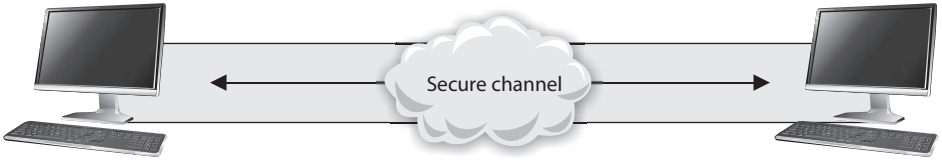


Figure 15-6 SSH is used for remote terminal-like functionality.

EXAM TIP Telnet is similar in overall purpose to SSH but provides none of the latter's security features. It is insecure and probably not the right answer to any question.

Once the handshake takes place and a secure channel is established, the two computers have a pathway to exchange data with the assurance that the information will be encrypted and its integrity will be protected.



Data Communications

Up to this point in this chapter, we've been focused on communications channels used by users. It is probably a good idea to also consider machine to machine data communications. Recall from Chapter 7 that there are multiple system architectures that require quite a bit of backend chatter between system components. For example, in an n-tier architecture, you may have an application server communicating quite regularly with a database. We must also map out and secure all these not-so-obvious data communications channels.

Network Sockets

A *network socket* is an endpoint for a data communications channel. A socket is a layer 4 (transport) construct that is defined by five parameters: source address, source port, destination address, destination port, and protocol (TCP or UDP). At any given time, a typical workstation has dozens of open sockets, each representing an existing data communications channel. (Servers can have thousands or even tens of thousands of them.) Each of these channels represents an opportunity for an attacker to compromise our systems. Do you know what all your data channels are?

This is one of the reasons why understanding our systems architectures is so critical. Many systems use default installation configurations that are inherently insecure. In addition to the proverbial (weak) default password, a brand-new server probably includes a number of services that are not needed and could provide an open door to attackers. Here are some best practices for securing sockets-based communications channels:

- Map out every authorized data communications channel to and from each server.
- Apply ACLs to block every connection except authorized ones.
- Use segmentation to ensure servers that communicate with each other regularly are in the same network segment.
- Whenever possible, encrypt all data communications channels.
- Authenticate all connection requests.

One of the challenges of securing data communications channels is that they rely on service accounts that usually run with elevated privileges. Oftentimes, these service accounts are excluded from the password policies that are enforced for user accounts. As a result, service account passwords are seldom changed and sometimes are documented in an unsecure manner. For example, we know of organizations that keep a list of their service accounts and passwords on a SharePoint or Confluence page for their IT team. These passwords should be protected just like any other privileged account and securely stored in a password vault.

Remote Procedure Calls

Moving up one level to the session layer (layer 5), a *remote procedure call (RPC)* allows a program somewhere in your network to execute a function or procedure on some other host. RPC is commonly used in distributed systems because it allows systems to divide larger tasks into subtasks and then hand those subtasks to other systems. Although the IETF defined an RPC protocol for Open Network Computing (ONC), the RPC concept can take many different forms in practice. In most networks (especially Windows ones), RPC services listen on TCP port 135. RPC use is ubiquitous in many enterprise environments because it is so powerful. However, by default, it doesn't provide any security beyond basic authentication.

If your organization uses RPC, then you should really consider upgrading its security. Secure RPC (S-RPC) provides authentication of both users and hosts as well as traffic encryption. As of February 9, 2021, Windows Active Directory (AD) systems require S-RPC. The IETF also released a standard for RPC security (RPCSEC) years ago, but because it is difficult to implement, it was never widely adopted. Instead, many organizations require TLS for authenticating hosts and encrypting RPC traffic. Other, vendor-specific implementations of RPC security exist, so you should research whatever versions are being used in your environment and ensure they are secure.

Virtualized Networks

A lot of the network functionality we have covered in this chapter can take place in virtual environments. You should remember from our coverage of virtual machines (VMs) in Chapter 7 that a host system can have virtual guest systems running on it, enabling multiple operating systems to run on the same hardware platform simultaneously. But the industry has advanced much further than this when it comes to virtualized technology. Routers and switches can be virtualized, which means you do not actually purchase a piece of hardware and plug it into your network, but instead you deploy software products that carry out routing and switching functionality. Obviously, you still need a robust hardware infrastructure on which to run the VMs, but virtualization can save you a lot of money, power, heat, and physical space.

These VMs, whether they implement endpoints or networking equipment, communicate with each other over virtual networks that behave much like their real counterparts, with a few exceptions. In order to understand some of these, let us first consider the simple virtual infrastructure shown in Figure 15-7. Let's suppose that VM-1 is an endpoint (perhaps a server), VM-2 is a firewall, and VM-3 is an IDS on the external side of the firewall. Two of these devices (VM-1 and VM-3) have a single virtual NIC (vNIC), while the other one (VM-2) has two vNICs. Every vNIC is connected to a virtual port on a virtual switch. Unlike the real world, any data that flows from one vNIC

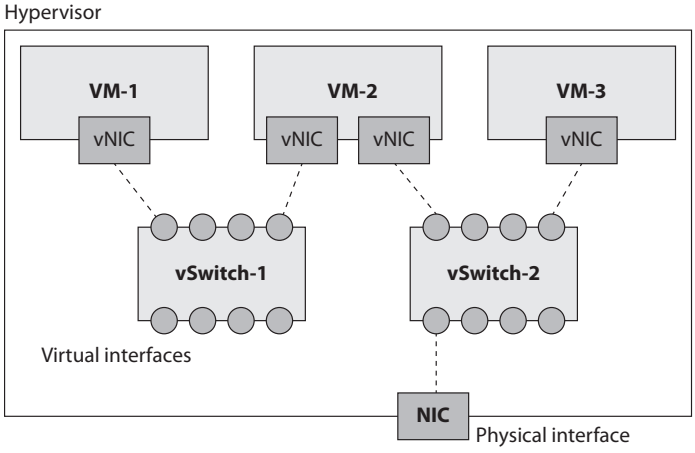


Figure 15-7 Virtualized networks

to another vNIC is usually just copied from one memory location (on the physical host) to another; it only pretends to travel the virtual network.

The single physical NIC in our example is connected to vSwitch-2, but it could just as easily have been directly connected to a vNIC on a VM. In this virtual network, VM-2 and VM-3 have connectivity to the physical network but VM-1 does not. The hypervisor stores in memory any data arriving at the physical NIC, asks the virtual switch where to send it, and then copies it into the memory location for the intended vNIC. This means that the hypervisor has complete visibility over all the data traversing its virtualized networks, whether or not it touches the physical NIC.

It should come as no surprise that one of the greatest strengths of virtualization, the hypervisor, is potentially also its greatest weakness. Any attacker who compromises the hypervisor could gain access to all virtualized devices and networks within it. So, both the good and the bad guys are intensely focused on finding any vulnerabilities in these environments. What should you do to ensure the security of your virtualized networks and devices? First, just as you should do for any other software, ensure you stay on top of any security patches that come out. Second, beware of third-party add-ons that extend the functionality of your hypervisor or virtual infrastructure. Ensure these are well tested and acquired from reputable vendors. Last, ensure that whoever provisions and maintains your virtualized infrastructure is competent and diligent, but also check their work. Many vulnerabilities are the result of misconfigured systems, and hypervisors are no different.

Third-Party Connectivity

We can't wrap up our discussion of securing the multitude of communications channels in our systems without talking about third parties. In Chapter 2, we covered the risks that third parties bring to our organizations and how to mitigate them. These third parties cover a broad spectrum that includes suppliers, service providers, and partners. Each of them may have legitimate needs to communicate digitally with our organizations, potentially in an automated manner. How can we provide this required connectivity to third parties without sacrificing our security? The answer can be found by applying the secure design principles we've been revisiting throughout the book:

- **Threat modeling** Always start by identifying the threats. What might malicious (or just careless) third parties be able to do with the communications channels we provide that would cause us harm? What are their likeliest and most dangerous actions? This deliberate exercise in understanding the threats is foundational.
- **Least privilege** Third parties will have legitimate connectivity requirements that we should minimally provide. If a contractor needs to monitor and control our HVAC systems remotely, we should segment those systems on the same VLAN and ensure that only specific calls from specific hosts to specific devices are allowed, and nothing more.

- **Defense in depth** Based on the threat model, we put in place controls to mitigate risks. But what happens if the first layer of controls fails to contain the threat? If that HVAC contractor is compromised in an island-hopping attack and the adversary is able to escape the VLAN, how do we detect the breach and then contain the attack?
- **Secure defaults** While ensuring that default configurations are secure is generally a best practice, it is particularly important on systems that will be used by third parties. One of the keys here is to enforce strict configuration management. For any system that will be accessible by a third party, we must ensure that all defaults are secure by testing them.
- **Fail securely** Speaking of testing, we should test the system under a range of conditions to see what happens when it breaks. For example, stress testing (under heavy usage loads), fuzzing, and power and network failure testing can show us what happens when a system fails. This is not specific to third-party systems, by the way.
- **Separation of duties** Giving third parties the least privileges needed actually makes separating duties easier. For example, it may be that the HVAC contractor does not normally start or stop the furnace, but this may be occasionally required. Because this can have an impact on our facility, the action must be approved by our site manager.
- **Keep it simple** This principle is centered on the statement of work (SoW) that describes the agreement with the third party and in the processes we build to support that work. A policy of "deny by default, allow by exception" can keep things simple, supports the least-privilege principle, and should be paired with a simple process for handling exceptions.
- **Zero trust** It goes without saying that we should not trust third parties when it comes to access to our systems. For every interaction of third parties with our systems, we must ensure that authentication, nonrepudiation, and audit controls are sufficient to detect and mitigate any threat (deliberate or otherwise) that they introduce into our environments.
- **Privacy by design** If we use this principle to guide the development of our entire security architecture (and we really ought to), then we really shouldn't have to do anything else to account for third parties using our systems, particularly if we couple privacy with least privilege in the first place.
- **Trust but verify** We already talked about auditability in the context of zero trust, but there is a difference between logging activities and analyzing those logs periodically (or even continually). What is the process by which our security staff verifies that the actions of third parties are appropriate? How are suspicious or malicious activities handled?
- **Shared responsibility** Finally, who is contractually responsible for what? As the saying goes, "good fences make good neighbors." It is important to define responsibilities in the service or partnership agreement so that there are no misunderstandings and, should someone fail, we can take financial or legal actions to recover our losses.

Chapter Review

With this chapter, we have finished our coverage of the fourth domain of the CISSP Common Body of Knowledge, Communication and Network Security, by discussing the myriad of technologies that allow us to create secure communications channels in our organizations. Though most people (particularly in the technology fields) would not consider voice to be their primary means of communication, it remains important for many reasons, not the least of which is the fact that traditional voice channels are more commonly used nowadays for digital data traffic. It is important to understand how these technologies blend in different ways so that we can better secure them.

The COVID-19 pandemic forced most organizations around the world to quickly move toward (or improve their ability at) supporting a remote workforce largely based in home offices. While the news media regularly featured stories on the vulnerabilities and attacks on our multimedia collaboration and remote access systems, it is remarkable how well these held up to the sudden increase in use (and attacks). We hope that this chapter has given you a better understanding of how security professionals can continue to improve the security of these systems while supporting a remote workforce and third-party connectivity.

Quick Review

- The public switched telephone network (PSTN) uses circuit switching instead of packet routing to connect calls.
- The Signaling System 7 (SS7) protocol is used for establishing and terminating calls in the PSTN.
- The main components of a PSTN network are signal switching points (SSPs) that terminate subscriber loops, signal transfer points (STPs) that interconnect SSPs and other STPs to route calls through the network, and service control points (SCPs) that control advanced features.
- A digital subscriber line (DSL) is a high-speed communications technology that simultaneously transmits analog voice and digital data between a home or business and a PSTN service provider's central office.
- Asymmetric DSL (ADSL) has data rates of up to 24 Mbps downstream and 1.4 Mbps upstream but can only support distances of about a mile from the central office without signal boosters.
- Very high-data-rate DSL (VDSL) is a higher-speed version of ADSL (up to 300 Mbps downstream and 100 Mbps upstream).
- G.fast is DSL that runs over fiber-optic cable from the central office to a distribution point near the home and then uses legacy copper wires for the last few hundred feet to the home or office. It can deliver data rates of up to 1 Gbps.
- Integrated Services Digital Network (ISDN) is an obsolescent pure digital technology that uses legacy phone lines for both voice and data.

- Basic Rate Interface (BRI) ISDN is intended to support a single user with two channels each with data throughput of 64 Kbps.
- Primary Rate Interface (PRI) ISDN has up to 23 usable channels, at 64 Kbps each, which is equivalent to a T1 leased line.
- Cable modems provide high-speed access to the Internet through existing cable coaxial and fiber lines, but the shared nature of these media result in inconsistent throughputs.
- Internet Protocol (IP) telephony is an umbrella term that describes carrying telephone traffic over IP networks.
- The terms "IP telephony" and "Voice over IP" are used interchangeably.
- Jitter is the irregularity in the arrival times of consecutive packets, which is problematic for interactive voice and video communications.
- The H.323 recommendation is a standard that deals with audio and video calls over packet-based networks.
- The Session Initiation Protocol (SIP) is an application layer protocol used for call setup and teardown in IP telephony, video and multimedia conferencing, instant messaging, and online gaming.
- The Real-time Transport Protocol (RTP) is a session layer protocol that carries data in media stream format, as in audio and video, and is used extensively in VoIP, telephony, video conferencing, and other multimedia streaming technologies.
- RTP Control Protocol (RTCP) is used in conjunction with RTP and is also considered a session layer protocol. It provides out-of-band statistics and control information to provide feedback on QoS levels of individual streaming multimedia sessions.
- Multimedia collaboration is a broad term that includes remotely and simultaneously sharing any combination of voice, video, messages, telemetry, and files in an interactive session.
- Telepresence is the application of various technologies to allow people to be virtually present somewhere other than where they physically are.
- Unified communications (UC) is the integration of real-time and non-real-time communications technologies in one platform.
- An always-on VPN is a system configuration that automatically connects the device to the VPN with no user interaction.
- A VPN kill switch is a system configuration that automatically cuts off Internet access unless a VPN session is established.
- A VPN split tunnel is a configuration that routes certain traffic through the VPN while allowing other traffic to access the Internet directly.

- The Password Authentication Protocol (PAP) is an obsolete and insecure authentication protocol that sends user credentials in plaintext and should not be allowed.
- The Challenge Handshake Authentication Protocol (CHAP) uses a challenge/response mechanism using the password as an encryption key to authenticate the user instead of having the user send a password over the wire.
- The Extensible Authentication Protocol (EAP) is a framework that enables many types of authentication techniques to be used when establishing network connections.
- Desktop virtualization technologies, such as remote desktops and virtual desktops, allow users to remotely interact with computers as if they were physically using them.
- Two of the most common approaches to providing remote desktops are Microsoft's Remote Desktop Protocol (RDP) and the open-source Virtual Network Computing (VNC) system.
- Virtual desktop infrastructure (VDI) is a technology that hosts multiple virtual desktops in a centralized manner and makes them available to authorized users.
- Secure Shell (SSH) is a secure tunneling mechanism that provides terminal-like access to remote computers.
- A network socket is an endpoint for a data communications channel, defined by five parameters: source address, source port, destination address, destination port, and protocol (TCP or UDP).
- Remote procedure calls allow a program somewhere in your network to execute a function or procedure on some other host.

Questions

Please remember that these questions are formatted and asked in a certain way for a reason. Keep in mind that the CISSP exam is asking questions at a conceptual level. Questions may not always have the perfect answer, and the candidate is advised against always looking for the perfect answer. Instead, the candidate should look for the best answer in the list.

1. In which type of networks is the Signaling System 7 (SS7) protocol used?

A. Integrated Services Digital Network (ISDN)

B. IP telephony network

C. Real-time Transport Protocol (RTP) network

D. Public switched telephone network (PSTN)