Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

**Module**

**11**

**Security in Network Design**

* [Module Introduction](javascript://)
* **11-1**[Network Hardening by Design](javascript://)
  + **11-1a**[Router and Switch Security Configurations](javascript://)
  + **11-1b**[Switch Security Configurations](javascript://)
* **11-2**[Network Security Technologies](javascript://)
  + **11-2a**[Proxy Servers](javascript://)
  + **11-2b**[Firewalls](javascript://)
  + **11-2c**[IDS (Intrusion Detection System)](javascript://)
  + **11-2d**[IPS (Intrusion Prevention System)](javascript://)
  + **11-2e**[Cloud Security Technologies](javascript://)
* **11-3**[Authentication, Authorization, and Accounting (AAA)](javascript://)
  + **11-3a**[Authentication](javascript://)
  + **11-3b**[Authorization](javascript://)
  + **11-3c**[Accounting](javascript://)
* **11-4**[Authentication Technologies](javascript://)
  + **11-4a**[Directory Services](javascript://)
  + **11-4b**[Kerberos](javascript://)
  + **11-4c**[SSO (Single Sign-On)](javascript://)
  + **11-4d**[RADIUS (Remote Authentication Dial-In User Service)](javascript://)
  + **11-4e**[TACACS+ (Terminal Access Controller Access Control System Plus)](javascript://)
* **11-5**[Module Review](javascript://)
  + **11-5a**[Module Summary](javascript://)
  + **11-5b**[Key Terms](javascript://)
  + **11-5c**[Review Questions](javascript://)
  + **11-5d**[Hands-On Projects](javascript://)
  + **11-5e**[Capstone Projects](javascript://)

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# Module Introduction

### Objectives

After reading this module and completing the exercises, you should be able to:

* **1**Incorporate security into the design of a network
* **2**Describe the functions and features of various network security devices
* **3**Explain how authentication, authorization, and accounting work together to help secure a network
* **4**Compare authentication technologies

**On the Job**

I was the network administrator and entire IT department for a mortgage company owned by a bank, which I’ll call Bank A. Much of my job focused on certain legal and financial processes that had to run at different intervals, ranging from daily to quarterly. To facilitate these processes, I created static routes that allowed for the direct encrypted transfer of files from the mortgage company’s servers to Bank A’s servers.

Eventually, our company was purchased by another bank, which I’ll call Bank B. Naturally, many of our legal and financial processes had to change. This in turn necessitated numerous changes to the network. Working with Bank B’s IT department, I began updating Access Control Lists, and providing detailed information to our third-party Intrusion Prevention Service to make sure legitimate business was not accidently blocked. We tested firewalls, ACLs, VPNs, batch processes, static routes, and were confident that everything had been implemented successfully. It took coordination from three teams of people working over two weeks and weekends, but we got everything ready by the time the acquisition was announced. Everything went smoothly. The people at the mortgage company saw no changes to their work. The people at Bank B received all the files that they needed in the correct formats.

Fast-forward two months. I was working off-site, and my phone starts ringing nonstop. Some files needed to be transferred to a federal agency within hours or the mortgage company would be fined hundreds of thousands of dollars. No person at the mortgage company had permission or knowledge to investigate the issue. I had to pull off the interstate, find a coffee shop with Internet access, and get to work solving the problem.

Eventually we figured out the issue was a static route associated with a quarterly process that we had overlooked. I started by calling the people at Bank A to ask if they could detect any network traffic trying to reach a specific IP address. After some digging around, they found traffic from the mortgage company’s network being refused by a decommissioned server. To fix the problem, I configured new static routes and the files made it in on time by less than an hour.

**Johnathan Yerby, Ph.D.**

**Middle Georgia State University**

In the previous module, you began your study of network security with an exploration of threats to the network, physical security, and security policies for users. A non-technical network user will likely be exposed to all these things at some point in a non-IT career. In this module, you’ll see what security precautions IT professionals need to implement behind-the-scenes on a network to help keep it secure.

Let’s begin with a discussion of how to implement security into the very design of a network. Then you’ll read about network security devices, which is a category that includes far more than just firewalls. You’ll then explore the complementary processes of network access control and dig into various authentication protocols that validate the identity of people, services, and devices on your network.

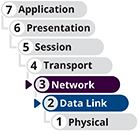
Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# 11-1Network Hardening by Design



### Certification

* 4.1

Explain common security concepts.

* 4.2

Compare and contrast common types of attacks.

* 4.3

Given a scenario, apply network hardening techniques.

* 5.5

Given a scenario, troubleshoot general networking issues.

Average reading time: 21 minutes

Throughout this course, you’ve learned about many security features and techniques used to secure software, devices, data, and traffic on a network. It’s easy to feel like your network is safe once you’ve implemented some of these strategies and created a secure perimeter around your valuable resources. However, network security professionals today can’t afford to be naïve. Attackers continue to develop their tactics and techniques to infiltrate networks, steal data, and damage resources. What’s outside the network has never been trusted, but now, what’s inside the network also can’t be trusted. This is called a [**zero trust**](javascript://) security model where everything in the network is considered untrustworthy until proven otherwise. Not only do routers at the edge of the network need firewall protection. Routers, switches, servers, and every other device inside the network are also at risk from malware, malicious users, intruders, bots, and even unintentional errors, misconfigurations, or failures. For these and many other reasons, security must be implemented in many, seemingly redundant layers that permeate the network and protect resources from every angle. As you’ve read in earlier modules, this strategy is called defense in depth.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-1aRouter and Switch Security Configurations

Networking devices such as routers and switches are designed to pass traffic through the network as quickly and efficiently as possible. However, opening these devices to unlimited traffic can cause problems, either for the devices themselves, for targets on the network, or for the network as a whole. Networking devices offer built-in features to help protect them from attacks and traffic spikes. In this section, you’ll read about security features configured on both routers and switches. In the next section, you’ll learn about security features specific to switches.

### Access Control Lists (ACLs)

Before a hacker can gain access to files on your network’s server, they must traverse one or more switches and routers. Although devices such as firewalls, described later in this module, provide more tailored security, manipulating switch and router configurations affords a small degree of security, especially when these devices sit on or near the edge of a network where they can control access to the network. This section describes a fundamental way to control traffic through routers, switches, and firewalls with the conversation focusing primarily on routers.

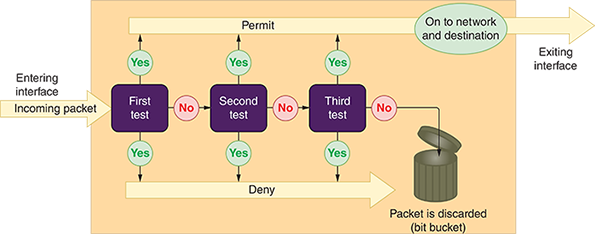
A router’s main function is to examine packets and determine where to direct them based on their network layer addressing information. Thanks to a router’s [**ACL (access control list)**](javascript://), or access list, routers can also decline to forward certain packets depending on their content. An ACL acts like a filter to instruct the router to permit or deny traffic according to one or more of the following variables:

* Network layer protocol (for example, IP or ICMP)
* Transport layer protocol (for example, TCP or UDP)
* Source IP address
* Destination IP address (which can restrict or allow certain websites)
* TCP or UDP port number

Each time a router receives a packet, it examines the packet and refers to its ACL to determine whether the packet meets criteria for permitting or denying travel on the network. See [Figure 11-1](javascript://). Each statement or test in the ACL specifies either a permit or deny flag. The router starts at the top of the list and makes a test based on the first statement. If a packet’s characteristics match a permit statement, the packet is released toward its destination. If the packet’s characteristics match a deny statement, the packet is immediately discarded. If the packet’s characteristics don’t match the first statement, the router moves down the list to the next statement in the ACL. If the packet does not match any criteria given in the statements in the ACL, the packet is dropped (as shown by the last “No” value in [Figure 11-1](javascript://)). This last decision is called the [**implicit deny**](javascript://) rule, which ensures that any traffic the ACL does not explicitly permit is denied by default.

**Figure 11-1**

A router uses an ACL to permit or deny traffic to or from a network it protects



Enlarge Image

On most routers, each interface must be assigned a separate ACL, and different ACLs may be associated with inbound and outbound traffic. When ACLs are installed on routers, each ACL is assigned a number or name.

The access-list command is used to assign a statement to an ACL on Cisco routers and similar routers. The command must identify the ACL and include a permit or deny argument. Here are a few sample commands used to create statements in the ACL named acl\_2, which controls incoming traffic to a router:

* To permit ICMP traffic from any IP address or network to any IP address or network:

A sample command used to create a statement in the access control list of a router to permit I C M P traffic from any I P address or network to any I P address or network. access hyphen list a c l underscore 2 permit i c m p any any.

* To deny ICMP traffic from any IP address or network to any IP address or network:

A sample command used to create a statement in the access control list of a router to deny I C M P traffic from any I P address or network to any I P address or network. access hyphen list a c l underscore 2 deny i c m p any any.

* To permit TCP traffic from 2.2.2.2 host machine to 5.5.5.5 host machine:

A sample command used to create a statement in the access control list of a router to permit T C P traffic from the host machine with the I P address 2 dot 2 dot 2 dot 2 to the host machine with the I P address 5 dot 5 dot 5 dot 5. access hyphen list a c l underscore 2 permit t c p host 2 dot 2 dot 2 dot 2 host 5 dot 5 dot 5 dot 5.

* To permit TCP traffic from 2.2.2.2 host machine to 3.3.3.3 host machine to destination web port 80 (the “eq” parameter says “equal to” and “www” is a keyword that stands for port 80):

A sample command used to create a statement in the access control list of a router to permit T C P traffic from the host machine with the I P address 2 dot 2 dot 2 dot 2 to web port 80 of the host machine with the I P address 3 dot 3 dot 3 dot 3. access hyphen list a c l underscore 2 permit t c p host 2 dot 2 dot 2 dot 2 host 3 dot 3 dot 3 dot 3 e q w w w.

Enlarge Image

Statements can also specify network segments (groups of IP addresses) by using a network address for the segment and a wildcard mask. The bits in a wildcard mask work opposite of how bits in a subnet mask work. The 0s in the wildcard mask say to match the IP address bits to the network address given, and the 1s say you don’t care what the value of those bits are. For example, a wildcard mask of 0.0.0.255 can be written as 00000000.00000000.00000000.11111111, which says the first three octets of an IP address must match the given network address and the last octet can be any value. For example, the following command permits TCP traffic to pass through when the first three octets of an IP address are 10.1.1 and the last octet can be any value:

The command that permits T C P traffic to pass through when the first three octets of an I P address are 10 dot 1 dot 1 and the last octet can be any value. access hyphen list a c l underscore 2 permit t c p 10 dot 1 dot 1 dot 0 0 dot 0 dot 0 dot 255.

**Note 11-1**

In ACL statements, any is equivalent to using a wildcard mask of 255.255.255.255, which allows all IP addresses to pass through.

An access list is not automatically installed on a router. If you don’t configure an ACL, the router allows all traffic through. Once you create an ACL and assign it to an interface, you have explicitly permitted or denied certain types of traffic. Naturally, the more statements or tests a router must scan (in other words, the longer the ACL), the more time it takes a router to act and, therefore, the slower the router’s overall performance.

When troubleshooting a problematic connection between two hosts, or between some applications or ports on two hosts, consider that the problem might be a misconfigured ACL blocking needed services, ports, or addresses. For example, suppose you can successfully ping a host, but Telnet and tracert attempts cannot connect with the same host. You can use a process of elimination on the device’s various ACLs to identify the incorrect ACL settings and correct the problem. Common errors include listing the ACL statements in the wrong order, using the wrong criteria when defining a rule, or constructing a rule incorrectly.

### Control Plane Policing (CoPP)

As you know, the control plane refers to the decision-making layer of connected networking devices. For example, a router’s control plane manages routing protocols to build and constantly update routing tables. This functionality can become overwhelmed in times of high traffic or during an attack. While ACLs filter traffic into and out of router interfaces on the data plane, the control plane needs a separate layer of protection. In this case, an adaptation of QoS (quality of service) filters can be used to rate-limit traffic on the control plane and management plane of routers and switches using a feature called **[CoPP (control plane policing)](javascript://)**.

The following steps describe the process of configuring CoPP on a switch or router. In this scenario, ICMP traffic is permitted with no limits from one trusted device at 192.168.2.2. All other ICMP traffic is limited and, when exceeding that limit, is dropped. You might have the opportunity to test these steps in the lab on a real router. However, Packet Tracer does not support some of the commands required here.

Begin by defining an ACL that will identify which traffic is relevant to your CoPP policies. For example, the following commands, entered in global configuration mode, create an ACL numbered 100 that defines the relevant ICMP traffic:

A command entered in global configuration mode that creates an A C L numbered 100 that defines the relevant I C M P traffic. Line 1: access hyphen list 100 permit i c m p any any. Line 2: access hyphen list 100 deny i c m p host 192 dot 168 dot 2 dot 2 any.

Next, you need to create a class map, which will classify traffic according to defined criteria such as an ACL. Use the following command to create a class map named limit-icmp and to enter class-map configuration mode (which is similar in concept to the interface configuration mode you’ve used in some of your projects):

The comand to create a class map named limit hyphen i c m p and to enter class hyphen map configuration mode. class hyphen map limit hyphen i c m p.

Now you can match the class to your identified traffic with the following command:

The command to match the class to the identified traffic. match access hyphen group 100.

Exit class-map configuration mode with the exit command. You now need a policy map, which will apply policies to the traffic identified by the class map. Use the following command to create a policy map named copp-test and to enter pmap configuration mode:

The command to create a policy map named copp hyphen test and to enter p map configuration mode. policy hyphen map c o p p hyphen test.

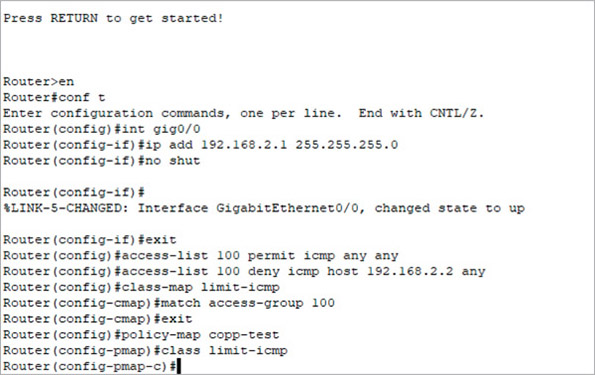
The class command entered in pmap configuration mode will pair the class map you created earlier to your new policy map, as follows:

The class command entered in p map configuration mode that pairs the class map created earlier to the new policy map. class limit hyphen i c m p.

This command also takes you into the pmap-class configuration mode, as shown in [Figure 11-2](javascript://).

**Figure 11-2**

Pmap-class configuration mode



Enlarge Image

Source: Cisco Systems, Inc.

Next, the police command will define actions to perform when these criteria are met. For example, if the bps (bits per second) rate stays below the threshold of 8000, the identified messages will be transmitted. However, if the rate exceeds 8000 bps, packets are dropped. The following command in pmap configuration mode defines this scenario:

The police command the defines the actions to perform when the following criteria are met. The message needs to be transmitted if the bits per second rate stays below the threshold of 8000. If this threshold is crossed, the packets need to be dropped. police 8000 conform hyphen action transmit exceed hyphen action drop.

You now need to shift to control plane configuration mode. From global configuration mode, enter the following command:

The command to shift to control plane configuration mode from global configuration mode. control hypen plane.

Finally, apply the QoS service policy you created earlier to the control plane on the device with the following command:

The command to apply the Quality of Service policy created earlier to the control plane on the device. service hyphen policy input c o p p hyphen test.

After all that work, you can test your configuration by running a lot of large pings from a trusted source and from an untrusted source to compare the outcomes. If you did it all correctly, the untrusted source’s pings will sometimes be dropped to enforce the defined maximum ICMP bits per second on the router’s control plane, while the trusted source’s pings will not be limited.

Go to pg.

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-1bSwitch Security Configurations

Some security features, such as CoPP, work on both routers and switches. Some features, however, are specific to switches due to the types of traffic switches manage. Consider the following security features on switches.

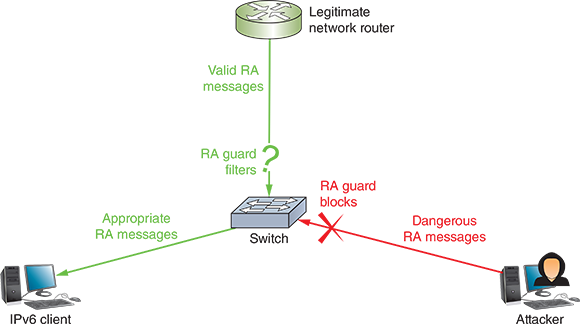
### Router Advertisement (RA) Guard

Default trust relationships between one network device and another might allow a hacker to access the entire network because of a single flaw. For example, recall from earlier in this course you learned how IPv6 clients receive RA (router advertisement) messages to determine network information, such as IP addresses of DNS servers and the default gateway as well as the network prefix. One weakness of this system is that IPv6 clients are a bit gullible—if several devices send RA messages, clients don’t know which advertisements to believe (see [Figure 11-3](javascript://)). Clients have no way of authenticating RA messages to know which of these messages come from legitimate sources and which might come from a network attacker. This creates two vulnerabilities:

* Malicious RA messages can be used to misconfigure network clients, thus hijacking network traffic.
* High volumes of RA messages can create a network DoS attack, slowing or disabling the network. In a project at the end of this module, you’ll practice using RA messages to flood and disable a virtual network.

**Figure 11-3**

An attacker sends illegitimate RA messages



The solution to this problem is to configure RA guards on network switches. The [**RA guard**](javascript://) feature filters RA messages so these messages can only come from specific interfaces on the switch. Additionally, RA guard offers other criteria that can filter RA messages on valid interfaces according to source MAC or IP address, router priority, or other options.

RA guard is configured on Cisco switches using the raguard command, which accesses the RA guard policy configuration mode (similar to other configuration modes you’ve used or read about here). You might create one RA guard policy for hosts attached to the switch, and this policy might be named something like HOSTS. The HOSTS policy blocks all RA messages for interfaces with that policy applied. You might create a second RA guard policy for routers attached to the switch, and this policy might be named something like ROUTERS. The ROUTERS policy might filter RA messages to ensure they’re coming from a trusted router.

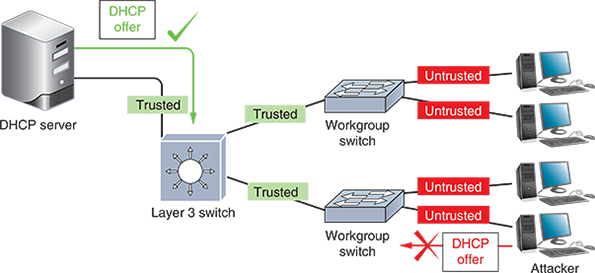
### DHCP Snooping

Similar to RA messages, by default, DHCP messages are allowed to flow freely through ports on switches so that clients can request and receive DHCP assignments. A [**rogue DHCP server**](javascript://) running on a client device, however, could be used to implement an on-path attack by configuring the attacker’s IP address as the victim computers’ default gateway. Alternatively, the attacker could give their IP address as the DNS server and then spoof websites.

DHCP messages should be monitored by enabling [**DHCP snooping**](javascript://) on a switch. This way, switch ports connected to clients won’t be allowed to transmit DHCP responses that should only come from a trusted DHCP server. In [Figure 11-4](javascript://), you can see the layer 3 switch trusts the DHCP offer made by the DHCP server, and this offer can be forwarded to the workgroup switches. A DHCP offer from the attacking computer on the bottom right of the figure, however, will not be trusted.

**Figure 11-4**

DHCP offer messages can only enter a trusted port on a switch, not an untrusted port

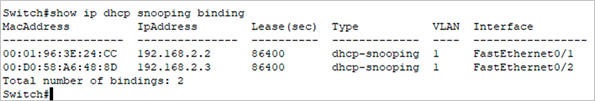


Enlarge Image

DHCP snooping is configured on a Cisco switch using the ip dhcp snooping command. When enabled, the switch snoops, or listens to, DHCP messages exchanged between the network’s DHCP server and its clients. The switch then gathers legitimate IP-to-MAC address pairings as assigned by DHCP and collects this information in a DHCP snooping binding database. [Figure 11-5](javascript://) shows a sample database from a switch where two PCs have received DHCP assignments from a legitimate DHCP server. The switch can use this information for other layers of security, as you’ll read about next. In a project at the end of this module, you’ll attack a network with a rogue DHCP server and then defend the network using DHCP snooping.

**Figure 11-5**

This switch collected IP-to-MAC address pairing information from eavesdropping on DHCP messages



Enlarge Image

Source: Cisco Systems, Inc.

### Dynamic ARP Inspection (DAI)

You learned in an earlier module that an attacker can redirect Internet or other network traffic from a legitimate server by altering DNS records in clients’ and servers’ caches. Similar to DNS caches, ARP tables can also be altered. Recall that ARP works in conjunction with IPv4 to discover the MAC address of a node on the local network. This information is stored in the ARP table or ARP cache, which maps IP addresses to MAC addresses on the LAN. However, ARP performs no authentication, and so, it is highly vulnerable to attack. As a result, attackers can use [**ARP spoofing**](javascript://), also called ARP poisoning or ARP cache poisoning, to send faked ARP replies that alter ARP tables in the network.

ARP vulnerabilities contribute to the feasibility of several other exploits, including DoS attacks, on-path attacks, and MAC flooding. MAC flooding is when an attacker floods a switch with fictitious MAC addresses. The switch accepts these fake MAC addresses and fills its MAC address table, which means it then must remove valid MAC addresses to keep processing the incoming information. Once its MAC address table is full of illegitimate information, legitimate network clients can no longer communicate except through broadcast transmissions, further slowing the network. Similar to the way you protect against a broadcast storm, port security on switches can reduce or eliminate the risks of MAC flooding. But what about other types of ARP spoofing?

[**DAI (dynamic ARP inspection)**](javascript://) can be configured on a switch to protect against ARP spoofing attacks. When DAI is enabled, the switch compares incoming messages with its DHCP snooping binding table to determine whether the message’s source IP address is appropriately matched with its source MAC address according to DHCP assignments on the network. DAI and DHCP snooping work together on network switches.

**Applying Concepts 11-1**

### Explore Computer Forensics Investigations

Now that you understand a bit more about security in network design, you’re ready to learn about how computer crimes are committed and discovered. As a network technician, you’ll be better prepared to spot security issues if you’re already familiar with breaches that have affected other networks in the past. In this activity, you research three computer forensics investigations. Use complete sentences, good grammar, and correct spelling in your answers. Complete the following steps:

1. 1

Using a search engine, find articles, blogs, or videos discussing three different computer forensics cases. Identifying information might have been changed to protect privacy, but be sure the cases are actual cases, not just theoretical ones. Document your source or sources for each case.

1. 2

Answer the following questions for each case:

* + How was the problem discovered?
  + What clues initiated the investigation?
  + What crime was committed or suspected?
  + What evidence was collected using computer forensics?
  + Were there any significant mistakes made in collecting this evidence?
  + What was the final outcome of each case?

**Remember This…**

* Explain zero trust and defense in depth security concepts.
* Compare technology-based attacks, including ARP spoofing and rogue DHCP.
* Apply common security best practices, including ACLs, control plane policing, RA guard, DHCP snooping, and DAI.

**Self-Check**

1. Which ACL rule will prevent pings from a host at 192.168.2.100?

Answer

* 1. access-list acl\_2 permit icmp any host 192.168.2.100
  2. access-list acl\_2 deny icmp host 192.168.2.100 any
  3. access-list acl\_2 deny tcp host 192.168.2.100 host 192.168.2.1
  4. access-list acl\_2 deny icmp any host 192.168.2.100

1. Which two features on a switch or router are integrated into CoPP? Choose two.

Answer

* 1. ICMP
  2. DHCP
  3. QoS
  4. ACLs

1. Which of the following defenses addresses a weakness of IPv6?

Answer

* 1. DHCP snooping
  2. CoPP
  3. DAI
  4. RA guard

**You’re Ready**

You’re now ready to complete [Project 11-1: Configure ACLs in Packet Tracer](javascript://), or you can wait until you’ve finished reading this module.

**You’re Ready**

You’re now ready to complete [Project 11-2: Configure DHCP Snooping in Packet Tracer](javascript://), or you can wait until you’ve finished reading this module.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# 11-2Network Security Technologies

### Certification

* 1.8

Summarize cloud concepts and connectivity options.

* 2.1

Compare and contrast various devices, their features, and their appropriate placement on the network.

* 2.3

Given a scenario, configure and deploy common Ethernet switching features.

* 3.3

Explain high availability and disaster recovery concepts and summarize which is the best solution.

* 4.1

Explain common security concepts.

* 4.3

Given a scenario, apply network hardening techniques.

* 5.2

Given a scenario, troubleshoot common cable connectivity issues and select the appropriate tools.

* 5.5

Given a scenario, troubleshoot general networking issues.

Average reading time: 34 minutes

Many devices on a network serve non-security purposes and yet are outfitted with significant security features and abilities. Others are designed specifically with network security in mind. In this section, you’ll read about many devices that work together to build a secure net around and throughout a network. Using multiple options for network security results in layered security, or defense in depth, as you saw in the [On the Job](javascript://) story at the beginning of this module. This approach provides more protection than any one type of device or defense can provide on its own. Let’s look at how each of these components contributes to security in network design.

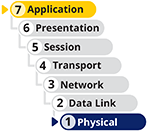
Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-2aProxy Servers



One technique for enhancing network security is adding a proxy server. A [**proxy server**](javascript://), or proxy, acts as an intermediary between the external and internal networks, screening all incoming and outgoing traffic. Proxy servers manage security at the application layer of the OSI model. Although proxy servers only provide low-grade security relative to other security devices, they can help prevent an attack on internal network resources such as web servers and web clients.

To understand how proxies work, think of the secure data on a server as the president and the proxy server as the secretary of state. Rather than have the president risk his safety by leaving the country, the secretary of state travels abroad, speaks for the president, and gathers information on the president’s behalf. In fact, foreign leaders may never actually meet the president. Instead, the secretary of state acts as the president’s proxy. In a similar way, a proxy server represents a private network to another network (usually the Internet).

Although a proxy server appears to the outside world as an internal network server, in reality it is merely another filtering device for the internal LAN. One of its most important functions is preventing the outside world from discovering addresses on the internal network. For example, suppose your LAN uses a proxy server, and you want to send an email message from your workstation inside the LAN to a colleague via the Internet. The following steps describe the process:

1. Step 1

Your message goes to the proxy server. Depending on the configuration of your network, you might or might not have to log on separately to the proxy server first.

1. Step 2

The proxy server repackages the data frames that make up the message so that, rather than your workstation’s IP address being the source, the proxy server inserts its own IP address as the source.

1. Step 3

The proxy server passes your repackaged data to a packet-filtering firewall, which you’ll learn more about later in this module.

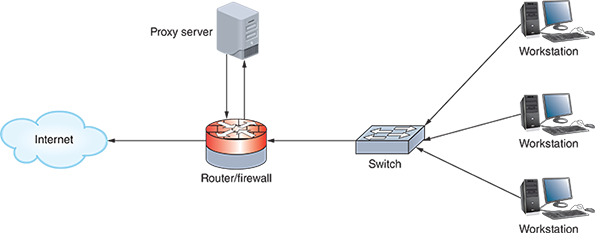
1. Step 4

The firewall verifies that the source IP address in your packets is valid (that it came from the proxy server) and then sends your message to the Internet.

Proxies are often used by enterprise networks to protect internal network clients. However, individuals sometimes rely on proxy servers to mask their Internet activities. In this case, users might want to circumvent traffic restrictions (such as accessing resources in a different political domain) or they might want to maintain anonymity. Examples of proxy server software include Smartproxy ([smartproxy.com](http://smartproxy.com/" \t "_blank)), Luminati ([luminati.io](http://luminati.io/" \t "_blank)), Squid ([squid-cache.org](http://squid-cache.org/" \t "_blank)), and, for Windows only, WinGate by Qbik ([wingate.com](http://wingate.com/" \t "_blank)), which includes firewall features as well. [Figure 11-6](javascript://) depicts how a proxy server might fit into a corporate network design.

**Figure 11-6**

A proxy server is used to connect to the Internet



Enlarge Image

Although proxies might sound similar to VPNs, there are critical differences. For example, a VPN encrypts traffic, while a proxy does not. You also might have noticed that proxy services sound suspiciously similar to NAT, which you learned about earlier. However, they differ significantly. You’ve already learned that proxy servers can provide some content filtering, which is possible because they function at the application layer rather than at the lower, network layer. Proxy servers can also improve performance for users accessing resources external to their network by caching files. For example, a proxy server situated between a LAN and an external web server can be configured to save recently viewed web pages. The next time a user on the LAN wants to view one of the saved web pages, content is provided by the proxy server. This eliminates the time required to travel over a WAN connection and retrieve the same content multiple times from the external web server. Essentially, the difference is that NAT is an addressing construct to better manage IP addressing schemes, while proxies are security devices to mediate traffic between servers and clients.

Whereas proxy servers access resources on the Internet for a client, a reverse proxy provides services to Internet clients from servers on its own network. In this case, the reverse proxy provides identity protection for the server rather than the client, as well as some amount of application layer firewall protection. Reverse proxies are particularly useful when multiple web servers are accessed through the same public IP address.

**Note 11-2**

Often, firewall and proxy server features are combined in one device. In other words, you might purchase a firewall that can block certain types of traffic from entering your network (a firewall function) and also modify the addresses in the packets leaving your network (a proxy function).

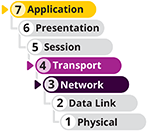
Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-2bFirewalls

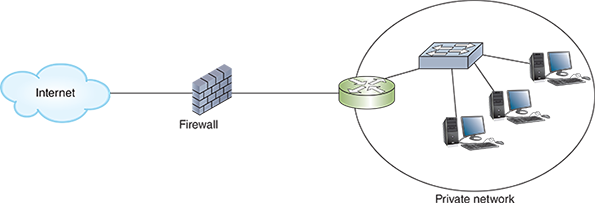


A firewall is a specialized device or software that selectively filters or blocks traffic between networks. A firewall protects a network by blocking certain traffic from traversing the firewall’s position, similar to a bouncer checking IDs at the entrance to a private club. While firewalls include filtering from ACLs, they also offer a wide variety of other methods to evaluate, filter, and control network traffic.

A firewall might be placed internally, residing between two interconnected private networks. More commonly, the firewall is placed on the edge of the private network, monitoring the connection between a private network and a public network (such as the Internet), as shown in [Figure 11-7](javascript://). This is an example of a [**network-based firewall**](javascript://), so named because it protects an entire private network. [Figure 11-8](javascript://) shows dedicated firewall appliances that might be purchased for a medium-sized or large corporation’s network. You’ll also see firewall features integrated into routers, switches, and other network devices.

**Figure 11-7**

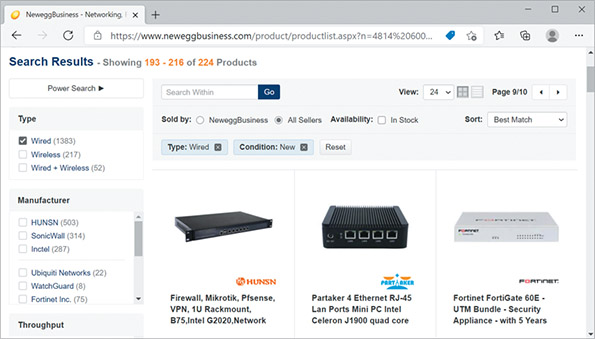
Placement of a firewall between a private network and the Internet



Enlarge Image

**Figure 11-8**

Dedicated firewall devices



Enlarge Image

Source: Newegg Business, Inc.

In contrast, [**host-based firewall**](javascript://) software only protects the computer on which it’s installed. These firewalls can be configured more specifically for each server or workstation, whereas network firewalls must be configured for all the traffic allowed on a network. For example, a network firewall might allow SSH on port 22 because several devices on the network must support remote access. In contrast, the firewall on a server that should never be accessed remotely should not allow traffic on port 22. Network firewalls offer the ability to configure much more complex rules and filters, while host firewalls offer protection specific to a single device.

The simplest form of a firewall is a [**packet-filtering firewall**](javascript://), which is a network device or application that examines the header of every packet of data it receives on any of its interfaces (called inbound traffic), as shown in [Figure 11-9](javascript://). The firewall refers to its ACL to determine whether that type of packet is authorized to continue to its destination, regardless of whether that destination is on the internal LAN or on an external network. If a packet does not meet the filtering criteria, the firewall blocks the packet from continuing. However, if a packet does meet filtering criteria, the firewall allows that packet to pass through to the network the firewall protects. This is a common feature of SOHO routers, and in fact, nearly all routers can be configured to act as packet-filtering firewalls via ACLs.

**Figure 11-9**

Arrows pointing toward the firewall are inbound for that device



Enlarge Image

[Figure 11-9](javascript://) shows the firewall filtering traffic that comes into the LAN from the Internet, and it also filters traffic that goes out of the LAN to the Internet. One possible reason for blocking Internet-bound traffic is to stop worms from spreading. For example, if you’re running a web server, which in most cases only needs to respond to incoming requests and does not need to initiate outgoing requests, you could configure a packet-filtering firewall to block certain types of outgoing transmissions initiated by the web server. In this way, you help prevent spreading worms that are designed to attach themselves to web servers and propagate themselves to other computers on the Internet.

Often, firewalls ship with a default configuration designed to block the most common types of security threats. In other words, the firewall may be preconfigured to accept or deny certain types of traffic. However, many network administrators choose to customize the firewall settings, for example, blocking additional ports or adding criteria for the type of traffic that may travel into or out of ports. Some common criteria by which a packet-filtering firewall might accept or deny traffic include the following:

* Source and destination IP addresses
* Source and destination ports (for example, ports that supply TCP/UDP connections, FTP, Telnet, ARP, ICMP, and so on)
* Flags set in the TCP header (for example, SYN or ACK)
* Transmissions that use the UDP or ICMP protocols
* A packet’s status as the first packet in a new data stream or a subsequent packet
* A packet’s status as inbound to or outbound from your private network

Based on these options, a network administrator might configure a firewall to prevent any IP address that does not begin with 10.121, the network ID of the addresses on the local network, from accessing the network’s router and servers. Furthermore, the admin might disable—or block—certain well-known ports, such as the insecure NetBIOS ports (137, 138, and 139). Blocking ports prevents any user from connecting to and completing a transmission through those ports. This technique is useful to further guard against unauthorized access to the network. In other words, even if a hacker could spoof an IP address that began with 10.121, they could not access NetBIOS ports (which are notoriously insecure) on the other side of the firewall.

**Note 11-3**

Ports can be blocked not only on firewalls, but also on routers, servers, or any device that uses ports. For example, if you established a web server for testing but did not want anyone in your organization to connect to your test web pages through a browser, you could block port 80 on that server. Be careful, however, when opening or blocking ports used by multiple protocols or types of connections, such as SSH’s port 22. An incorrectly configured firewall is an easy thing to overlook when, for example, troubleshooting a newly installed application on a host.

For greater security, you can choose a firewall that performs more complex functions than simply filtering packets. When shopping for firewalls (or other devices), be sure to consider which features are included for the licenses you plan to purchase, as this can affect which features you can actually use. Among the factors to consider when making your decision are the following:

* Does the firewall support encryption?
* Does the firewall support user authentication?
* Does the firewall allow you to manage it centrally and through a standard interface?
* How easily can you establish rules for access to and from the firewall?
* Does the firewall provide internal network logging and auditing capabilities, such as IDS or IPS? IDS and IPS are described later in this module.
* Does the firewall protect the identity of your internal LAN’s addresses from the outside world?
* Can the firewall monitor packets according to existing traffic streams? A [**stateful firewall**](javascript://) can inspect each incoming packet to determine whether it belongs to a currently active connection (called a stateful inspection) and is, therefore, a legitimate packet. A [**stateless firewall**](javascript://) manages each incoming packet as a stand-alone entity without regard to currently active connections. Stateless firewalls are faster than stateful firewalls but are not as sophisticated.
* Does the firewall support filtering at the highest layers of the OSI model, not just at the network and transport layers? [**Application layer firewalls**](javascript://) can block designated types of traffic based on application data contained within packets. For example, a school might configure its firewall to prevent responses from a website with questionable content from reaching the client that requested the site. To do this, the firewall must examine the payloads of messages to determine whether these messages are attempting to communicate with a blacklisted website. An application layer firewall can also examine words, phrases, or code contained within that payload, which allows the network to monitor internal traffic for signs of infiltration, malware, or data theft. This capability supports a zero trust environment where messages inside the network aren’t automatically trusted simply because they’re inside the network.

A SOHO wireless router typically acts as a firewall and includes packet-filtering options. At the other end of the spectrum, devices made by Cisco or Fortinet for enterprise-wide security are known as security appliances and can perform several functions, such as encryption, load balancing, and IPS, in addition to packet filtering. Examples of software that enable a computer to act as a packet-filtering firewall include iptables (a command-line firewall utility for Linux systems), ZoneAlarm ([zonealarm.com](http://zonealarm.com/" \t "_blank)), and Comodo Firewall ([comodo.com](http://comodo.com/" \t "_blank)). Some operating systems, such as Windows 10, include firewall software. You’ll explore Windows Firewall in the next section. In a project at the end of this module, you’ll practice using iptables in Ubuntu Server.

**Applying Concepts 11-2**

### Windows Defender Firewall

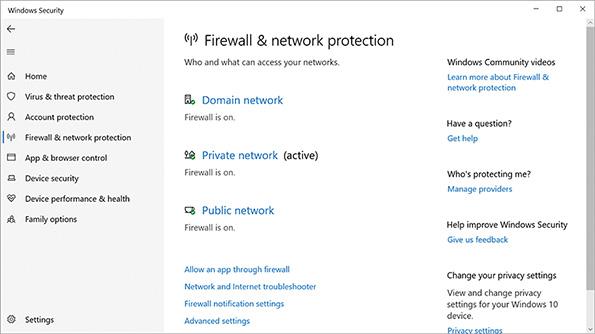
Follow these steps to find out how to configure Windows Defender Firewall on a Windows 10 computer:

1. 1

From the search box, search for **Firewall & network protection** and open it in the Settings app. Notice in [Figure 11-10](javascript://) that the firewall is turned on for all three listed network types.

**Figure 11-10**

Windows Defender Firewall shows the computer is currently connected to a private network



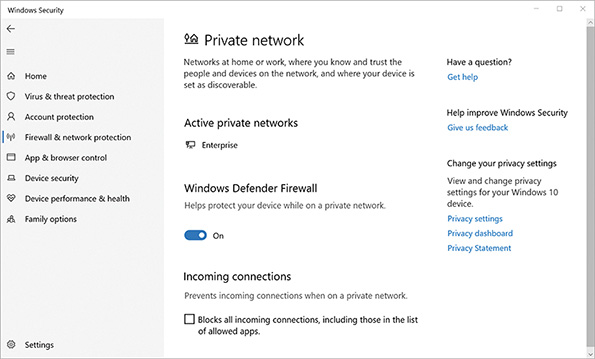
Enlarge Image

1. 2

To control firewall settings for each type of network location, click that network type. Use the slider to turn the firewall on or off for that network type (see [Figure 11-11](javascript://)).

**Figure 11-11**

Customize settings for a private or public network



Enlarge Image

1. 3

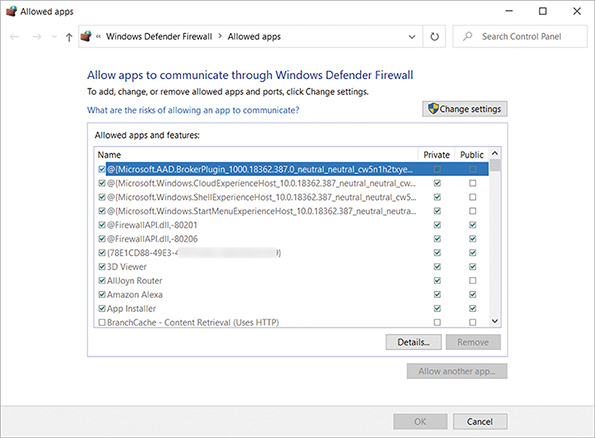
If you needed to allow no exceptions through the firewall on a network type, you would check **Blocks all incoming connections, including those in the list of allowed apps**. In the left pane, click **Firewall & network protection** to return to the firewall’s main page.

1. 4

To change the apps and programs allowed through the firewall, click **Allow an app through firewall**. The Allowed apps window appears (see [Figure 11-12](javascript://)). Click **Change settings**.

**Figure 11-12**

Allow apps to communicate through the firewall



Enlarge Image

1. 5

Scroll down to find the app you want to allow to initiate a connection from a remote computer to this computer and then, in the right side of the window, click the **Private** check box and/or the **Public** check box to indicate which type of network location the app is allowed to use. If you don’t see your app in the list, near the bottom of the window click **Allow another app…** to see more apps or to add your own. When you are finished making changes, click **OK** to return to the Firewall & network protection window.

1. 6

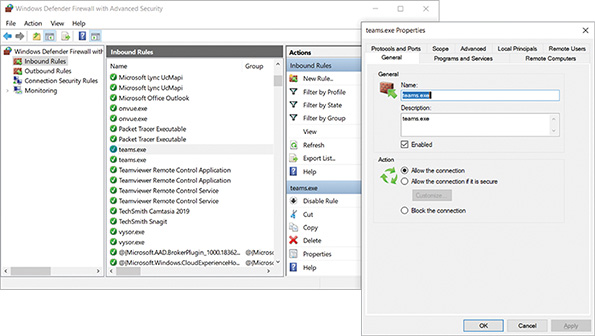
For even more control over firewall settings, in the Firewall & network protection window, click **Advanced settings** and click **Yes** on the UAC page. The Windows Defender Firewall with Advanced Security window opens.

1. 7

In the left pane, select **Inbound Rules** or **Outbound Rules**. A list of apps appears in the middle pane. Right-click an app and select **Properties** from the shortcut menu. The Properties dialog box gives full control of how exceptions get through the firewall, including which users, protocols, ports, and remote computers can use each opening (see [Figure 11-13](javascript://)).

**Figure 11-13**

Use advanced settings to control exactly how an app can get through Windows Defender Firewall



Enlarge Image

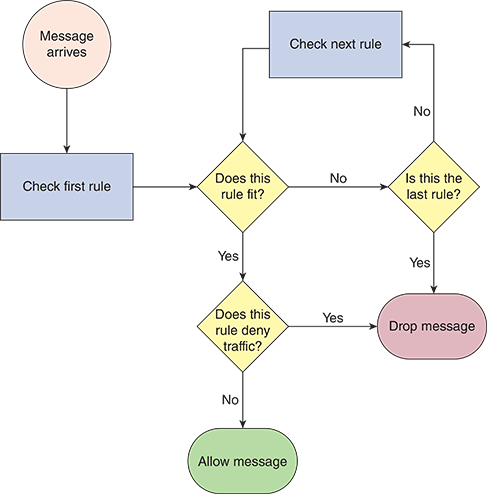
### Firewall Rules

Firewall rules function similarly to ACL rules that you read about earlier in this module. An ACL on a router can filter traffic according to its IP address or port number and can be used to sculpt traffic flows according to internal network needs (such as routing between VLANs). However, ACLs function on a single interface at a time, aren’t intended to manage a long list of rules, and can’t match rules to multiple streams of traffic within a single conversation (in other words, ACLs are stateless). Firewall rules provide more granular control of filters to secure traffic as it enters, exits, and traverses the network.

When a message crosses a firewall (physical or virtual), the firewall checks its rules to determine whether the message is allowed to pass. Earlier, you learned that routes are applied to a message according to which route in a route table most closely matches the message’s destination. With firewalls, however, rules are usually applied in order of priority. The highest-priority rule is checked first. If the message is allowed or denied according to that rule, no further rules are checked. A matching allow rule applies an explicit allow policy that lets the traffic through, while a matching deny rule applies an explicit deny policy that blocks the traffic. If the rule doesn’t apply to that message, the firewall moves to the next rule. This process continues until no rules remain. Like with ACLs, firewalls maintain an implicit deny policy for any messages that don’t match a specific rule. [Figure 11-14](javascript://) shows a diagram for how this process plays out.

**Figure 11-14**

Firewall rules are checked in order of priority to determine whether a message is allowed to pass



These rules check for information such as port, protocol, and IP address or CIDR range. If the information matches, the message is allowed or denied according to the matched rule. Firewall rules are configured differently for inbound and outbound traffic. In other words, there’s a different list of rules for inbound traffic, which is typically more restricted, than there is for outbound traffic.

However, firewalls can also alter their rules. For example, suppose the firewall is configured to deny incoming ICMP messages so Internet devices cannot ping your web server. At the same time, you’ve allowed outgoing ICMP messages so your server can ping other devices on the Internet. For your ping to work, however, ICMP responses must be able to enter your network through your firewall. A stateful firewall can acknowledge incoming ICMP responses that match your outgoing ICMP request, and the firewall will allow the return messages. In a Capstone Project at the end of this module, you’ll allow ICMP through a Windows firewall so you can ping a Windows VM from another VM.

### Troubleshooting Firewalls

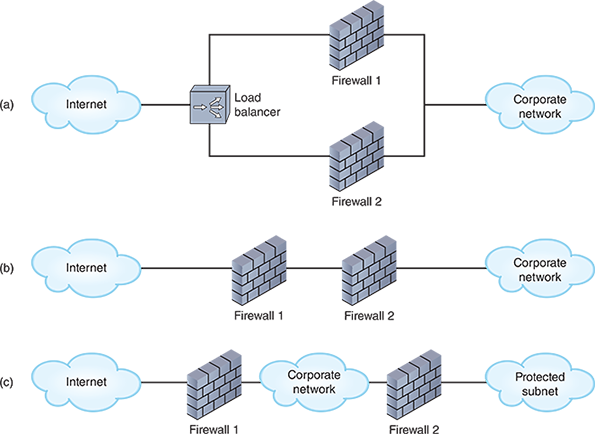
The most common cause of firewall problems is firewall misconfiguration. Blocked services, ports, or addresses on a firewall can cause a frustrating failure of service until you locate the specific firewall rule blocking your traffic. At the same time, a misconfigured firewall might allow risky traffic into your network that no one notices until it’s too late. Configuring an enterprise-level firewall can take weeks to achieve the best results. The configuration must not be so strict that it prevents authorized users from transmitting and receiving necessary data, yet not so lenient that you unnecessarily risk security breaches.

Further complicating the matter is that you might need to create exceptions to the rules. For example, suppose that your HR manager is working from a conference center in Salt Lake City while recruiting new employees, and they need to access the Denver server that stores payroll information. In this instance, the Denver network administrator might create an exception to allow transmissions from the HR manager’s workstation’s IP address to reach that server. In the networking industry, creating an exception to the filtering rules is called “punching a hole” in the firewall.

Less commonly, you might experience a full-scale firewall failure. While expensive, adding one or more redundant firewalls to your network can lessen the impact of this type of problem. You can help justify the expense of multiple firewalls by configuring load balancing and active-active failover so all firewalls contribute to increased performance, and any firewall can fail without losing service. In this arrangement, traffic is split between two (or more) firewalls, as shown in [Figure 11-15a](javascript://). Alternatively, redundant firewalls might be arranged in a serial fashion to provide enhanced protection for the internal network (see [Figure 11-15b](javascript://)) and more nuanced protection for different network segments (see [Figure 11-15c](javascript://)).

**Figure 11-15**

(a) Two load-balanced firewalls; (b) Enhanced security for the internal network; (c) Optimized security for a protected subnet



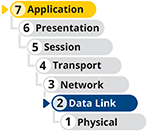
Enlarge Image

Go to pg.

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

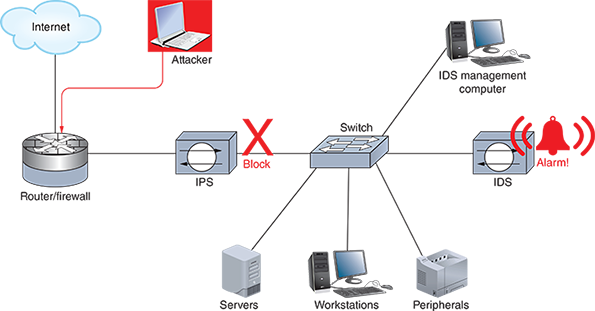
## 11-2cIDS (Intrusion Detection System)



An [**IDS (intrusion detection system)**](javascript://) is a stand-alone device, an application, or a built-in feature running on a workstation, server, switch, router, or firewall. It monitors network traffic, generating alerts about suspicious activity (see the right side of [Figure 11-16](javascript://)). Whereas a router’s ACL or a firewall acts like a bouncer at a private club who checks everyone’s ID and ensures that only club members enter through the door, an IDS is generally installed to provide security monitoring inside the network, similar to security personnel sitting in a private room monitoring closed-circuit cameras in the club and alerting other security personnel when they see suspicious activity.

**Figure 11-16**

An IDS detects traffic patterns, while an IPS can intercept traffic that might threaten a corporate network



Enlarge Image

An IDS uses two primary methods for detecting threats on the network:

* **Statistical anomaly detection**—Compares network traffic samples to a predetermined baseline to detect anomalies beyond certain parameters.
* **Signature-based detection**—Looks for identifiable patterns, or [**signatures**](javascript://), of code that are known to indicate specific vulnerabilities, exploits, or other undesirable traffic on the organization’s network (such as games). To maintain effectiveness, these signatures must be regularly updated in a process called **[signature management](javascript://)**. This also includes retiring irrelevant signatures and selecting the signatures most relevant to a specific network’s needs to most efficiently use memory and processing resources when scanning network traffic.

The most thorough security employs both IDS implementations listed as follows to detect a wider scope of threats and provide multiple levels of defense:

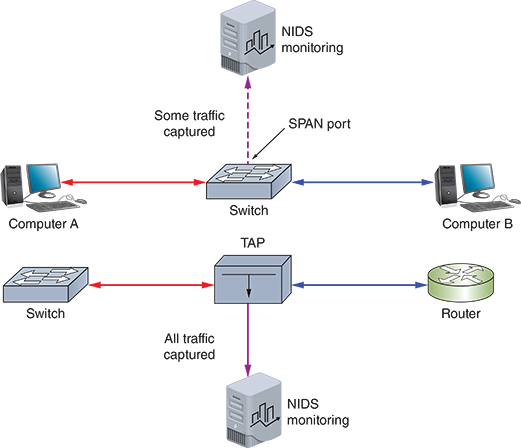
* An [**HIDS (host-based intrusion detection system)**](javascript://) runs on a single computer to detect attacks to that one host. For example, an HIDS might detect an attempt to exploit an insecure application running on a server or repeated attempts to log on to the server. An HIDS solution might also include [**FIM (file integrity monitoring)**](javascript://), which alerts the system of any changes made to files that shouldn’t change, such as operating system files. FIM works by generating a baseline checksum of the monitored files and then recalculating the checksum at regular intervals to determine if anything has changed.
* An [**NIDS (network-based intrusion detection system)**](javascript://) protects a network or portion of a network and is usually situated at the edge of the network or in a network’s protective perimeter, known as a screened subnet (formerly called a DMZ, or demilitarized zone). Here, it can detect many types of suspicious traffic patterns, such as those typical of DoS attacks.

An NIDS sits off to the side of network traffic and is sent duplicates of packets traversing the network. One technique that an NIDS might use to monitor traffic carried by a switch is port mirroring. In [**port mirroring**](javascript://), also called SPAN (switched port analyzer), one port on a switch is configured to send a copy of all the switch’s traffic to the device connected to that port. The device runs a monitoring program, which can now see much of the traffic the switch receives. This configuration is managed using the monitoring session command on Cisco switches and consists of identifying the source interface (the port to be monitored) and the destination interface (the port where copied messages are sent).

Similarly, an NIDS monitoring device might be connected to a [**TAP (test access point)**](javascript://), which can capture all traffic traversing a network connection, not just some of it (see [Figure 11-17](javascript://)). For example, the TAP might be inserted between a switch and a router. Depending on the volume of traffic, the TAP might feed this traffic to multiple monitoring devices.

**Figure 11-17**

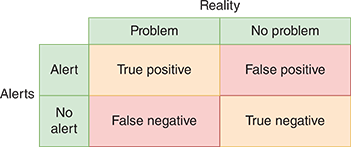
SPAN captures some traffic crossing a switch, but a TAP can capture all traffic between two devices



One drawback to using an IDS is the number of false positives it can generate. A false positive is an alert indicating a problem that isn’t actually a problem. To understand this term, it might help to compare the possible scenarios for each event or alert on a system. Consider the diagram in [Figure 11-18](javascript://). A true positive is an alert sent for an actual problem. A true negative is when no alert is sent and there is no problem. Either of these situations means the IDS is functioning well. However, if you get an alert when there’s no problem (a false positive) or you don’t get an alert when there is a problem (a false negative), this can result in extra work for no reason or, worse, missed problems.

**Figure 11-18**

A false positive occurs when there is an alert but no real problem



An example of a false positive would be when multiple logon attempts of a legitimate user who forgot their password are interpreted as a security threat. Suppose the IDS is configured to alert the network manager each time such an event occurs. The network manager might get overwhelmed by an endless stream of warnings and eventually ignore all the IDS’s messages. As you can see, it’s important to take the time to customize IDS software thoughtfully. Also, make sure to update it regularly and remember to reevaluate the rules of detection to ensure the software continues to guard against new threats.

Major vendors of networking hardware, such as Cisco, Juniper Networks, and Palo Alto, sell IDS-equipped devices. However, most IDS solutions these days are software-based and can be installed on a variety of network-connected machines. Examples of popular, open-source IDS software include Suricata ([suricata-ids.org](http://suricata-ids.org/" \t "_blank)) and the very popular Snort ([snort.org](http://snort.org/" \t "_blank)).

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-2dIPS (Intrusion Prevention System)

Although an IDS can only detect and log suspicious activity, an [**IPS (intrusion prevention system)**](javascript://) stands in-line between the attacker and the targeted network or host where it can prevent traffic from reaching that network or host (see the left side of [Figure 11-16](javascript://)). If an IDS is similar to security personnel using closed-circuit cameras to monitor a private club, an IPS would be similar to security personnel walking around in the club available to escort unruly patrons to the exit door. IPSs were originally designed as a more comprehensive traffic analysis and protection tool than firewalls. However, firewalls have evolved, and as a result, the differences between a firewall and an IPS have diminished.

Because an IPS stands in-line with network traffic, it can stop that traffic. For example, if an IPS detects a hacker’s attempt to flood the network with traffic, it can prevent that traffic from proceeding to the network. Thereafter, the IPS might quarantine that malicious user based on the sending device’s IP address. At the same time, the IPS continues to allow valid traffic to pass.

As with IDS, an [**NIPS (network-based intrusion prevention system)**](javascript://) can protect entire networks while an [**HIPS (host-based intrusion prevention system)**](javascript://) protects a specific host. Using NIPS and HIPS together increases the network’s security. For example, an HIPS running on a file server might accept a hacker’s attempt to log on if the hacker is posing as a legitimate client. With the proper NIPS, however, such a hacker would likely never get to the server. Like an IDS, an IPS requires careful configuration to avoid an abundance of false alarms.

Both an IDS and IPS can be placed inside a network or on the network perimeter. Notice in [Figure 11-19](javascript://) one NIPS is used to monitor and protect traffic in the DMZ. A second NIPS is positioned inside the private network on the perimeter of segment A to monitor and protect traffic on this one network segment. In the figure, you can see that HIPS software is also running on a server.

**Figure 11-19**

Placement of IPS devices and software on a network



Enlarge Image

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-2eCloud Security Technologies

The security devices you’ve read about so far can be virtualized and run in the cloud. However, cloud platforms often embed security technologies within their services to protect against security challenges unique to the cloud. In response to these changing needs, many technologies have emerged to protect cloud resources from external attacks. If you think about cloud resources in layers—platform, network, instances, applications, and data—you can see where defense-in-depth strategies can be applied at each of these layers.

While there are some interesting variations in how each security tool functions from platform to platform, most include the following:

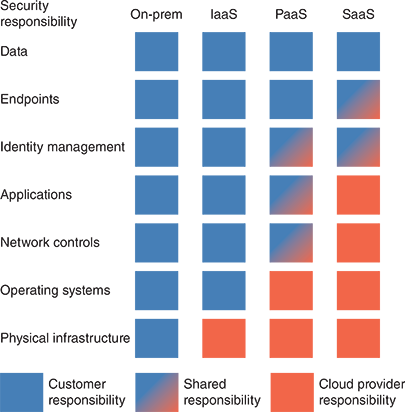
* **Granularity**—Some of these security appliances are designed specifically to hold a secure perimeter around individual resources (similar to a host-based firewall). For example, an AWS security group filters traffic into and out of a single EC2 instance or other resource. In contrast, an AWS NACL (network ACL, pronounced nackle) filters traffic to an entire VPC (virtual private cloud), similar to a network-based firewall.
* **Awareness**—Some cloud security technologies are stateless and don’t map incoming and outgoing traffic streams to each other, while other technologies are stateful and maintain an awareness of how traffic streams are related to each other. For example, Google’s virtual firewall rules are stateful, meaning that traffic allowed in one direction automatically allows traffic in the other direction for an active connection. In contrast, AWS’s NACLs are stateless.
* **Default configuration**—Each tool comes with a default configuration that either allows everything or denies everything so admins know what to expect when the tool is initially deployed and can plan for needed configuration changes. For example, you’ve seen how AWS security groups automatically allow SSH traffic over port 22 if you’re creating a Linux EC2 instance. Similarly, Azure’s NSG (Network Security Group) tool contains six default rules that cannot be changed or deleted but can be overridden by changing their priority ratings.

Cloud platforms include many other built-in security configurations that help prevent common security mistakes. For example, AWS’s S3 (Simple Storage Service) buckets are automatically configured to deny any sort of access from the Internet. In fact, you must perform several configuration steps in S3 to allow Internet access to an S3 bucket, with multiple confirmation steps along the way.

Cloud security works according to the [**shared responsibility model**](javascript://), meaning that the cloud provider is partially responsible for your cloud’s security and you’re responsible for the rest of it. [Figure 11-20](javascript://) shows how this shared responsibility breaks down according to the type of cloud deployment.

**Figure 11-20**

The customer’s responsibility for security shifts with different cloud deployment structures



**Remember This…**

* Compare and contrast network security devices, including proxy servers, firewalls, IDS, and IPS devices.
* Manage firewalls and firewall rules.
* Troubleshoot firewall settings.
* Compare SPAN and TAP.
* Summarize security implications of cloud computing.

**Self-Check**

1. Which device can be used to increase network performance by caching websites?

Answer

* 1. Firewall
  2. Proxy server
  3. IDS
  4. Security group

1. Which firewall type can protect a home network from adult content not suitable for the family’s children?

Answer

* 1. Packet-filtering firewall
  2. Host-based firewall
  3. Stateless firewall
  4. Application layer firewall

1. Which security device relies on a TAP or port mirroring?

Answer

* 1. NIDS
  2. HIPS
  3. FIM
  4. HIDS

**You’re Ready**

You’re now ready to complete [Project 11-3: Configure Cloud Security in AWS](javascript://), or you can wait until you’ve finished reading this module.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# 11-3Authentication, Authorization, and Accounting (AAA)

### Certification

* 4.1:

Explain common security concepts.

* 4.3:

Given a scenario, apply network hardening techniques.

* 4.4:

Compare and contrast remote access methods and security implications.

Average reading time: 24 minutes

As you’ve seen throughout this course, IT security permeates a network in many ways. Monitoring traffic entering, exiting, and traversing your network is handled by devices such as firewalls and IDS or IPS. You also need to manage which devices and users can access your network and how they prove their identity before being given access. This process is called [**access control**](javascript://). Physical access control, as you’ve already learned, might include locked doors, badges, or a locked rack. But you also need to implement information access control to limit the users and devices that can get to your data and other network resources.

Controlling access to a network and its resources consists of three major elements: authentication, authorization, and accounting. Together, this framework is abbreviated as [**AAA (authentication, authorization, and accounting)**](javascript://) and is pronounced triple-A. Occasionally you’ll see the acronym [**AAAA (authentication, authorization, accounting, and auditing)**](javascript://) to further emphasize monitoring and security standards involved in these processes; however, most IT security professionals wrap auditing into accounting and so use the AAA acronym. The components required to manage access control to a network and its resources are described next:

* **Authentication**—As you know, authentication (in this case, user authentication) is the process of verifying a user’s credentials (typically a username and password) to grant the user access to secured resources on a system or network. In other words, authentication asks the question, “Who are you?”
* [**Authorization**](javascript://)—Once a user has access to the network, the authorization process determines what the user can and cannot do with network resources. In other words, authorization asks the question, “What are you allowed to do?” Authorization restrictions affect layer 2 segmentation, layer 3 filtering, and layer 7 entitlements. For example, what VLAN are you assigned to? What servers or databases can you access? What commands can you run on a device?
* [**Accounting**](javascript://)—The accounting system logs users’ access and activities on the network. In other words, accounting asks, “What did you do?” The records that are kept in these logs are later audited, either internally or by an outside entity, to ensure compliance with existing organizational rules or external laws and requirements.
* **Auditing**—As you already know, a network security audit consists of a posture assessment that analyzes network configurations for vulnerabilities and might be performed in the context of legal compliance requirements, such as for HIPAA compliance. In the context of AAA, auditing refers to checking user configurations for problems, searching for signs of account misuse, and monitoring a network for compromised accounts. Normally, this is considered a part of network accounting functions and is not listed separately.

Let’s look at each of the elements of AAA in more detail, beginning with authentication.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-3aAuthentication

The identity of any user or device attempting to access a network, data, or device must be accurately verified before being given access to these resources. This is the role of authentication. Notice that a user can be authenticated to a single device and its local resources or to the network and its resources, which might be local in the on-prem data center or might be distributed across many data centers. When authenticating to a single device using that device’s own resources, the process is called local authentication. For example, a user can sign into Windows using a local user account. With network authentication, they can sign into the network using a network user account that is stored in an authentication database, such as Active Directory on a Windows domain. Consider the following explanations.

### Local Authentication

Local authentication processes are performed on the local device. Usernames and passwords are stored locally, which has both advantages and disadvantages, as follows:

* **Low security**—Most end user devices are less secure than network servers. A hacker can attempt a brute force attack or other workarounds to access a single device. If those same credentials are used on other devices, then all these devices are compromised. Also, local authentication does not allow for remotely locking down a user account.
* **Convenience varies**—For only a handful of devices, managing local accounts can be done a lot more easily than setting up a Windows domain, directory services, and all the supporting configurations. However, once you surpass about a dozen devices, the convenience of local authentication declines considerably.
* **Reliable backup access**—In the case of a network failure or server failure, the only workable option is local authentication. For this reason, networking devices and servers should be configured with a local privileged account that is only used when authentication services on the network are unavailable, and of course this account should have very secure credentials.

**Applying Concepts 11-3**

### Apply Local Security Policies

With local authentication, you can set security policies to require all local users to have passwords and to rename default user accounts. The [**Group Policy**](javascript://) (gpedit.msc) utility is a Windows console that controls what users can do and how the system can be used. Group Policy works by making entries in the Registry; applying scripts to Windows start-up, shutdown, and logon processes; and adjusting security settings. Policies can be applied to the computer or to the user.

Follow these steps to set a few important security policies on a Windows 10 Professional, Enterprise, or Education computer (Windows 10 Home does not include Group Policy):

1. 1

Sign into Windows using an administrator account. Press **Win**+**R** on your keyboard and then enter **gpedit.msc** to open the Local Group Policy Editor window.

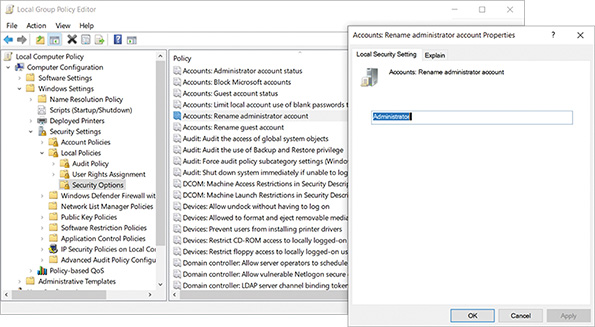
1. 2

To change a policy, first use the left pane to select the appropriate policy group and then use the right pane to view and edit a policy. Consider making the following important security policy adjustments:

* 1. **Change default usernames**—A hacker is less likely to hack into the built-in Administrator account or Guest account if you change the names of these default accounts. To change the name of the Administrator account, select the Security Options group in the left pane as follows: **Computer Configuration, Windows Settings, Security Settings, Local Policies, Security Options**. See the left side of [Figure 11-21](javascript://). In the right pane, double-click **Accounts: Rename administrator account**. In the Properties dialog box for this policy (see the right side of [Figure 11-21](javascript://)), change the name and click **OK**. To change the name of the Guest account, use the policy **Accounts: Rename guest account.**

**Figure 11-21**

Change default usernames in the Local Group Policy Editor

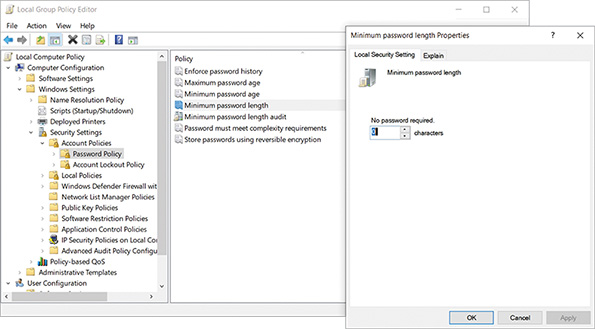


Enlarge Image

* 1. **Require user passwords**—The **password policy** is probably the most important policy used to secure a system. To require that all user accounts have passwords, select the Password Policy group in the left pane as follows: **Computer Configuration, Windows Settings, Security Settings, Account Policies, Password Policy**. See the left side of [Figure 11-22](javascript://). Use the **Minimum password length** policy and set the minimum length to 12 characters or more (see the right side of [Figure 11-22](javascript://)).

**Figure 11-22**

Require user passwords



Enlarge Image

* 1. **Allow only a single logon**—By default, Windows allows fast user switching, which means multiple users can log on to Windows at the same time. By disabling this feature, you require a user to save their work and log off the computer for another user to sign into it. This frees up computer resources and protects user data. To disable access to the fast user switching feature and allow only a single logon, select the Logon group in the left pane as follows: **Computer Configuration, Administrative Templates, System, Logon**. Double-click the **Hide entry points for Fast User Switching** policy. Enable this policy so that the Switch user option is dimmed and not available on the sign-in screen, the Start menu, and Task Manager.

1. 3

When you finish setting your local security policies, close the Local Group Policy Editor window. To implement your changes, reboot the system or enter the command **gpupdate.exe** in a PowerShell or Command Prompt window.

**Note 11-4**

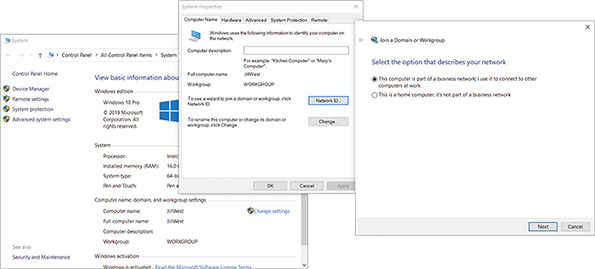
Sometimes policies overlap or conflict. To see the current policies for a particular computer or user, you can use the gpresult command in a PowerShell or Command Prompt window. To learn about the appropriate parameters for this command, search the [docs.microsoft.com](http://docs.microsoft.com/" \t "_blank) website.

With local authentication, every computer (workstation or server) on the network is responsible for securing its own resources. If several users need access to a file server, for example, each user must have a local user account on the file server. This local account and password must match the user account and password that the account holder used to sign into Windows at their workstation. As a network grows, keeping all these local accounts straight can become an administrative nightmare. The time will come when you will want to move on to a Windows domain.

In Windows, you can switch from local authentication to network authentication on the domain using the System Properties dialog box. To make the switch, in the System Properties dialog box (see [Figure 11-23](javascript://)), click **Network ID** and then select **This computer is part of a business network; I use it to connect to other computers at work**. Click **Next** and select **My company uses a network with a domain**. When you click **Next**, you are given the opportunity to enter your username, password, name of the Windows domain, and the name of your computer on the Windows domain. All this information is stored in Active Directory by Windows Server. When you complete the process, the next time you sign into Windows, you will use the network username to sign into the Windows domain. Active Directory then controls the access you have to resources on the network.

**Figure 11-23**

Switch from local authentication to authentication on a Windows domain



Enlarge Image

### Network Authentication and Logon Restrictions

Network authentication might be performed locally, such as when a student signs into a lab computer, or remotely, such as when a remote employee uses a VPN to access network resources. Regardless of the user’s physical location or network connection type, you can harden your network by requiring secure passwords and implementing other authentication restrictions. The following is a list of additional authentication restrictions that strengthen network security:

* **Time of day**—Some user accounts may be active only during specific hours, for example, between 8:00 a.m. and 5:00 p.m. Specifying valid hours for an account can increase security by preventing any account from being used by unauthorized personnel after hours.
* **Total time logged on**—Some user accounts may be restricted to a specific number of hours per day of logged-on time. Restricting total hours in this way can increase security of temporary user accounts. For example, suppose that your organization offers an Adobe Photoshop training class to a group of high school students one afternoon, and the Photoshop program and training files reside on your staff server. You might create accounts that could log on for only four hours on that day.
* **Source address**—You can specify that user accounts may log on only from certain workstations or certain areas of the network (that is, domains or segments). This restriction can prevent unauthorized use of accounts from workstations outside the network.
* **Unsuccessful logon attempts**—Hackers might repeatedly attempt to log on under a valid username for which they do not know the password. As the network administrator, you can set a limit on how many consecutive, unsuccessful logon attempts from a single user ID the server will accept before blocking that ID from even attempting to log on.
* **Geographic location**—Recall that geofencing determines a client’s geographic location to enforce a virtual security perimeter. In other words, the client must be located within a certain area to gain access to the network. With geofencing, GPS (global positioning system) or RFID (radio frequency identification) data can be sent to the authentication server to report the location of the device attempting to authenticate to the network.

**Note 11-5**

A special kind of DoS attack called an authentication attack floods a AAA server with authentication requests that must all be processed and responded to. This can force the server to shut down. By default, a floodguard feature might be configured on the AAA server to reclaim compromised resources. Floodguard settings can be changed with the floodguard command.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-3bAuthorization

Even the best authentication techniques—including encryption, computer room door locks, security policies, and password rules—make no difference if you give users access to resources they shouldn’t see. Once a user is given access to your network, that doesn’t mean they should have access to everything, such as sensitive data or critical network configurations. Controlling what a user can do once inside your network is the job of authorization. For example, you might have employees working from home who use a VPN to remote into your network. Can those employees print files to the printers? Can they access the customer database? Which applications can they run? What changes can they make to their own data or to data managed by other employees?

User access to network resources typically falls into one of these two categories:

1. the privilege or right to run, install, and uninstall software; and
2. permission to read, modify, create, or delete data files and folders.

The way user privileges and permissions are managed vary depending on who determines these privileges and permissions and how users are matched to privileges and permissions.

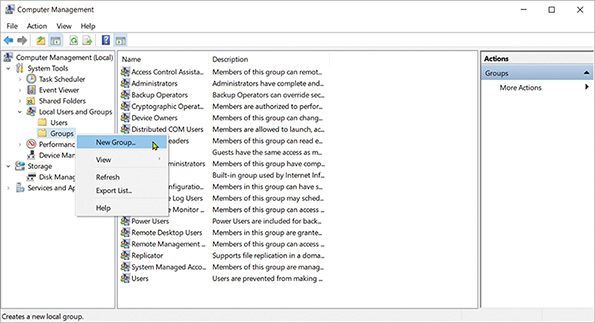
The most popular authorization method is [**RBAC (role-based access control)**](javascript://). With RBAC, a network administrator receives from a user’s supervisor a detailed description of the roles or jobs the user performs for the organization. The administrator is responsible for assigning the privileges and permissions necessary for the user to perform only these roles. In addition, all users might require access to certain public resources on the network, for example, a portion of the company website available to all employees. In most cases, these public rights are very limited.

With role-based access control, a network administrator creates user groups associated with these roles and assigns privileges and permissions to each user group. Each user is assigned to a user group that matches a requirement for their job. In most cases, a user can belong to more than one user group. In some situations, however, a checks and balances safety net is enforced by implementing role separation; this means each user can only be a member of a single group in order to perform any tasks at all. If a user is listed in more than one group, all privileges and permissions are locked down for that user.

For Windows, [Figure 11-24](javascript://) shows the Computer Management window where you can see several built-in user groups and the option to create your own, new groups. For example, the IT Department at a large university will most likely need more than one person who can create new user IDs and passwords for students and faculty. Naturally, the staff in charge of creating these credentials need the correct privileges to perform this task. You could assign the appropriate rights to each staff member individually, but a more efficient approach is to create an identity management group, and put all the IT personnel in that group. Later, when someone leaves the IT Department or joins the department, you can easily remove users from or add users to the group.

**Figure 11-24**

Windows allows you to create new groups and add users to these groups



Enlarge Image

Windows provides the option to create local groups on individual workstations. Active Directory gives additional options for creating domain local groups, which are centrally managed for the entire network.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-3cAccounting

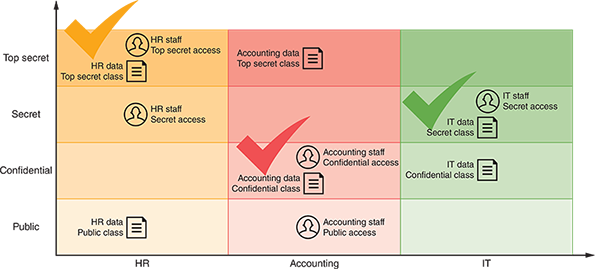
An important concept in accounting (whether that’s the field of accounting or the AAA component of accounting in IT security) is **[SoD (separation of duties)](javascript://)**, which refers to a division of labor that ensures no one person can singlehandedly compromise the security of data, finances, or other resources. To accomplish this goal, sensitive privileges and responsibilities are distributed to multiple people. A classic example in the field of accounting is having one person responsible for writing checks and a different person responsible for balancing the financial records. (While this scenario is not the origin of the phrase “checks and balances,” it is a good illustration of the concept.)

**Note 11-6**

Two other popular methods of access control in addition to RBAC are DAC and MAC. The least secure of these options is [**DAC (discretionary access control)**](javascript://). This is where users decide for themselves who has access to that user’s resources. The most restrictive is [**MAC (mandatory access control)**](javascript://). In this case, resources are organized into hierarchical classifications, such as “confidential” or “top secret.” This is a vertical organization. Resources are also grouped into categories, perhaps by department, which is a horizontal organization (see [Figure 11-25](javascript://)). Users, then, are also classified and categorized. If a user’s classification and category match those of a resource, then the user is given access.

**Figure 11-25**

A user’s classification and category must match a resource’s classification and category for the user to have access to that resource



Enlarge Image

Essentially, SoD ensures that no single person is given sufficient power in a system to commit fraud or otherwise deeply compromise the system’s integrity. In the context of AAA’s accounting and auditing components, SoD requires that no one is responsible for monitoring and reporting on themselves, which would create a conflict of interest for that person. Accounting and auditing activities should be sufficiently spread across multiple job roles to reduce the company’s vulnerability to fraud (intentional damage) or mistakes (unintentional damage).

As actions are performed on data and other resources, all this activity is logged for further analysis. This logging and the follow-up audit of the information is part of AAA’s accounting component. Throughout this course, you have been learning about the many logs that systems generate so that an administrator can troubleshoot and audit these systems. With a Linux NOS, most logs are generated as text files. These text files can get quite long, and a network administrator is responsible for making sure they don’t hog server storage space. In addition, you can install a log file viewer to make it easier to monitor log files for interesting or suspicious events.

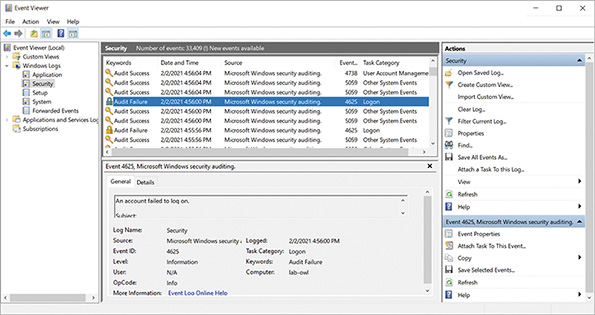
**Note 11-7**

Check out Linux commands tail, head, grep, sed, and awk, which are useful for searching very long log files.

In Windows, you can use Event Viewer to view Windows logs. For example, in [Figure 11-26](javascript://), you can see an Audit Failure event related to an account that failed to log on. As you can see in the figure, Audit events appear in the Windows Logs, Security group of Event Viewer. Also, before these logon events are logged, you must use Group Policy to turn on the feature.

**Figure 11-26**

Windows Event Viewer displays a security audit event



Enlarge Image

Data stored in logs must be monitored and analyzed to be of particular use in real time. [**SIEM (Security Information and Event Management)**](javascript://) systems can be configured to evaluate all this data, looking for significant events that require attention from the IT staff according to predefined rules. When one of these rules is triggered, an alert is generated and logged by the system. If programmed to do so, a notification is then sent to IT personnel via email, text, or some other method. The challenge is to find the right balance between sensitivity and workload. For example, a SIEM that isn’t sensitive enough will miss critical events that require response. However, a few hundred notifications per day will quickly overwhelm IT staff; they can’t possibly respond to so many alerts and will eventually start ignoring them.

A SIEM’s effectiveness is partly determined by how much storage space is allocated for the generated data and by the number of events it processes per second. As for data storage space, consider all the devices (including switches, routers, servers, and security systems) that will feed data to the SIEM, and allow for future growth of this traffic as well.

The network administrator can fine-tune a SIEM’s rules for the specific needs of a particular network by defining which events should trigger which responses. The SIEM system can also be configured to monitor particular indicators of anticipated problems or issues. These rules should be reevaluated periodically. Also, network technicians should review the raw data on a regular basis to ensure that no glaring indicators are being missed by existing rules. Examples of SIEM software include AlienVault OSSIM (Open Source SIEM), IBM Security QRadar, SolarWinds Security Event Manager, and Splunk Enterprise Security.

**Remember This…**

* Explain common network access control concepts, including defense in depth, separation of duties, role-based access, and local authentication.
* Describe the components of AAA: authentication, authorization, and accounting.

**Self-Check**

1. Which access control technique is responsible for detection of an intruder who succeeds in accessing a network?

Answer

* 1. Authentication
  2. Accounting
  3. Geofencing
  4. Separation of duties

1. Which authorization method will allow Nancy, a custodian, to access the company’s email application but not its accounting system?

Answer

* 1. RBAC
  2. Auditing
  3. DAC
  4. Local authentication

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# 11-4Authentication Technologies

### Certification

* 1.5

Explain common ports and protocols, their application, and encrypted alternatives.

* 4.1

Explain common security concepts.

Average reading time: 24 minutes

Of the three AAA processes, authentication tends to be the most complicated. Let’s look more closely at the building blocks that make authentication happen. Authentication protocols are the processes that require users, devices, and services to prove their identity before being allowed access to the network or other resources. Several types of authentication services and protocols exist, and some also incorporate authorization and auditing components. These technologies vary according to which encryption schemes they rely on and the steps they take to verify credentials.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

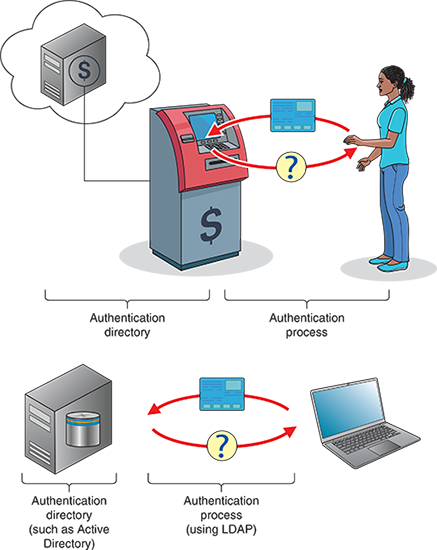
## 11-4aDirectory Services

For clients to authenticate to network resources (as opposed to individual devices), some sort of directory server on the network must maintain a database of account information, such as usernames, passwords, and any other authentication credentials. Often this is accomplished in AD (Active Directory) or something more Linux-focused like OpenLDAP ([openldap.org](http://openldap.org/" \t "_blank)) or 389 Directory Server ([directory.fedoraproject.org](http://directory.fedoraproject.org/" \t "_blank)).

All these options are built to be LDAP-compliant. LDAP was briefly defined when you studied network protocols in an earlier module. As you read then, LDAP (Lightweight Directory Access Protocol) is a standard protocol for accessing an existing directory. For example, consider the process you use to withdraw money from your bank account. Perhaps you go into the bank and present a cash withdrawal slip. You hand the form to the teller, and they ask for your ID. Alternatively, you might present your bank card at an ATM. The ATM responds by requesting your PIN, and then it asks you how much money you want to withdraw. Either way, you must complete a process that proves you are who you say you are and provides all the information requested by the bank. LDAP works much the same way. It defines the rules for how you communicate with an authentication directory so you can prove you are who you say you are (see [Figure 11-27](javascript://)). LDAP provides the rules and structure for the process of authentication, while the authentication directory (such as AD) provides the stored credentials and other relevant data.

**Figure 11-27**

The data and process of authentication



Enlarge Image

The mechanisms of LDAP dictate some basic requirements for any directory it accesses, and so there is a lot of commonality in how directory servers are configured regardless of the software used. LDAP can query the database, which means to draw information out of the database. It can also be used to add new information or edit existing data.

One weakness of LDAP is that it transmits user credentials in plaintext. Previously, when local networks defined the perimeter of a trusted environment, this wasn’t so bad. But these days, you can’t assume that your network is a safe place to transmit passwords and other sensitive data without encryption. Instead, today’s on-prem authentication processes should use the more secure LDAPS (LDAP over SSL/TLS), which communicates over port 636 instead of the insecure LDAP ports 389 or 2889.

By default, AD is configured to use the Kerberos protocol, which you’ll learn about shortly. However, AD can use LDAP instead or use both side by side. When supporting AD together, Kerberos provides authentication with the database, and then LDAP provides authorization by determining what the user can do while they’re on the network. Let’s take a closer look at Kerberos.

**Applying Concepts 11-4**

### Compare Windows Server AD to Azure AD

AD DS (Active Directory Domain Services) is the component of Active Directory that is responsible for storing user account information. AD has been around since Windows 2000 Server and took a significant leap forward with Windows Server 2008. However, AD (as it’s most commonly called) was not designed to integrate cloud services within its domain. Microsoft’s answer to this problem was to introduce Windows Azure Active Directory (AD). Azure AD is not intended as a replacement for AD, but rather as a supplement. Many of the concepts you’ve learned about in this course, such as cloud computing and security in network design, have built a solid foundation for you to now begin exploring Azure AD. Complete the following steps:

1. 1

Many employers, when interviewing technicians for job openings, will ask the job applicant what they know about Active Directory. Just like you need to be familiar with the user interface for Windows, Linux, and macOS, you also should know your way around AD. Spend some time researching online about how AD works and what it does. Watch some videos for basic functions, such as configuring users and user groups.

1. 2

Write a paragraph or two describing what you’ve learned. As practice for business-quality communications, carefully edit your writing. Use complete sentences, good grammar, and correct spelling.

**Note 11-8**

Some people have a hard time “hearing” their own writing. They use incomplete sentences or poorly constructed sentences without realizing it, and yet they don’t make these kinds of mistakes when speaking. If you struggle to write well, consider having someone else read your paragraph back to you out loud so you can hear your own mistakes. You might also copy and paste the text into Google Translate ([translate.google.com](http://translate.google.com/" \t "_blank)), which can read it back to you. Listen for statements that don’t make sense, that didn’t say what you intended them to say, or that could be interpreted in many different ways.

1. 3

Spend some time researching Azure AD and comparing it to Active Directory. What services does Azure AD offer that are the same as AD? What services does Azure AD offer that are different? Which protocols does Azure AD rely on? What AD limitations does Azure AD address? Also watch some videos showing how to use Azure AD.

1. 4

Write two paragraphs describing what you’ve learned. As practice for business-quality communications, carefully edit your writing. Use complete sentences, good grammar, and correct spelling. Consider drawing a diagram or two to illustrate the information you’re sharing.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-4bKerberos

Recall that Kerberos is the authentication protocol configured by default on Active Directory that works in partnership with LDAP. [**Kerberos**](javascript://) is a cross-platform authentication protocol that uses key encryption to verify the identity of clients and to securely exchange information after a client logs on to a system. It is an example of a private key encryption service and is considered especially secure. Let’s see how this works.

Kerberos does not automatically trust clients. Instead, it requires clients to prove their identities through a third party. This is similar to what happens when you apply for a passport. The government does not simply believe that you are, for example, “Peter Parker,” but instead requires you to present proof, such as your birth certificate. In addition to checking the validity of a client, Kerberos communications are encrypted and unlikely to be deciphered by any device on the network other than the intended client.

To understand specifically how Kerberos authenticates a client, you need to understand some of the terms used when discussing this protocol:

* [**Principal**](javascript://)—A Kerberos client or user
* [**KDC (Key Distribution Center)**](javascript://)—The server that issues keys to clients during initial authentication
* [**Ticket**](javascript://)—A temporary set of credentials a client presents to network servers to prove its identity has been validated

**Note 11-9**

A Kerberos ticket is not the same as a Kerberos key. Using a key is similar to using your credit card to pay for entrance into a carnival or county fair. Your entrance fee includes a time-limited wristband that you can use to obtain a separate ticket for each game, ride, or beverage that you consume during the event. As long as you’re at the event for that evening, you can get more tickets by showing your wristband, and each ticket is exchanged for another game, ride, or beverage. However, when you come back the next night, you must start over by using your credit card to purchase a new wristband. Keys, like credit cards, belong to the user or server and initially validate the user’s and server’s identity to each other during the authentication process to create a session. A ticket, like a carnival ticket, is used to gain access to another network service, such as email, an internal payroll site, a printer, or a file server.

A Kerberos server runs two services:

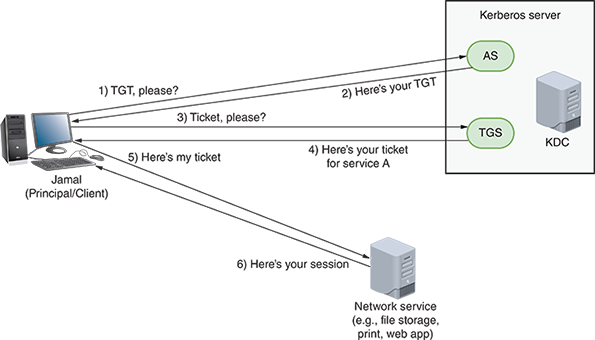
* **AS (authentication service)**—Initially validates a client. In the carnival analogy, this would be the box office at the entrance gate.
* **TGS (ticket-granting service)**—Issues tickets to an authenticated client for access to services on the network. This would be the ticket booth inside the fairgrounds, where you show your wristband to get more tickets.

Now that you have learned the basic terms used by Kerberos, you can follow the process it requires for client-server communication. Bear in mind that the purpose of Kerberos is to connect a valid user with a network service the user wants to access, such as email, printing, file storage, databases, or web applications. To accomplish this, both the user and the service must register their own keys with the AS ahead of time.

[Figure 11-28](javascript://) shows how TGS works. Suppose the principal (the client) is Jamal. The following steps describe the Kerberos authentication process:

**Figure 11-28**

The Ticket-Granting Service offers a client a ticket for each network service it needs to access



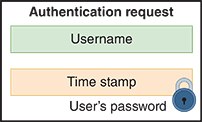
Enlarge Image

1. Step 1

When Jamal first logs on to the network, his computer sends an authentication request to the AS (see [Figure 11-29](javascript://)). This request contains Jamal’s username, but not his password. However, the time stamp on the request is encrypted with Jamal’s password.

**Figure 11-29**

Step 1: Authentication request

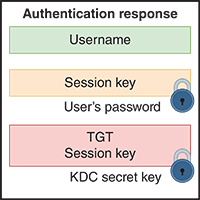


1. Step 2

The AS on the KDC first confirms that Jamal is listed in its database and uses his password (retrieved from its database) to decrypt the timestamp. If all goes well, the AS generates a session key, which is used for encryption and decryption for future communication. It encrypts this key with the user’s password (see [Figure 11-30](javascript://)). The AS also generates a TGT (Ticket-Granting Ticket), which will expire within a specified amount of time (by default, this limit is 10 hours). The TGT is like the wristband in the carnival analogy. To prevent counterfeiting, the TGT is encrypted with a secret KDC key so that only the KDC can read it and confirm its legitimacy.

**Figure 11-30**

Step 2: Authentication response

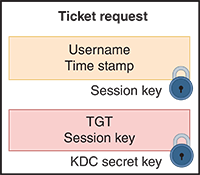


1. Step 3

After receiving the TGT, the principal decrypts the session key using the user’s password. If the correct password is used and decryption is successful, the principal can then submit a ticket request to the TGS for access to a network service (see [Figure 11-31](javascript://)). The request includes the user’s name and a time stamp that are both encrypted using the session key. It also includes the fully encrypted TGT, which the principal never decrypted.

**Figure 11-31**

Step 3: Ticket request

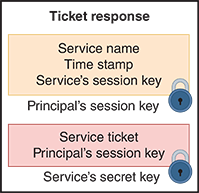


1. Step 4

The TGS validates the TGT and the rest of the request message’s contents and then creates a ticket that allows Jamal to use the network service. This ticket (see [Figure 11-32](javascript://)) includes the service’s name, a time stamp, and the service’s session key, all encrypted using the session key issued to the principal earlier. It also includes information the service will need to confirm the request is valid, including the principal’s session key. This part is encrypted using a secret key that the service knows, but the principal does not.

**Figure 11-32**

Step 4: Ticket response

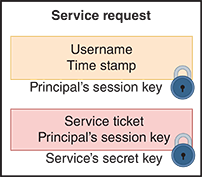


1. Step 5

Jamal’s computer decrypts the information it needs using the session key. It then creates a service request (see [Figure 11-33](javascript://)) that contains the encrypted information from the TGS, plus a time stamp encrypted with the session key.

**Figure 11-33**

Step 5: Service request



1. Step 6

The service decrypts the ticket using its own secret key, finds the principal’s session key included in the ticket, and then decrypts the remainder of the message to confirm its validity. Finally, the service verifies that the principal requesting its use is truly Jamal as the KDC indicated and then allows access.

**Note 11-10**

Kerberos, which was designed at MIT (Massachusetts Institute of Technology), is named after the three-headed dog in Greek mythology who guarded the gates of Hades. The three heads of the Kerberos authentication protocol are the principal, the network server providing a service, and the KDC. MIT still provides free copies of the Kerberos code. In addition, many software vendors have developed their own versions of Kerberos.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-4cSSO (Single Sign-On)

Kerberos is an example of [**SSO (single sign-on)**](javascript://), a form of authentication in which a client signs on one time to access multiple systems or resources. The primary advantage of single sign-on is convenience. Users don’t have to remember several passwords, and network administrators can limit the time they devote to password management. The biggest disadvantage to single sign-on authentication is that once the obstacle of authentication is cleared, the user has access to numerous resources. A hacker needs fewer credentials to gain access to potentially many files or connections.

For greater security, some systems—especially those using SSO—require clients to supply two or more pieces of information to verify their identity. In a [**2FA (two-factor authentication)**](javascript://) scenario, a user must provide something and know something. For example, they might have to provide a fingerprint scan as well as know and enter a password.

An authentication process that requires two or more pieces of information is known as [**MFA (multifactor authentication)**](javascript://). The following list gives the five most common categories of authentication factors, along with some examples of each:

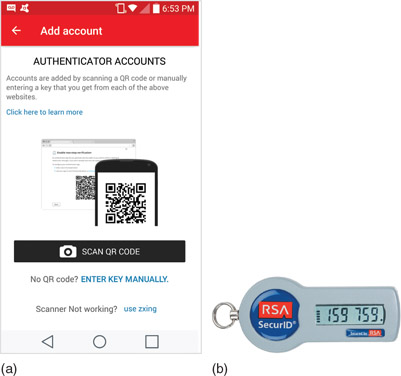
* **Something you know**—A password, PIN, or biographical data
* **Something you have**—An ATM card, smart card, or key
* **Something you are**—Your fingerprint, facial pattern, or iris pattern
* **Somewhere you are**—Your location in a specific building or secured closet
* **Something you do**—The specific way you type, speak, or walk

Multifactor authentication requires at least one authentication method from at least two different categories. For example, logging into a network might require a password, a fingerprint scan, plus a piece of information generated from a security token. A [**security token**](javascript://) is a device or application that stores or generates information (such as a series of numbers or letters) known only to its authorized user.

On the left side of [Figure 11-34](javascript://), a smartphone app requests a website-generated QR code to set up a user’s account, such as Facebook, for 2FA. Once established, a random code is generated every 30 seconds that must be entered in addition to the user’s password to access the account. An example of a hardware-based token is the popular RSA SecurID keychain fob from RSA Security, as shown on the right side of [Figure 11-34](javascript://). The RSA SecurID device generates a password that changes every 60 seconds. When logging on, a user provides the number that currently appears on the fob. Before the user is allowed access to secured resources, the network checks with RSA Security’s service to verify the number is correct. Similarly, Google Authenticator, Google’s number generator service, provides free, software-based security tokens.

**Figure 11-34**

(a) A smartphone 2FA app; (b) A SecurID fob



Source for (a): Twilio, Inc. Source for (b): Courtesy of RSA Security LLC

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-4dRADIUS (Remote Authentication Dial-In User Service)

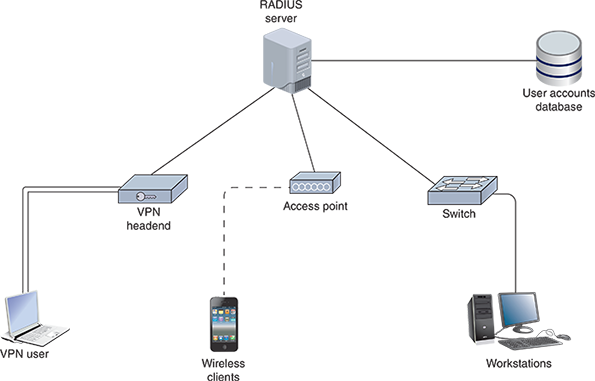
While Active Directory offers powerful authentication and other services, it’s limited to working with Windows or Linux clients, and it can’t directly support authentication through wireless access points. An alternative to Active Directory is the cross-platform RADIUS (Remote Authentication Dial-In User Service), which is an open-source standard developed by Livingston Enterprises in 1991 and later standardized by the IETF. It runs in the application layer; and in the transport layer, it can use either UDP or, as of 2012, TCP. RADIUS treats authentication and authorization as a single process, meaning that the same type of packet is used for both functions, while accounting is a separate process. Due to its origins as a remote authentication service, RADIUS specializes in supporting clients not directly wired into the network, such as clients of wireless access points or VPN-based clients.

RADIUS can operate as a software application on a remote access server or on a computer dedicated to this type of authentication, called a RADIUS server. Because RADIUS servers are highly scalable, many ISPs use RADIUS as a central authentication point for wireless, mobile, and remote users. RADIUS services are often combined with other network services on a single machine. For example, an organization might combine a DHCP server with a RADIUS server to manage allocation of addresses and privileges assigned to each address on the network.

[Figure 11-35](javascript://) illustrates a RADIUS server managing network access for local and remote users. RADIUS can run on almost all modern OSs. In fact, there are some scenarios where you might want to implement RADIUS alongside Active Directory, as illustrated in [Figure 11-36](javascript://). While RADIUS includes some very sophisticated accounting features, it also only encrypts the password in transmission, and so is not as secure as TACACS+, discussed next.

**Figure 11-35**

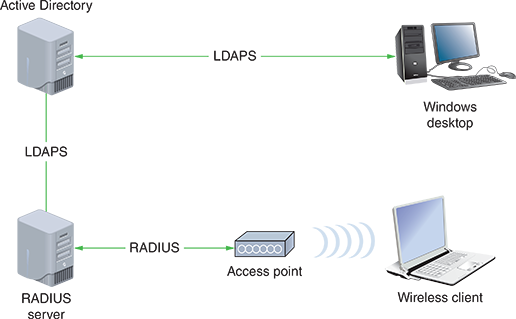
A RADIUS server on a network servicing various types of users



Enlarge Image

**Figure 11-36**

RADIUS and Active Directory can work together to support different types of clients



**Note 11-11**

A newer protocol called Diameter was developed in 1998 to replace RADIUS. In geometry, the diameter of a circle is twice its radius; the implication is that the Diameter protocol is twice as good as the RADIUS protocol. While many vendors offer Diameter options on their systems, for the most part, RADIUS is still more widely used.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 11-4eTACACS+ (Terminal Access Controller Access Control System Plus)

Another AAA protocol, [**TACACS+ (Terminal Access Controller Access Control System Plus)**](javascript://), offers network administrators the option of separating the authentication, authorization, and auditing capabilities. For instance, TACACS+ might provide access and accounting functions, but use another technique, such as Kerberos (discussed earlier in this module), to authenticate users. TACACS+ differs from RADIUS in that it does the following:

* Relies on TCP, not UDP, at the transport layer
* Was developed by Cisco Systems, Inc., for proprietary use
* Is typically used to authenticate to network devices such as routers and switches rather than workstations or servers
* Is most often used for device administration access control for technicians, although it can be used for network resource access control for users
* Encrypts all information transmitted for AAA (RADIUS only encrypts the password)

**Applying Concepts 11-5**

### Protocol Synopsis

Each of the protocols covered in this and previous modules plays an important role in securing transmissions between devices and locations. It’s important to have the big picture in mind regarding how these protocols interact with each other and the roles they play in various parts of the system when troubleshooting connectivity and security issues.

In this activity, you synthesize the major characteristics of each protocol into a single reference table. You can create the following [Table 11-1](javascript://) in a word-processing program or a spreadsheet program. Then refer to earlier modules to fill in the missing pieces. Protocol types include encryption, connection, authentication, tunneling, trunking, hashing, and AAA. Some of the listed protocols are included in more than one of these categories.

**Table 11-1**

### Notable Encryption and Authentication Methods

| **Security method** | **Type** | **Primary use(s)** | **Notes** |
| --- | --- | --- | --- |
| IPsec | Encryption | TCP/IP transmissions |  |
| SSL |  | TCP/IP transmissions |  |
| TLS |  |  | Secure transmission of HTTP sessions |
| SSH | Connection, authentication, encryption |  |  |
| RDP |  | Remote access |  |
| VNC | Connection |  |  |
| L2TP | Tunneling | VPN |  |
| GRE |  | VPN |  |
| OpenVPN |  | VPN |  |
| IKEv2 |  | VPN |  |
| VTP | Trunking |  |  |
| SHA |  | Data integrity |  |
| LDAP | Authentication | Directory access |  |
| Kerberos |  | Client validation | Verify the identity of clients and securely exchange information after a client logs on to a system |
| RADIUS |  |  | Central authentication point for network users, including wireless, mobile, and remote users |
| TACACS+ | AAA (authentication, authorization, and accounting) | Client validation and monitoring |  |
| EAP |  | Client verification |  |
| 802.1X | Authentication |  |  |
| AES |  | Wi-Fi and other uses |  |

Enlarge Table

**Remember This…**

* Explain common authentication protocols, including LDAP, LDAPS, RADIUS, TACACS+, and Kerberos.
* Compare authentication methods, including multifactor authentication and SSO.

**Self-Check**

1. Which authentication protocol is optimized for wireless clients?

Answer

* 1. RADIUS
  2. Active Directory
  3. TACACS+
  4. Kerberos

1. What does a client present to a network server to access a resource on that server?

Answer

* 1. Key
  2. Principal
  3. Ticket
  4. Ticket-Granting Ticket

**You’re Ready**

You’re now ready to complete [Project 11-4: Configure RADIUS in Packet Tracer](javascript://), or you can wait until you’ve finished the Review Questions for this module.

**You’re Ready**

After you finish the Hands-On Projects, you’re ready to complete the [Module 11 Capstone Projects](javascript://).

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# Module Review

## 11-5a**Module Summary**

### Network Hardening by Design

* A zero trust security model considers everything untrustworthy until proven otherwise. Security must be implemented in many, seemingly redundant layers that permeate the network and protect resources from every angle, which is a strategy called defense in depth.
* A router’s main function is to examine packets and determine where to direct them based on their network layer addressing information. Thanks to a router’s ACL (access control list), or access list, routers can also decline to forward certain packets depending on their content.
* While ACLs filter traffic into and out of router interfaces on the data plane, the control plane needs a separate layer of protection. In this case, an adaptation of QoS (quality of service) filters can be used to rate-limit traffic on the control plane and management plane of routers and switches using a feature called CoPP (control plane policing).
* IPv6 clients have no way of authenticating RA messages to know which of these messages come from legitimate sources and which might come from a network attacker. The solution to this problem is to configure RA guards on network switches. The RA guard feature filters RA messages so these messages can only come from specific interfaces on the switch.
* A rogue DHCP server running on a client device could be used to implement an on-path attack by configuring the attacker’s IP address as the victim computers’ default gateway. Alternatively, the attacker could give their IP address as the DNS server and then spoof websites. DHCP messages should be monitored by enabling DHCP snooping on a switch.
* ARP performs no authentication, and so it is highly vulnerable to attack. As a result, attackers can use ARP spoofing, also called ARP poisoning or ARP cache poisoning, to send faked ARP replies that alter ARP tables in the network. DAI (dynamic ARP inspection) can be configured on a switch to protect against ARP spoofing attacks. When DAI is enabled, the switch compares incoming messages with its DHCP snooping binding table to determine whether the message’s source IP address is appropriately matched with its source MAC address according to DHCP assignments on the network.

### Network Security Technologies

* A proxy server, or proxy, acts as an intermediary between the external and internal networks, screening all incoming and outgoing traffic. Although proxy servers only provide low-grade security relative to other security devices, they can help prevent an attack on internal network resources such as web servers and web clients.
* A firewall is a specialized device or software that selectively filters or blocks traffic between networks. A firewall might be placed internally, residing between two interconnected private networks. More commonly, the firewall is placed on the edge of the private network, monitoring the connection between a private network and a public network, such as the Internet.
* An IDS (intrusion detection system) is a stand-alone device, an application, or a built-in feature running on a workstation, server, switch, router, or firewall. It monitors network traffic, generating alerts about suspicious activity.
* Although an IDS can only detect and log suspicious activity, an IPS (intrusion prevention system) stands in-line between the attacker and the targeted network or host where it can prevent traffic from reaching that network or host.
* Cloud security works according to the shared responsibility model, meaning that the cloud provider is partially responsible for your cloud’s security and you’re responsible for the rest of it.

### Authentication, Authorization, and Accounting (AAA)

* Controlling access to a network and its resources consists of three major elements: authentication, authorization, and accounting. Together, this framework is abbreviated as AAA (authentication, authorization, and accounting) and is pronounced triple-A. Occasionally you’ll see the acronym AAAA (authentication, authorization, accounting, and auditing) to further emphasize monitoring and security standards involved in these processes; however, most IT security professionals wrap auditing into accounting and so use the AAA acronym.
* The identity of any user or device attempting to access a network, data, or device must be accurately verified before being given access to these resources. This is the role of authentication.
* Even the best authentication techniques—including encryption, computer room door locks, security policies, and password rules—make no difference if you give users access to resources they shouldn’t see. Once a user is given access to your network, that doesn’t mean they should have access to everything, such as sensitive data or critical network configurations. Controlling what a user can do once inside your network is the job of authorization.
* An important concept in accounting (whether that’s the field of accounting or the AAA component of accounting in IT security) is SoD (separation of duties), which refers to a division of labor that ensures no one person can singlehandedly compromise the security of data, finances, or other resources. In the context of AAA’s accounting and auditing components, SoD requires that no one is responsible for monitoring and reporting on themselves, which would create a conflict of interest for that person. Accounting and auditing activities should be sufficiently spread across multiple job roles to reduce the company’s vulnerability to fraud (intentional damage) or mistakes (unintentional damage).

### Authentication Technologies

* For clients to authenticate to network resources (as opposed to individual devices), some sort of directory server on the network must maintain a database of account information, such as usernames, passwords, and any other authentication credentials. Often this is accomplished in AD (Active Directory) or something more Linux-focused like OpenLDAP or 389 Directory Server.
* The mechanisms of LDAP dictate some basic requirements for any directory it accesses, and so, there is a lot of commonality in how directory servers are configured regardless of the software used. LDAP can query the database, which means to draw information out of the database. One weakness of LDAP is that it transmits user credentials in plaintext. Today’s on-prem authentication processes should use the more secure LDAPS (LDAP over SSL/TLS), which communicates over port 636 instead of the insecure LDAP ports 389 or 2889.
* Recall that Kerberos is the authentication protocol configured by default on Active Directory, and it works in partnership with LDAP. Kerberos is a cross-platform authentication protocol that uses key encryption to verify the identity of clients and to securely exchange information after a client logs on to a system.
* Kerberos is an example of SSO (single sign-on), a form of authentication in which a client signs on one time to access multiple systems or resources. The primary advantage of single sign-on is convenience.
* An alternative to Active Directory is the cross-platform RADIUS (Remote Authentication Dial-In User Service), which treats authentication and authorization as a single process, meaning that the same type of packet is used for both functions, while accounting is a separate process. Due to its origins as a remote authentication service, RADIUS specializes in supporting clients not directly wired into the network, such as clients of wireless access points or VPN-based clients.
* Another AAA protocol, TACACS+ (Terminal Access Controller Access Control System Plus), offers network administrators the option of separating the authentication, authorization, and auditing capabilities. It’s most often used for device administration access control for technicians, although it can be used for network resource access control for users.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# Module Review

## 11-5b**Key Terms**

* [**2FA (two-factor authentication)**](javascript://)
* [**AAA (authentication, authorization, and accounting)**](javascript://)
* [**AAAA (authentication, authorization, accounting, and auditing)**](javascript://)
* [**access control**](javascript://)
* [**accounting**](javascript://)
* [**ACL (access control list)**](javascript://)
* [**application layer firewall**](javascript://)
* [**ARP spoofing**](javascript://)
* [**authorization**](javascript://)
* [**CoPP (control plane policing)**](javascript://)
* [**DAC (discretionary access control)**](javascript://)
* [**DAI (dynamic ARP inspection)**](javascript://)
* [**DHCP snooping**](javascript://)
* [**FIM (file integrity monitoring)**](javascript://)
* [**Group Policy**](javascript://)
* [**HIDS (host-based intrusion detection system)**](javascript://)
* [**HIPS (host-based intrusion prevention system)**](javascript://)
* [**host-based firewall**](javascript://)
* [**IDS (intrusion detection system)**](javascript://)
* [**implicit deny**](javascript://)
* [**IPS (intrusion prevention system)**](javascript://)
* [**KDC (Key Distribution Center)**](javascript://)
* [**Kerberos**](javascript://)
* [**MAC (mandatory access control)**](javascript://)
* [**MFA (multifactor authentication)**](javascript://)
* [**network-based firewall**](javascript://)
* [**NIDS (network-based intrusion detection system)**](javascript://)
* [**NIPS (network-based intrusion prevention system)**](javascript://)
* [**packet-filtering firewall**](javascript://)
* [**port mirroring**](javascript://)
* [**principal**](javascript://)
* [**proxy server**](javascript://)
* [**RA guard**](javascript://)
* [**RBAC (role-based access control)**](javascript://)
* [**rogue DHCP server**](javascript://)
* [**security token**](javascript://)
* [**shared responsibility model**](javascript://)
* [**SIEM (Security Information and Event Management)**](javascript://)
* [**signature**](javascript://)
* [**signature management**](javascript://)
* [**SoD (separation of duties)**](javascript://)
* [**SSO (single sign-on)**](javascript://)
* [**stateful firewall**](javascript://)
* [**stateless firewall**](javascript://)
* [**TACACS+ (Terminal Access Controller Access Control System Plus)**](javascript://)
* [**TAP (test access point)**](javascript://)
* [**ticket**](javascript://)
* [**zero trust**](javascript://)

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# Module Review

## 11-5c**Review Questions**

1. At what layer of the OSI model do proxy servers operate?
   1. Layer 3
   2. Layer 2
   3. Layer 7
   4. Layer 4
2. Which of the following ACL commands would permit web-browsing traffic from any IP address to any IP address?
   1. access-list acl\_2 deny tcp any any
   2. access-list acl\_2 permit https any any
   3. access-list acl\_2 deny tcp host 2.2.2.2 host 3.3.3.3 eq www
   4. access-list acl\_2 permit icmp any any
3. Which of the following criteria can a packet-filtering firewall not use to determine whether to accept or deny traffic?
   1. Destination IP address
   2. SYN flags
   3. Application data
   4. ICMP message
4. What information in a transmitted message might an IDS use to identify network threats?
   1. Signature
   2. FIM
   3. Port mirroring
   4. ACL
5. Which principle ensures auditing processes are managed by someone other than the employees whose activities are being audited?
   1. Separation of duties
   2. Principle of least privilege
   3. Shared responsibility model
   4. Defense in depth
6. Who is responsible for the security of hardware on which a public cloud runs?
   1. The cloud customer
   2. It depends
   3. Both the cloud customer and the cloud provider
   4. The cloud provider
7. Which of the following is not one of the AAA services provided by RADIUS and TACACS+?
   1. Authentication
   2. Authorization
   3. Administration
   4. Accounting
8. Which device would allow an attacker to make network clients use an illegitimate default gateway?
   1. RA guard
   2. DHCP server
   3. Proxy server
   4. Network-based firewall
9. Which policy ensures messages are discarded when they don’t match a specific firewall rule?
   1. Implicit allow
   2. Explicit deny
   3. Explicit allow
   4. Implicit deny
10. Active Directory and 389 Directory Server are both compatible with which directory access protocol?
    1. LDAP
    2. RADIUS
    3. Kerberos
    4. AD DS
11. What are the two primary features that give proxy servers an advantage over NAT?
12. What kinds of issues might indicate a misconfigured ACL?
13. Any traffic that is not explicitly permitted in the ACL is , which is called the .
14. What kind of ticket is held by Kerberos’s TGS?
15. What’s the essential difference between an IPS and an IDS?
16. What causes most firewall failures?
17. What is the purpose of an ACL when configuring CoPP?
18. Why do network administrators create domain groups to manage user security privileges?
19. What characteristic of ARP makes it particularly vulnerable to being used in a DoS attack?
20. Why would you need separate RA guard policies for network hosts and routers attached to a switch?

Go to pg.

[**help**](javascript://)