

CHAPTER

14

Network Device Security

This chapter is about how to use routers and switches to increase the security of the network. The first half of the chapter presents a tutorial on the basics of routers and switches, while the second half provides configuration steps for protecting the devices themselves against attacks. Because Cisco routers are the dominant platform in use today, we will show some examples based on the Cisco platform, but keep in mind that other platforms perform similar if not identical functions.

Traditionally, routers and switches have been managed by using a command-line interface (CLI), but interfaces have evolved over time toward graphical configuration solutions. CLIs are still available, but *web user interfaces (web UIs)* have become ubiquitous and are the most commonly used configuration tools these days. Additional functionality has converged into “all-in-one” devices such as *unified threat management (UTM)* platforms (firewalls combined with network antivirus, web filtering, application network communication control, IPS, and other network-oriented security functions, often bundled into switches both large and consumer sized). It is important to consider that adding the management capabilities of a web UI and other software-oriented capabilities to a network device also adds vulnerability—you now have a web server and additional software running on the very devices that control everything else in the environment. Paying special attention to securing the device interface and feature set will help ensure that you protect the network devices appropriately.

Chapter 15 will discuss firewalls and their ability to filter TCP/IP traffic—firewalls decide which traffic is permitted to enter and exit a given network. While firewalls can be thought of as the traffic cops of the information superhighway, routers and switches can be thought of as the major interchanges and the on and off ramps of those highways.

Switch and Router Basics

The dominant internetworking protocol in use today is known as *Transmission Control Protocol/Internet Protocol version 4 (TCP/IP or IPv4)*, although *IPv6* is on the horizon and is deployed in some carrier networks today. TCP/IP provides all the necessary components and mechanisms to transmit data between two computers over a network. TCP/IP is actually a suite of protocols

and applications that have discrete functions that map to the *Open Systems Interconnection (OSI) model*, sometimes referred to as the *OSI stack*, which will be covered in the next section. For this chapter, we are primarily concerned with TCP/IP functions at the second and third layers of the OSI model, commonly known as the data-link layer and network layer, respectively. These layers are also described in a bit more detail in Chapter 15, in the context of firewalls.

MAC Addresses, IP Addresses, and ARP

Each device on a network actually has two network-related addresses: a layer two address known as the *Media Access Control (MAC) address* (also known as the *hardware address* or *physical address*), and a layer three address known as the *IP address*. MAC addresses are 48-bit hexadecimal numbers that are uniquely assigned to each hardware network interface by the manufacturer, or virtually created by a hypervisor in a virtualized environment from a set range of addresses (different hypervisors have different methods for generating these). Some networking protocols also generate and use virtual MAC addresses for high-availability features, such as *Hot Standby Router Protocol (HSRP)* or *high-availability (HA)* clusters that maintain one active virtual device no matter which piece of hardware has assumed the active role.

Each hardware manufacturer has been assigned a range of MAC addresses to use, and each MAC address that has ever been assigned to a physical *network interface card (NIC)* is globally unique because it allows the underlying communication protocols to select the right system for network communications (although virtual MAC addresses may be used in more than one place, because although the algorithms used to generate them are similar and can start with the same reference values, as long as the same two MACs do not appear on the same network segment, they will work). IPv4 addresses are 32-bit numbers assigned by your network administrator that allow for the creation of logical and ordered addressing on a local network. IPv6 addresses are 128-bit, but, like IPv4, each IP address must be unique on a given network.

To send traffic, a device must have the destination device's IP address as well as a MAC address. Knowing the destination device's host name, the sending device can obtain the destination device's IP address using protocols such as *Domain Name Service (DNS)*. To ascertain a MAC address, the host uses the *Address Resolution Protocol (ARP)*, which functions by sending a broadcast message to the network that basically says, "Who has 192.168.2.10, tell 192.168.2.15." If a host receives that broadcast and knows the answer, it responds with the MAC address: "ARP 192.168.2.10 is at ab:cd:ef:00:01:02." Does this sound like an overly trusting protocol? It was designed by people who had no reason to think anybody would ever abuse it. However, note that no authentication or verification is done for any ARP replies that are received. This facilitates an attack known as *ARP poisoning*, discussed later in this chapter and in more detail in Chapter 2. ARP poisoning is one of the most effective and hard-to-defend attack techniques still in widespread use today.

For traffic destined to nonlocal segments, the MAC address of the local router is used. MAC addresses are really only relevant for devices that are locally connected, not those that require packets to travel through layer three devices, such as routers.

NOTE This is a very simplified review of TCP/IP. For a complete discussion, read *TCP/IP Illustrated*, volumes 1 and 2, by Richard Stevens.

TCP/IP

The fundamental purpose of TCP/IP is to provide computers with a method of transmitting data from one computer to another over a network. The purpose of a firewall is to control the passage of TCP/IP packets between hosts and networks.

In actuality, TCP/IP is a suite of protocols and applications that perform discrete functions corresponding to specific layers of the Open Systems Interconnection (OSI) model. Data transmission using TCP/IP is accomplished by independently transmitting blocks of data across a network in the form of packets, and each layer of the TCP/IP model adds a header to the packet. Depending on the firewall technology in use, the firewall will use the information contained in these headers to make access control decisions. If the firewall is application-aware, as application gateways are, access control decisions can also be made on the data portion or payload of the packet.

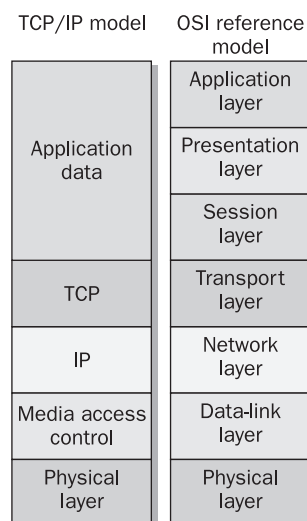


Figure 14-1 The TCP/IP model and the OSI reference model

Brief Overview of the OSI Layer

The OSI model uses a seven-layer structure to represent the transmission of data from an application residing on one computer to an application residing on another computer. TCP/IP does not strictly follow the seven-layer OSI model, having integrated the upper OSI layers into a single application layer. Figure 14-1 shows a graphical representation of the OSI reference model and its relationship to the TCP/IP implementation.

Table 14-1 highlights the functions performed by each layer of the OSI reference model.

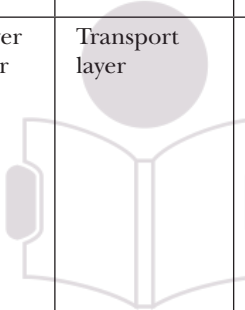
Ports and TCP/IP

To enable communications within the TCP/IP stack and to facilitate connections between two hosts, most well-known services are run on universally known ports. Firewalls base some or all of their access control decisions on the port information contained within packet headers.

Without universally known ports, providers of services would need to inform each of their users of the proper ports to use. For example, port 80 is the well-known port for HTTP, and almost all web servers on the Internet are configured to service HTTP requests on port 80. Connecting on any other port would result in an error unless the web server had been configured to listen on that nondefault port and respond to the requests. If an administrator chose to have the web server use port 81, they would have to inform all their users to specifically connect on port 81 (usually done in a browser by specifying the port at the end of the URL, like this: `www.example.com:81`).

In addition to the destination ports, TCP (and UDP) packets also contain a source port. The source port is the port where the client TCP/IP stack initiated communications to the server's destination port. This port becomes the destination port for the packets sent back by the server.

The source port is normally assigned semi-randomly by the TCP (or UDP) process on the source host, and it is typically some number above 1,023 but below 65,535, although



Layer seven	Application layer	Provides the protocol (commonly accepted and published language syntax and functions) for applications to access networked services. The most well-known application-layer protocols in use today are HTTP (which presents data to a web browser or other application that “tunnels” through the protocol—more on that in Chapter 15), along with SMTP, POP3, and IMAP (for sending and receiving e-mail on mobile devices).
Layer six	Presentation layer	Used to convert application data into acceptable and compatible formats for transmission. At this layer, data is encoded and encrypted. For example, audio, video, or image files transferred between systems might use MP3, MPEG4, or GIF encoding. Data compression (for example, with a Lempel-Ziv algorithm commonly used in Zip type file archiving) is also done at this layer. Network encryption is done at this layer as well.
Layer five	Session layer	Provides mechanisms for two hosts to maintain a network connection, or <i>session</i> , across a network. As long as a session is established, two hosts can continue to send data back and forth. This concept is important in the next chapter on firewalls, in the context of maintaining a session once it has been properly validated and accepted by the firewall policy configuration. NetBIOS is often classified as a session-layer protocol, and is SQL.
Layer four	Transport layer	Connects the upper OSI layers (five through seven) to the lower layers (one through three). The transport layer differentiates each application by assigning it a port number. These port numbers are familiar to most people in the context of “port 80” (for HTTP) or “port 53” (for DNS). Firewalls make access control decisions based on these port numbers (as discussed in the next chapter). TCP and UDP are the two most common transport-layer protocols. The main difference between the two is that TCP provides additional transmission services, such as ordered and reliable delivery, that UDP does not. Often, TCP is described as being <i>connection-oriented</i> , whereas UDP is described as being <i>connectionless</i> . TCP is used when an application must ensure that every packet is received, such as when transferring files. UDP is most appropriate when the resending of data is not needed or is not useful (especially over unreliable connections), such as with streaming video or voice applications.
Layer three	Network layer	Provides a unique address to every host on the network. Layer three also provides a means to connect layer one and two networks together using routers. IP is the most common layer three protocol in use worldwide. IP addresses are examples of layer three objects. IP (version 4) addresses consist of four groups of numbers between 0 and 255, like 192.168.0.1 or 10.1.55.223.
Layer two	Data-link layer	Composed of two different sublayers: Media Access Control (MAC) and Logical Link Control (LLC). The MAC is used to manage the sending of electrical signals across the physical medium with other hosts on the local segment. The LLC provides flow control, error checking, and synchronization. MAC addresses are 12 hexadecimal digits (usually grouped in pairs for easy readability), like 20-10-7a-3e-94-c7 or cc:af:78:bb:73:1d.
Layer one	Physical layer	Used to define and control electrical signals over the physical media. Ethernet is a commonly recognized example.

Table 14-1 The OSI Seven-Layer Protocol Stack

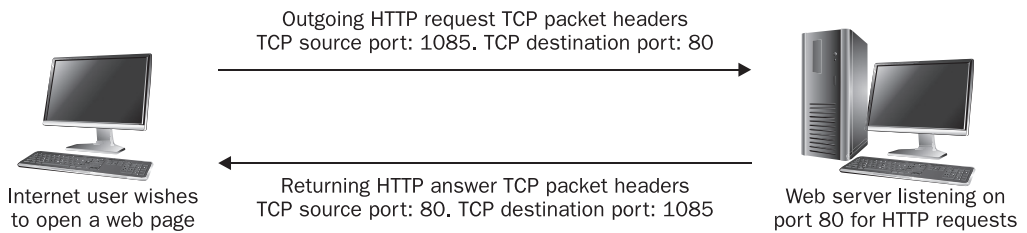


Figure 14-2 TCP port numbers in an HTTP request

being above 1,023 is not a requirement. Figure 14-2 shows how port numbers are used within TCP/IP packets. Source ports are necessary for the TCP/IP stack to connect the data received from the network to the application process that is requesting it. The application service/port combination creates a “socket” that the client and server use to communicate.

The list of TCP port numbers and the applications they are associated with is available in RFC 1700, “Assigned Numbers.” Table 14-2 lists some of the most popular services and their assigned ports.

Hubs

Hubs were dumb devices used to solve the most basic connectivity issue: how to connect more than two devices together. They transmitted packets between devices connected to

Service	Protocol	Port
FTP	TCP	21
FTP-data		20
SSH	TCP	22
Telnet	TCP	23
SMTP	TCP	25
DNS (zone transfers)	TCP	53
DNS (queries)	UDP	53
HTTP	TCP	80
NetBIOS	TCP UDP	137–139, 445
POP3	TCP	110
IMAP	TCP	143
SNMP	UDP/TCP	161
SNMP Traps	UDP	162
HTTPS	TCP	443

Table 14-2 Popular TCP and UDP Protocol Port Numbers

them, and they functioned by retransmitting each and every packet received on one port out through all of its other ports without storing or remembering any information about the hosts connected to them. This created scalability problems for legacy half-duplex Ethernet networks, because as the number of connected devices and volume of network communications increased, collisions became more frequent, degrading performance.

A collision occurs when two devices transmit a packet onto the network at almost the exact same moment, causing them to overlap and thus mangling them. When this happens, each device must detect the collision and then retransmit their packet in its entirety. As more devices are attached to the same hub, and more hubs are interconnected, the chance that two nodes transmit at the same time increases, and collisions became more frequent. In addition, as the size of the network increases, the distance and time a packet is in transit over the network also increases, making collisions even more likely. Thus, it is necessary to keep the size of such networks very small to achieve acceptable levels of performance.

Although most modern “hubs” offer 100-Mbps full-duplex or gigabit connectivity (there are no half-duplex connections in gigabit networks—the Gigabit Ethernet standard is always full duplex) to address the collision issue, and actually do perform some type of switching, the basic behavior of a hub still cannot address the scaling problem of a single broadcast domain. For that reason, hubs are rarely if ever seen anymore in enterprise network environments. Thus, we’ll say little more about them.

Switches

Switches are the evolved descendents of the network hub. From a network operation perspective, switches are layer two devices and routers are layer three devices (referring to their level of operation in the OSI stack), though as technology advances, switches are being built with capabilities at all seven layers of the OSI model, such as the UTM functions mentioned earlier.

Switches were developed to overcome the historical performance shortcomings of hubs. Switches are more intelligent devices that learn the various MAC addresses of connected devices and transmit packets only to the devices they are specifically addressed to. Since each packet is not rebroadcast to every connected device, the likelihood that two packets will collide is significantly reduced. In addition, switches provide a security benefit by reducing the ability to monitor or “sniff” another workstation’s traffic. With a hub, every workstation would see all traffic on that hub; with a switch, every workstation sees only its own traffic.

A switched network cannot absolutely eliminate the ability to sniff traffic. An attacker can trick a local network segment into sending it another device’s traffic with an attack known as *ARP poisoning*. ARP poisoning works by forging replies to ARP broadcasts. For example, suppose malicious workstation Attacker wishes to monitor the traffic of workstation Victim, another host on the local switched network segment. To accomplish this, Attacker would broadcast an ARP packet onto the network containing Victim’s IP address but Attacker’s MAC address. Any workstation that receives this broadcast would update its ARP tables and thereafter would send all of Victim’s traffic to Attacker. This ARP packet is commonly called a *gratuitous ARP* and is used to announce a new workstation attaching to the network. To avoid alerting Victim that something is wrong, Attacker would immediately forward any packets received for Victim to Victim. Otherwise Victim would soon wonder why network communications weren’t working. The most severe form of this attack is where the Victim is the local router interface. In this situation, Attacker would receive and monitor all traffic

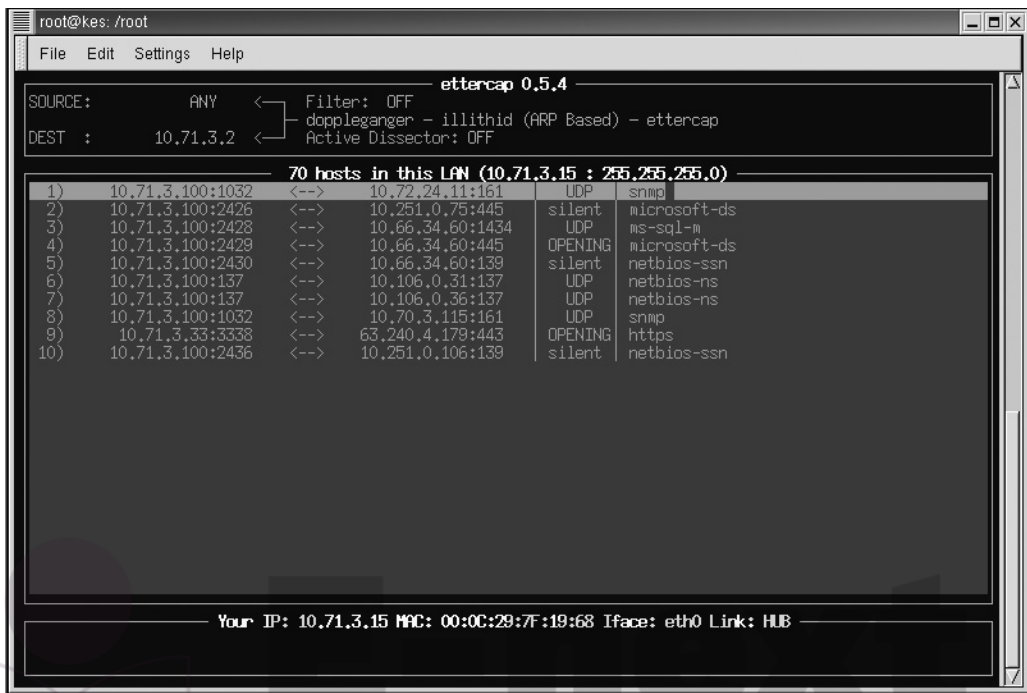


Figure 14-3 Ettercap spoofing the default gateway

entering and leaving the local segment. While ARP poisoning attacks appear complicated, there are several tools available that automate the attack process, such as Ettercap. Figure 14-3 shows an attacker using Ettercap to ARP poison the local segment's default gateway on a switched network.

To reduce a network's exposure to ARP poisoning attacks, segregate sensitive hosts between layer three devices or use *virtual LAN (VLAN)* functionality on switches. For highly sensitive hosts, administrators may wish to statically define important MAC entries, such as the default gateway. Statically defined MAC entries will take precedence over MAC entries that are learned via ARP. Statically defining ARP entries carries a high administrative burden and does not scale well, but can protect small networks that require high security. Before doing this, ensure that you determine whether any of the devices on your network use ARP spoofing for legitimate functional purposes, such as new host redirection to a captive portal, or for any HA functionality.

Routers

Routers operate at layer three, the network layer of the OSI model, and the dominant layer three protocol in use today is Internet Protocol version 4 (IPv4). Routers are primarily used to move traffic between different networks, as well as between different sections of the same network. Routers learn the locations of various networks in two different ways: dynamically via routing protocols and manually via administratively defined static routes. Networks usually use a combination of the two to achieve reliable connectivity between all necessary networks.

Static routes are required when a network can't or shouldn't be directly learned via a routing protocol. For example, to ensure that they aren't tricked into routing traffic to an attacker, firewalls typically do not run routing protocols. If a firewall is not informing the network of any networks behind it, those routes must be statically added to a network router and propagated. Additionally, static routes can be added for any interconnected network that cannot or does not communicate with the routing protocols on the network.

Controlling which devices can advertise routes for your network is an important security concern. Rogue or malicious routes in the network can disrupt normal communications or cause confidential information to be rerouted to unauthorized parties. While a number of routing protocols, such as Routing Information Protocol version 2 (RIPv2), Open Shortest Path First (OSPF), and the Border Gateway Protocol (BGP), can perform authentication, a common method is to disable or filter routing protocol updates on necessary router interfaces. For example, to disable routing updates on the first Ethernet interface of a Cisco router, issue the following command:

```
Router(config-router)#passive-interface ethernet 0
```

This is useful if no routing information should be received from or sent out this interface. However, this is not useful if some routing updates should be permitted and others blocked. When such a situation is encountered, *distribution lists* can be used. In the following example, routing updates for the router will be permitted inbound from the 10.108.0.0 network and outbound to the 10.109.0.0 network:

```
access-list 1 permit 10.108.0.0
access-list 2 permit 10.109.0.0
router rip
 network 10.108.0.0
  distribute-list 1 in
 network 10.109.0.0
  distribute-list 2 out
```

NOTE Cisco access lists all end with an implicit deny, meaning that all traffic that is not specifically allowed will be dropped when an ACL is applied.

Routing Protocols

There are two main types of layer three routing protocols: distance-vector protocols and link-state protocols (some proprietary protocols borrow mechanisms from both types, such as Cisco's Enhanced Internet Gateway Routing Protocol, or EIGRP, which Cisco calls an "advanced hybrid"). The main difference between the two types is in the way they calculate the most efficient path to the ultimate destination network.

Distance-vector protocols are more simplistic, are better suited for smaller networks (less than 15 routers), and require less CPU power on the devices that run them. Distance-vector protocols maintain tables of distances to other networks. Distance is measured in terms of *hops*, with each additional router that a packet must pass through being considered a hop. The most popular distance-vector protocol is the Routing Information Protocol (RIP).

Link-state protocols were developed to address the specific needs of larger networks. Link-state protocols use several different metrics to determine the best route to another network, and they maintain maps of the entire network that enable them to determine alternative and parallel routing paths to remote networks. Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (IS-IS) are examples of link-state protocols. Link-state protocols perform metric calculation and maintain databases of the entire network topology, and require significantly more CPU and memory capability than distance-vector protocols. As router hardware has evolved and more functions have been handled in silicon, such as in Cisco's Content Addressable Memory (CAM) and Ternary Content Addressable Memory (TCAM), a type of memory used by Cisco devices, even low-end routers can generally handle link-state routing (although many still have a limit on the number of routes they can handle).

For networks to function properly, all routers in a given network must maintain the same view or topology of the network, and the process by which routers come to agree upon the network topology is called *convergence*. Distance-vector and link-state protocols use different mechanisms to converge. The ability of a routing protocol to detect and respond to changes in network topologies is a significant advantage over the use of static routes.

However, when networks are unstable, such as just after a failure, or when network devices have different views of the topology, network routing loops can occur. A routing loop occurs when two routers decide that the best path to a given network is only available via each other, meaning that Router A believes the best route to a network is available via Router B, and at the same time Router B believes that the best route to the same network is only available via Router A. Thus, Router A will forward all packets received for that network to Router B, which will in turn forward them right back to Router A, preventing them from ever reaching their destination.

Each routing protocol has different mechanisms by which they detect and prevent routing loops. For example, a process called *split horizon* instructs RIP not to advertise a route on the same interface from which it learned the route. Another RIP mechanism is a hold-down timer, which instructs a router to not accept additional routing updates for a specified period. This is useful while the network is unstable immediately following a topology change.

Distance-vector protocols do not perform any proactive detection of their neighbors. They are configured to learn their directly connected neighbors and to periodically send and receive their entire routing tables to each other. Topology changes are detected when a router fails to receive a routing table from a neighbor during the required interval. Link-state protocols establish formal connections to their neighbors, and topology changes are automatically detected when a connection is lost. They also communicate regularly via keep-alive signals, making failure detection on the network much faster.

Although the choice of routing protocol does not have a large impact on network security, there are attacks including malformed packets and buffer vulnerabilities that can be directed at specific routing protocols. Research into the devices that will be used on the network is a good idea, including which code revisions will be used. And, as always, make sure that unneeded features are explicitly turned off. As mentioned, controlling where and with whom routing information is exchanged is a security best practice on a given network. When choosing a routing protocol, be sure it meets the needs of your anticipated network size, because once deployed, switching protocols can be a prohibitively expensive and time-consuming process. For high-security network devices, such as firewalls, it is more secure to define all routes statically, ensuring that the firewall is not vulnerable to a routing protocol attack.

Network Hardening

There are a number of configuration steps that you can take to ensure the proper operation of your routers and switches. These steps include applying patches as well as taking the time to configure the device for increased security. The more steps and time you take to patch and harden a device, the more secure it will be. The various steps that are available in most environments are detailed in the following sections.

Patching

Patches and updates released by the product vendor should be applied in a timely manner. Quick identification of potential problems and installation of patches to address newly discovered security vulnerabilities can make the difference between a minor inconvenience and a major security incident. To ensure you receive timely notification of such vulnerabilities, subscribe to your vendor's e-mail notification services, as well as to general security mailing lists. You will want to keep a special eye out for knowledge base (KB) articles and release notes, which describe changes in device behavior and default settings from one code version to another, in addition to specific vulnerabilities or code bugs being addressed. Ignoring these details can cause potential security issues on your network by negating previous steps you've taken to secure your devices.

Switch Security Practices

Network nodes are not directly aware that switches handle the traffic they send and receive, effectively making switches the silent workhorse of a network. Other than sometimes offering an administrative interface, layer two switches do not maintain layer three IP addresses, so hosts cannot send traffic to them directly. The primary attack against a switch is the ARP poisoning attack described earlier in the "Switches" section of this chapter. ARP poisoning attacks are also discussed in more detail in Chapter 2 and Chapter 17.

However, the possibility of an ARP attack doesn't mean switches cannot be used as security control devices. As mentioned earlier, MAC addresses are unique for every network interface card, and switches can be configured to allow only specific MAC addresses to send traffic through a specific port on the switch. This function is known as *port security*, and it is useful where physical access over the network port cannot be relied upon, such as in public kiosks. With port security, a malicious individual cannot unplug the kiosk, plug in a laptop, and use the switch port, because the laptop MAC will not match the kiosk's MAC and the switch would deny the traffic. While it is possible to spoof a MAC address, locking a port to a specific MAC creates a hurdle for a would-be intruder.

Switches can also be used to create *virtual local area networks (VLANs)*, layer two broadcast domains that are used to further segment LANs. As described earlier, ARP broadcasts are sent between all hosts within the same VLAN. To communicate with a host that is not in your VLAN, a switch must pass the host's packets through a layer three device and route them to the appropriate VLAN. Although there are some very specific exceptions to this rule for applications such as multicast (search the Web for "Multicast VLAN Registration" for details), in general, VLAN boundaries are helpful for containing and managing network segmentation, in addition to creating a foundation for applying differing levels of security to different networks based on the specific security needs.

Access Control Lists

Routers have the ability to perform IP packet filtering (packet filtering is discussed in detail in Chapter 15). Access control lists (ACLs) can be configured to permit or deny TCP, UDP, or other types of traffic based on the source or destination address, or both, as well as on other criteria such as the TCP or UDP port numbers contained in a packet. While firewalls are capable of more in-depth payload inspection, strategically placed router ACLs can significantly increase network security. For example, ACLs can be used on edge or border routers to drop obviously unwanted traffic (such as RFC 1918 traffic originating from a source on the Internet), removing the burden from the border firewalls. ACLs can also be used on WAN links to drop broadcast and other unnecessary traffic, thus reducing bandwidth usage.

Additionally, ACLs are often used to protect the router itself, and for other more advanced functions. It is a best practice to use an ACL to allow only the management stations or hosts on a network used by administrative staff authorized to log in to the network devices to connect to the administrative services (such as Telnet, SSH, or HTTP) on a router. Many vendors have unique functionality embedded in the ACL engines within their devices.

Not all ACLs are created equal; it is a good practice to understand a vendor's implementation and use of ACLs within its technology, as some specific features may be more or less desirable to networks performing different functions. A simple ACL in a Cisco router could be implemented with the following commands:

```
router(config)#access-list 101 deny tcp host 10.1.2.3 any eq www
router(config)#access-list 101 permit ip any any
```

This basic ACL tells the router to disallow HTTP sessions with a source address of 10.1.2.3 to all destinations. The second line of the ACL permits all other traffic.

To enforce this ACL, it must be applied to an interface with the `access-group` command:

```
router(config)#interface ethernet 0
router(config-if)#ip access-group 101 in
```

Disabling Unused Services

As with general-purpose operating systems, as discussed in Chapters 21 and 22, routers run services that are not required for the process of routing packets. Taking steps to disable and protect such services can increase the overall security of the network.

Proxy ARP

Proxy ARP allows one host to respond to ARP requests on behalf of the real host. This is commonly used on a firewall that is proxying traffic for protected hosts. Cisco routers have Proxy ARP enabled by default, and this may allow an attacker to mount an ARP poisoning attack against a host that is not on the local subnet or VLAN.

Network Discovery Protocols

There are several automatic discovery protocols, some of which are vendor specific, such as Cisco Discovery Protocol (CDP), others of which are open standard, such as Link Layer

Discovery Protocol - Media Endpoint Devices (LLDP-MED). In all cases, while these may provide some level of convenience for administering networks, they also present the opportunity for anyone sniffing the network to learn a significant amount of information about the network topology. If these protocols are not actively used, they should be disabled, and if they are used, careful attention should be paid to securing them as much as possible.

Other Extraneous Services

All routers provide a number of services that can be disabled if they are not needed. The following is a list of example services, but this is not intended to replace proper due diligence in your environment. Take the time to learn about which services can run on the devices in your network, whether or not they are turned on by default, and how to disable them or secure them from unauthorized access or use.

Diagnostic Services Most routers have a number of diagnostic services enabled for certain UDP and TCP services, including echo, chargen, and discard. These services should be disabled when not in use for troubleshooting or testing. Certain debug functions are particularly resource intensive, and an attacker could create a denial of service (DoS) condition simply by accessing a compromised router and turning on a debug process that consumes all of the available resources on the device. An administrator could also inadvertently create an outage in the same manner. Different vendors have different approaches to how much resources on a router these functions are allowed to use at any given time, including some with adjustable thresholds, and you may want to consider this when selecting network equipment.

BOOTP Server Routers can be used to provide DHCP addresses to clients through the BOOTP service. For small office/home office (SOHO) and residential setups, the router frequently is the DHCP server, but for enterprise, it is less common. If not in use, disable the unneeded service.

TFTP Server The Trivial File Transfer Protocol (TFTP) server can be used to simply transfer configuration files and software upgrades to and from the router. However, TFTP does not provide authentication or authorization services for its use. Most administrators run a TFTP server external to the router and enable it as needed.

Finger Server The finger service can be queried to see who is logged in to the router and from where. To disable this source of information leakage, disable finger.

Web Server Many vendors provide a web server for making configuration changes. If the router will not be managed in this manner, the web server can be disabled.

These services and several others pose security risks to the normal operation of the router while they are running. Different equipment manufacturers will have a variety of services that can potentially run on their devices, and it is important to understand what these are and which are really needed for operation. In many cases, security breaches can be avoided simply by learning about and following the best practice recommendations from a manufacturer about its equipment. When in doubt, turn off the services until they are needed.

Administrative Practices

Routers have a number of methods by which they can be managed. A command-line interface is accessible directly from a console or remotely via either Telnet or the Secure Shell protocol (SSH). SSH is recommended, as Telnet is sent over the network in cleartext, covered in more detail later in this chapter. Additionally, a web interface can be accessed via a browser, or the router can be monitored and managed via the Simple Network Management Protocol (SNMP). Beyond the normal aforementioned methods, there are other custom applications for some routers that allow you to download the configuration and manipulate it, compile it, and test that it is compatible and correct (sometimes called “well-formedness,” which is horrible English but technically correct) prior to uploading to the device. Securing each of these management protocols is of paramount importance, so they cannot be abused by attackers.

Another important step when hardening network devices is to configure a banner that is displayed whenever a connection is established as part of the login process, often called a login banner or message-of-the-day (MOTD) banner. In addition to ensuring that the banner doesn’t include any important information that may identify the device type or the operating system on the device, it is a good practice to include in the banner a warning message regarding unauthorized use of the device. This ensures that an individual cannot argue that they didn’t know that their use was unauthorized. In addition to these best practices, it is a good idea not to include details such as the physical location of the device or the name of the organization it belongs to—essentially, you want the banner to be a stern and clear warning, but otherwise as generic as possible. There is no good reason to offer a potential attacker any details that they could use for malevolent purposes or to support another component of an attack, such as social engineering.

Here is an example of boilerplate banner text that could be used to establish implied consent (meaning that if a user accesses the device via authorized or unauthorized means, their agreement to all of the language in the banner is implied). This example is similar, though not identical, to some standard language that the U.S. Department of Defense uses in its MOTD banners.

```
-----W A R N I N G-----
                        THIS IS A PRIVATE SYSTEM !!!

This system is provided only for authorized use. All systems may be monitored
for all lawful purposes, including ensuring that their use is authorized, for
management of the system, to facilitate protection against unauthorized access,
and to verify security procedures, survivability and operational security. During
monitoring, information may be examined, recorded, copied, and used for any
authorized purposes. All information including personal information, placed on
or sent over this system may be monitored. Any use of this system, authorized
or unauthorized, constitutes consent to monitoring of all activities performed
on this system. Unauthorized use may subject you to criminal prosecution.
Evidence of any such unauthorized use collected during monitoring may be used for
administrative, criminal or other adverse action. Use of this system constitutes
consent to monitoring for these purposes.
```

CAUTION Be careful to change default banners to remove any identifying information. By using information obtained from banners, such as the operating system version, attackers may identify relevant vulnerabilities inherent in the device and its firmware version to customize their attacks.

Remote Command Line

Telnet is a very old command-line protocol for remote connections, left over on many of today's devices from the very earliest days of multiuser computing. A weakness of the Telnet connectivity protocol is that it does not protect communications while they are in transit over the network (it does not use any encryption). As a more secure alternative, most routers support the Secure Shell (SSH) protocol. SSH provides the same interface and access as Telnet, but it encrypts all communications. Failure to encrypt administrative connections to network routers may allow an attacker to capture sensitive information, such as passwords and configuration parameters, while they are in transit over the network.

On many network devices, to enable SSH, it is necessary to configure host and domain names on the router, generate an encryption key, configure accounts, and set required SSH parameters. The commands to complete the configuration vary on a per-device basis, but if a network device you are considering using does not support this method or another encrypted management tool (such as forcing SSL only for web sessions), you should seriously consider whether or not the level of risk presented is allowable.

By default, many network devices maintain one password to access the device and a second password to access configuration commands, commonly called “privileged” or “enable” access. Even if this is not the default behavior, it can usually be configured. This is not true in all cases, depending on the manufacturer, but no matter what logo is on the device, you should understand the default account setup and how accounts are handled and permissioned. To provide granular authorization and full accountability, individual user accounts can and should be created, although not necessarily locally on the device; the approach and decision to use named accounts on a per-individual basis is much more important than where the accounts are created and stored. See the next section on AAA for more details on centralized accounts. Even if individual or generic local accounts will be used, be sure to change the passwords for any default accounts from their default values.

On some network devices, locally stored account information will be stored in cleartext unless otherwise configured. Under the hood, many network devices today use a commodity operating system, typically some flavor of Unix. Because of this, you need to pay special attention to making sure that the passwords are stored as a hashed value or in an encrypted file—otherwise, they may be easier to find than you may suspect.

Determining the type of encryption and methods used for locally stored passwords is relevant and should be understood prior to deploying devices into production, if for no other reason than to understand what a password recovery procedure might involve. It is entirely conceivable that being able to recover a password on a device could prove invaluable—such as in a scenario where an attacker finds a way to lock an administrator out of a critical piece of infrastructure, which can sometimes be done without actually gaining access to the device (too many failed attempts from a certain account, for example).

Centralizing Account Management (AAA)

In large-scale environments, it is cumbersome to synchronize and maintain individual user accounts on each network switch, router, and device. While it is possible to automate and

simplify local account management through scripting or tools, most network devices can be configured to authenticate against a central account repository via authentication, authorization, and accounting (AAA). This also helps remove usernames and passwords from local configurations (although having a backup local account with a more complex password is a best practice, when it is reserved only for emergencies). *Authentication* is the component that determines if an incoming connection is allowed, *authorization* determines what level of access or privilege the authenticated account is allowed, and *accounting* keeps track of everything that the authenticated and authorized account does.

Using the AAA methodology for device access is a best practice and can fully support auditability, an important consideration when trying to unravel what may have happened with too many hands in the cookie jar. While AAA is the mechanism, it is also critical to have a strong methodology around administrative device account creation and use policy. Requiring that administrators have a separate account specifically for administrative purposes will help protect critical network equipment, as you can take additional steps such as more frequent password rotation for admin accounts or more stringent password complexity requirements.

The only tangible downside to the approach (besides the overhead of maintaining one or more systems that perform the AAA functionality) is that if one of these elevated privilege accounts is compromised, an attacker would have access to things they may not have otherwise. However, this is less risky than using the same local username and password on all devices, because once the breach is discovered, you can disable the rogue account from a single place.

While there are many different access control server products available, the two most common protocols used for these access devices to perform device-level AAA communication are RADIUS and TACACS (now TACACS+). RADIUS is documented in RFC 2865, while TACACS+ is a Cisco-developed protocol. A newer protocol, *Diameter*, currently under development at IEEE, is aimed at replacing RADIUS. Defined in RFC 3588, Diameter will eventually become the newer AAA framework, with more advanced functionality than RADIUS or TACACS+ but essentially intended to solve the same basic problem. Because of its more advanced functionality and security, TACACS+ has been adopted or licensed for use by a variety of different equipment manufacturers and is currently the most common AAA protocol.

These AAA protocols constitute part of a framework and can achieve the same goal—secure AAA communication between an authenticator and endpoint. In addition to devices that use RADIUS and TACACS+, there are some devices that can directly query an LDAP or X.500 directory, but it is most common to find RADIUS or TACACS+ in use for communicating with network devices, even if some sort of directory like LDAP or Active Directory is the actual back-end database (many of the access control server products can map groups to an LDAP directory). Using the granular controls and variables offered by these protocols, administrative control can be finely tuned for specific needs.

While there is an administrative burden in building and maintaining the rules, it may be a worthwhile investment of time to implement command-level authorization sets built to guarantee that specific roles can access devices but cannot do certain things (such as enable a “debug all” or similar command). This also supports the best practices of a least-privilege model, and is a safer approach for critical environments that have many people who need to support them. A fair bit of knowledge is required to understand which commands are useful together and what level of administrator is capable of wielding them successfully, so adequate time needs to be invested in designing and testing this access if this approach will

be used. Keep in mind that, while you don't want security to be a compromise, there is a functional balance between too much access and not enough. Finding this balance for your specific environment will help maintain the actual risk level within the organization's acceptable risk tolerance.

Authentication to a network device should not rely completely on a remote authentication server. Should the server be down or unavailable, no one could log in. Therefore, keeping a local backup account is a good precautionary measure.

Beyond simply authenticating access to the router, it is good practice to limit the locations from which such connections can be initiated. For example, why permit Telnet or SSH sessions to the border routers from external networks, or to core routers from the entire internal network? ACLs are packet filters that will either accept or deny packets based on the packets' layer three header information, and can be employed to control the access management to the device itself.

When deciding where devices should be accessible from, apply the commonsense "sniff" test: would anyone ever need to log in to this device from that location for a legitimate reason? You may decide that you want to explicitly control a small number of locations on the network where devices are accessed from—this can be a very secure approach to securing administrative access, but it requires forethought and design to ensure that adequate access is preserved to manage the environment. Having only one workstation that is allowed to access network devices is highly secure, but not necessarily practical, and can carry its own set of risks if anything happens to that management station. Administrators can and should configure ACLs to restrict administrative access to authorized hosts and subnets, but adequate time should be spent designing this access prior to implementation to avoid creating risk by making the environment unmanageable.

Simple Network Management Protocol (SNMP)

Network devices can also be monitored and managed via Simple Network Management Protocol (SNMP), which provides a centralized mechanism for monitoring and configuration. SNMP can be used to monitor such things as link operation, port status and statistics, and CPU load via *Management Information Base Object Identifiers (MIB OIDs)*, a structured format database that describes objects within a device that can be monitored or managed by SNMP or another management protocol.

As SNMP has evolved as a tool, and its capabilities have expanded, its security has improved. The first version, SNMPv1, was originally released in 1988, and as with many other software protocols at the time, there was little consideration for the security of the protocol itself. A "community string" similar to a password for a built-in account was used for authenticating the protocol (with a well-known default value that was rarely changed), and all other protections were left to the configuration of the device. Also, depending on the configuration within the device, the community string could be used as Read-Only [RO] or Read-Write [RW], the second option offering a tempting vehicle for malicious intent by providing access to change, not just read, device settings.

SNMPv2 addressed many of the issues with SNMPv1, including performance, but added significant complexity. The most common flavor of SNMPv2 in use in the field is SNMPv2c. SNMPv3, the current version, doesn't change the protocol functionally but adds the capability of encryption, message integrity, and authentication of traffic.

SNMP can be a powerful tool to alert personnel to detected problems by sending traps to configured consoles. *Traps* are unsolicited messages that a device will send when a

configured threshold is exceeded or a failure occurs. SNMP consoles can be used to proactively monitor network devices and generate alerts if connectivity is lost or if other defined threshold conditions are violated. SNMP is the most prolific approach in use today for monitoring networks.

NOTE One very important step when configuring SNMP strings is to change them from their default values of “public” for Read-Only (RO) and “private” for Read-Write (RW).

Protecting SNMP communications can be done by configuring an ACL on each device to control which stations are allowed to query the device via SNMP, and what they are allowed to do. RW SNMP should be used only if there is specific functionality or automation that requires read-write access. Otherwise, for node managers that are only gathering statistics, use RO.

Historically, SNMP has posed significant security risks, partly because of the widespread use of version 1 and its lack of security capability, and partly because of implementation practices. SNMPv1 traffic, including authentication credentials, is not encrypted. Authentication consists of a community string, sent in cleartext over the network, and many implementations did not change the default strings from “public” for read access and “private” for write access.

Internet Control Message Protocol (ICMP)

The Internet Control Message Protocol (ICMP) provides a mechanism for reporting TCP/IP communication problems, as well as utilities for testing IP layer connectivity. It is an invaluable tool when troubleshooting network problems. However, ICMP can also be used to glean important information regarding network topologies and available host services.

ICMP was originally defined by RFC 792, but has since been updated by several other RFCs and is currently described by RFC 4884. Many different types of ICMP communications are defined, and are commonly referred to as messages. The following relevant ICMP functions present various risks when used for malicious purposes.

ECHO and Traceroute

Echo requests and replies, more commonly known as *pings*, are used to determine if another host is available and reachable across the network. If one host can successfully ping another host, it can be concluded that the hosts have proper network operation up to and including layer three of the OSI model. This does not guarantee that there are no other barriers or restrictions in place, but it does at least demonstrate reachability. While there are many cases where reachability exists but ping does not work (on a network where routers have been configured to drop ICMP messages, for example), it is a basic tool used by systems and network administrators to very quickly determine if a particular host is up or down.

An attacker can use ping to scan publicly accessible networks to identify available hosts, though more experienced attackers avoid ping and use more stealthy methods of host identification. Another use of ICMP echo and echo reply has been to create covert channels through firewalls that allow malicious traffic to pass through unchecked, under the assumption that nothing bad can be contained in an ICMP packet. ICMP echo requests and replies should be dropped at the network perimeter.

Traceroute is not itself an ICMP message type, but rather a method that frequently employs ICMP messages. It is also used to troubleshoot network-layer connectivity by mapping the network path between the source and destination hosts. Traceroute is useful in pinpointing where along the network path any connectivity troubles are occurring.

Traceroute works by sending out consecutive packets with the time to live (TTL) field incremented by one each time. When a network device routes a packet, it always decreases the TTL by 1. When a packet's TTL is decreased to zero, it is dropped, and an ICMP TTL Exceeded message is returned to the sender. This prevents packets from bouncing around networks forever. For example, a host can send out ICMP packets with TTLs of one, two, and three to identify the first three routers between itself and a destination.

In the hands of an attacker, TTL packets can be used to identify open ports in perimeter firewalls. Using this technique, attackers have devised a method for scanning networks using UDP, TCP, and ICMP packets that expire one hop beyond the perimeter firewall. The attack relies upon receiving ICMP TTL Exceeded messages from firewalled hosts, so dropping TTL Exceeded packets can defend against such attacks.

Additionally, there are situations where what a firewall doesn't reply with is as important as what it would reply with, and understanding the taxonomy of a covert attack can become very important when someone is probing your defenses. Knowing the difference between "drop" and "reject" when configuring ICMP message handling can make a big difference in how discoverable your network perimeter is from the outside. Research what options are available for this type of message handling on your external firewalls so that you can explicitly configure the behavior to suit your organization's needs.

Unreachable Messages

Another type of ICMP message is a Type 3 Destination Unreachable message. A router will return an ICMP Type 3 message when it cannot forward a packet because the destination address or service specified is unreachable. There are over 15 different types of codes that can be specified within the ICMP Destination Unreachable message, the more popular of which are outlined in Table 14-3.

Code	Message	Description
0	Network unreachable	The router does not have a route to the specified network.
1	Host unreachable	The host on the destination network does not respond to ARP.
2	Protocol unreachable	The layer four protocol specified is not supported through the router.
3	Port unreachable	The layer four protocol cannot contact a higher layer protocol specified in the packet.
4	Fragmentation needed	The size of the packet exceeds the maximum size allowed on the segment, but the packet's DO NOT FRAGMENT bit is set.
5	Source route failed	The next hop specified by the source route option is not available.
9 and 13	Communication administratively prohibited	A router has been configured to drop such communications to the destination host or network.

Table 14-3 ICMP Unreachable Code Types

NOTE While these messages may seem necessary for proper network operation, a malicious individual can use these message types to determine available hosts and services on the network. It is a good practice to drop all ICMP Destination Unreachable messages at the border of the network

There is an important consequence to dropping all Destination Unreachable messages. Code 4, Fragmentation Needed, is a very important message for proper network operation, and disruptions can occur if hosts cannot be informed that the packets they are sending into the network exceed the maximum transmission unit (MTU) of your network.

Directed Broadcasts

The first and last IP addresses of any given network are treated as being special. These addresses are known as the network address and the broadcast address, respectively. Sending a packet to either of these addresses is akin to sending an individual packet to each host on that network. Thus, someone who sends a single ping to the broadcast address on a subnet with 75 hosts could receive 75 replies.

This functionality has become the basis for a genre of attacks known as *bandwidth amplification attacks*. Examples of tools that use this attack are known as smurf and fraggle. In a *smurf attack*, the attacker sends ICMP traffic to the broadcast address of a number of large networks, inserting the source address of the victim. This is done so that the ICMP replies are sent to the victim and not the attacker. In a *fraggle attack*, the attacker also sends packets to large broadcast addresses in order to create a large number of responses, but this attack uses UDP ECHO packets instead of ICMP ECHO packets. The end result is the same. These UDP packets will generate responses from each system that is reachable and answering in the network range, and it can be more effective due to the behavior of certain services that use the UDP protocol. Modern firewalls can detect and defend against these types of attacks, but often rely on being configured to do so. Defending against these types of exploits should be a basic component of any firewall setup.

Redirects

ICMP redirects are used in the normal course of network operation to inform hosts of a more efficient route to a destination network. This is common on networks where multiple routers are present on the same subnet. However, a malicious user may be able to manipulate routing paths, and redirects should be disabled on router interfaces to untrusted and external networks.

Anti-Spoofing and Source Routing

Another type of attack used against networks is to insert fake or spoofed information in TCP/IP packet headers in the hopes of being taken for a more trusted host. Address spoofing is an attempt to slip through external defenses by masquerading as an internal host, and internal packets should obviously not be arriving inbound on border routers. Dropping such packets protects the network against such attacks, and border routers can be used to drop inbound packets containing source IP addresses matching the internal network. Additionally, routers should also drop packets containing source addresses matching RFC 1918 “private” IP addresses and broadcast packets.

In addition to spoofed packets, routers should be configured to drop packets that contain source routing information. Source routing is used to dictate the path that a packet should take through a network. Such information could be used to route traffic around known filters or to cause a denial of service situation by forcing large amounts of traffic through a single router, overloading it. There are times when an administrator would want to use source routing for testing or for specific technology applications, but this should be carefully evaluated on a per-case basis to determine if it is really necessary.

Logging

As with any device, it is a good idea to maintain logs for routers. Routers are able to log information related to ACL activity as well as system-related information. Most routers do not have large disks for locally logging information about network and system activity, but they do provide facilities for remote logging to a syslog server. In addition, the syslog facilities allow for the centralization and aggregation of all the dispersed network logs into a single repository. Syslog can become a critical component for troubleshooting something that happened on a network, or for performing forensics. While the logging host itself needs to be managed, an exercise to determine the right duration of archival logs should be performed when deploying a network; 30 or 60 days' worth of logs is common, although in some cases they are needed for longer.

Determining the right logging level (known as a “facility,” different levels of which dictate the verbosity and severity of the incident required to trigger a log action or trap) and for how long you will keep those logs gives you a window of time into the past, allowing a look back at what was going on in different places around the network at a given point in time. There is an advanced technology in the security industry, Security Information and Event Management (SIEM), which is described further in Chapter 18, dedicated to collecting, analyzing, and correlating logs and then either taking action or making action recommendations based on the event information.

Summary

Routers and switches provide a number of mechanisms that, when properly implemented, increase the overall security and performance of the local network. Designing the network to be segmented with VLANs or different subnets offers a way to create control boundaries between different areas of the network. Routers and many modern switches provide the ability to implement ACLs to screen and drop unwanted traffic. In addition, taking the time to harden the network devices against attacks will also increase the security of the network. This chapter also touched upon the various ICMP message types and the risks they pose, and covered SNMP for managing a network and syslog for logging. Proactive control of these tools can prevent an attacker from learning significant information about network topologies.

References

- Akin, Thomas. *Hardening Cisco Routers*. O'Reilly, 2002.
- Davis, Peter. *Securing and Controlling Cisco Routers*. Auerbach, 2002.
- Hogg, Scott, and Eric Vyncke. *IPv6 Security*. Cisco Press, 2008.

Jackson, Chris. *Network Security Auditing*. Cisco Press, 2010.

Liu, Dale. *Cisco Router and Switch Forensics: Investigating and Analyzing Malicious Network Activity*. Syngress, 2009.

McClure, Stuart, Joel Scambray, and George Kurtz. *Hacking Exposed: Network Security Secrets and Solutions*. 6th ed. McGraw-Hill, 2009.

McNab, Chris. *Network Security Assessment: Know Your Network*. 2nd ed. O'Reilly, 2007.

Schudel, Gregg, and David Smith. *Router Security Strategies: Securing IP Network Traffic Planes*. Cisco Press, 2008.

Vyncke, Eric, and Christopher Paggen. *LAN Switch Security: What Hackers Know About Your Switches*. Cisco Press, 2007.



E-next

THE NEXT LEVEL OF EDUCATION