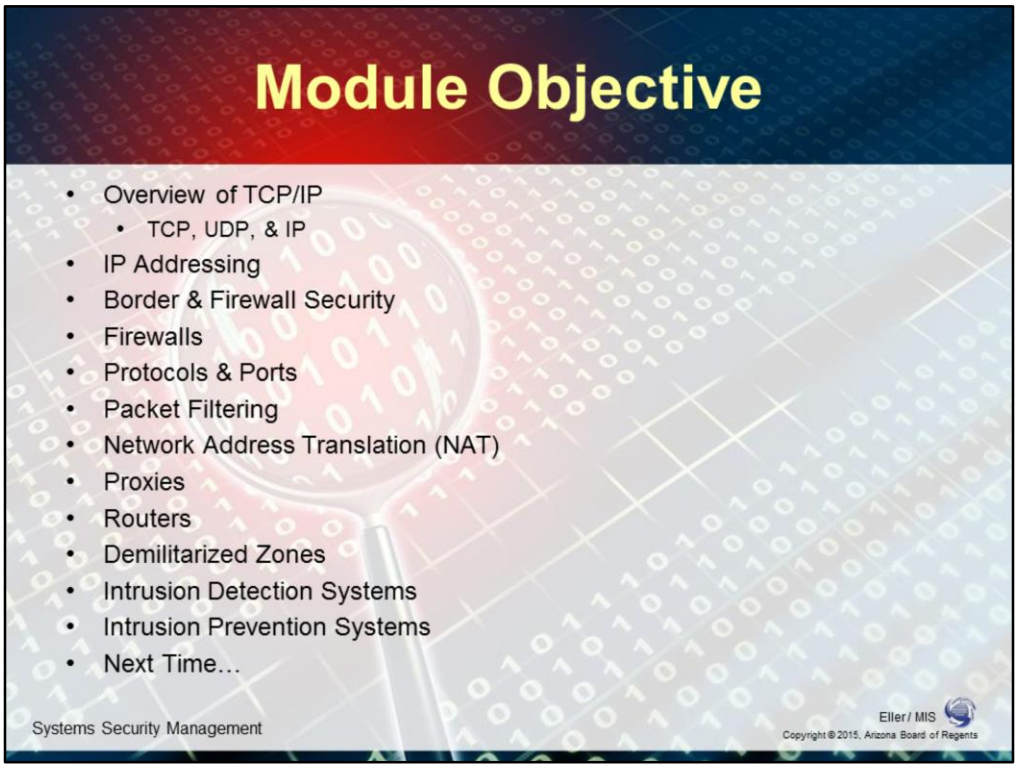
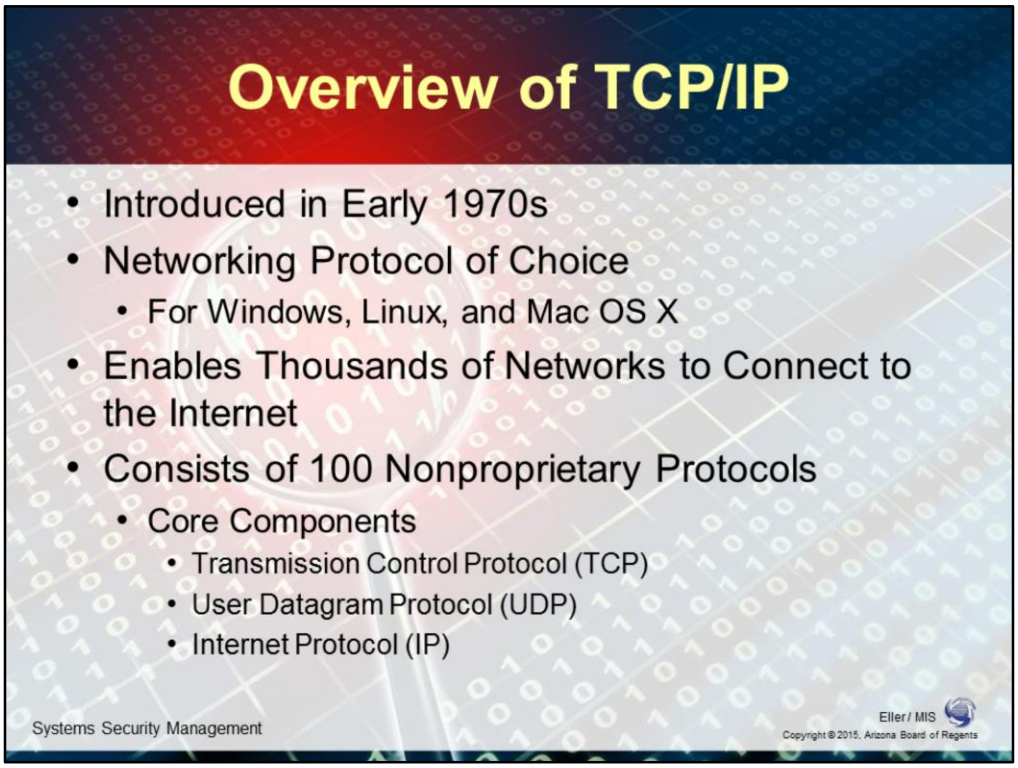

Firewalls & Border Security
Module 13
Systems Security Management
Eller / MIS
Copyright © 2015, Arizona Board of Regents



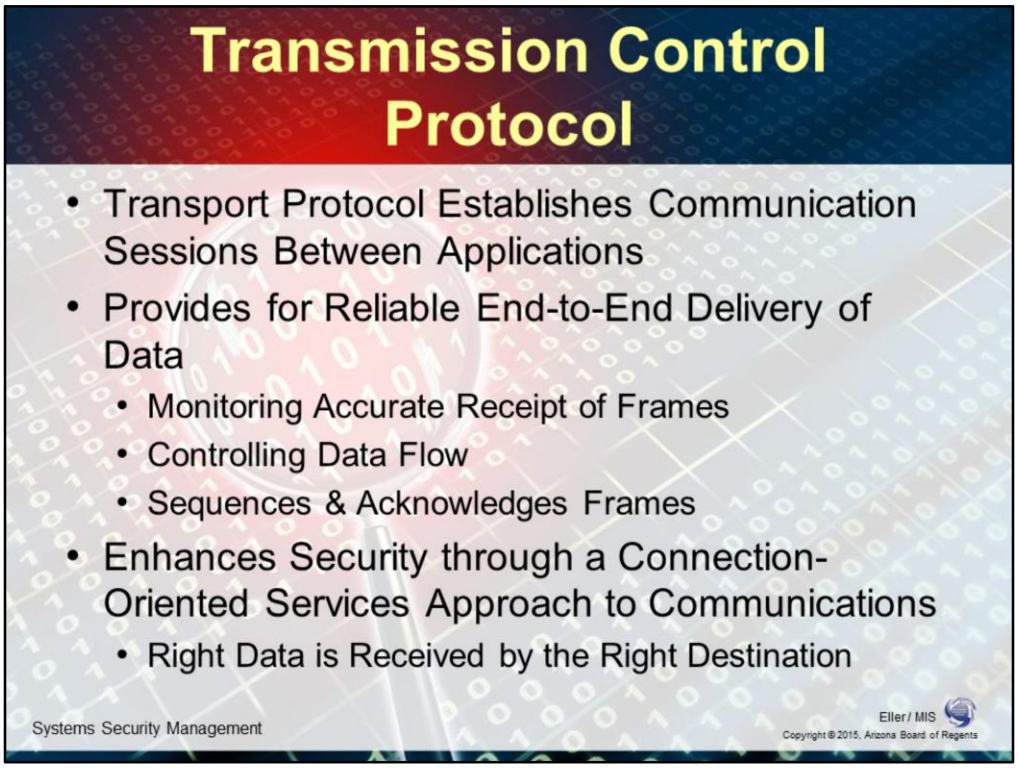
At the end of the module, you should be able to:

- Describe the TCP/IP protocol suite.
- Describe TCP, UDP, and IP.
- Describe IP Addressing and how it is used.
- Describe how a firewall works and why they are necessary on a network border.
- Describe the common network protocols and the ports they communicate on and how they can be affected by a firewall.
- Describe what packet filtering is and what is can be used for.
- Understand what network address translation is and how it can be used to secure a network.
- Understand how proxies and routers can be used to help secure a network.
- Describe the purpose of a demilitarized zone and how it can be used to further secure a network.
- Understand the differences between intrusion detection and prevention systems.



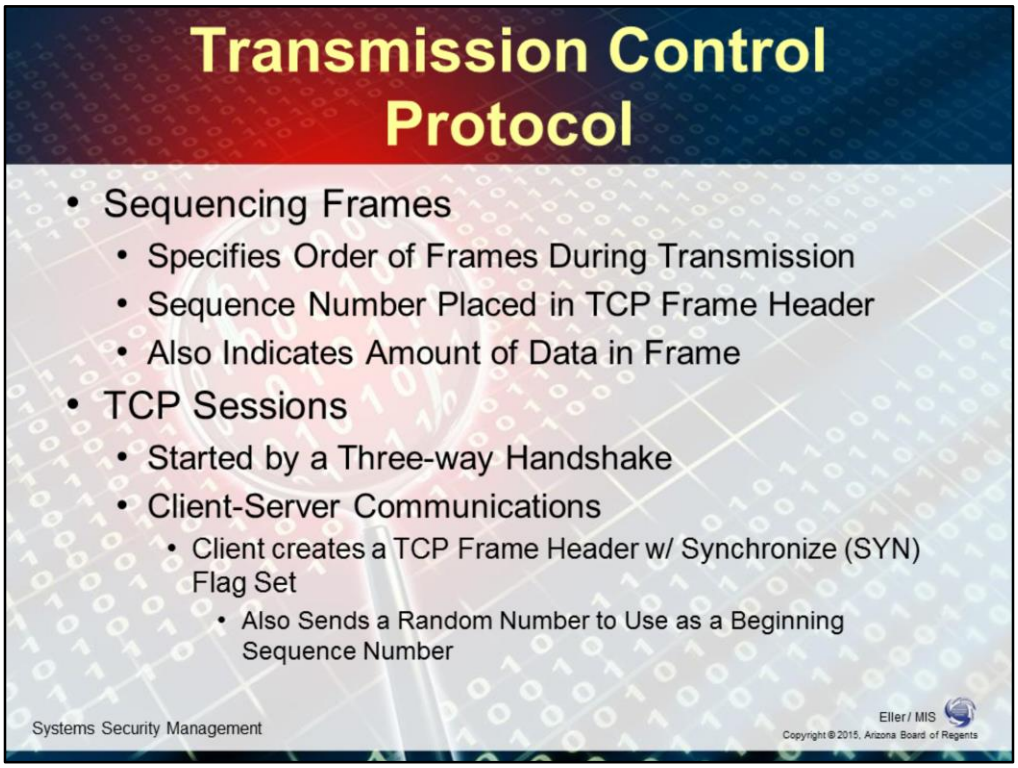
Overview of TCP/IP

The Transmission Control Protocol / Internet Protocol, or TCP/IP was introduced in the early 1970s and has since become the networking protocol of choice for Windows, Linux, and Mac OS X systems. The TCP/IP protocol suite enables thousands of networks to connect and communicate with the Internet. This protocol consists of 100 nonproprietary protocols, with the core components being made up by the Transmission Control Protocol (TCP), the User Datagram Protocol (UDP), and the Internet Protocol (IP).



Transmission Control Protocol

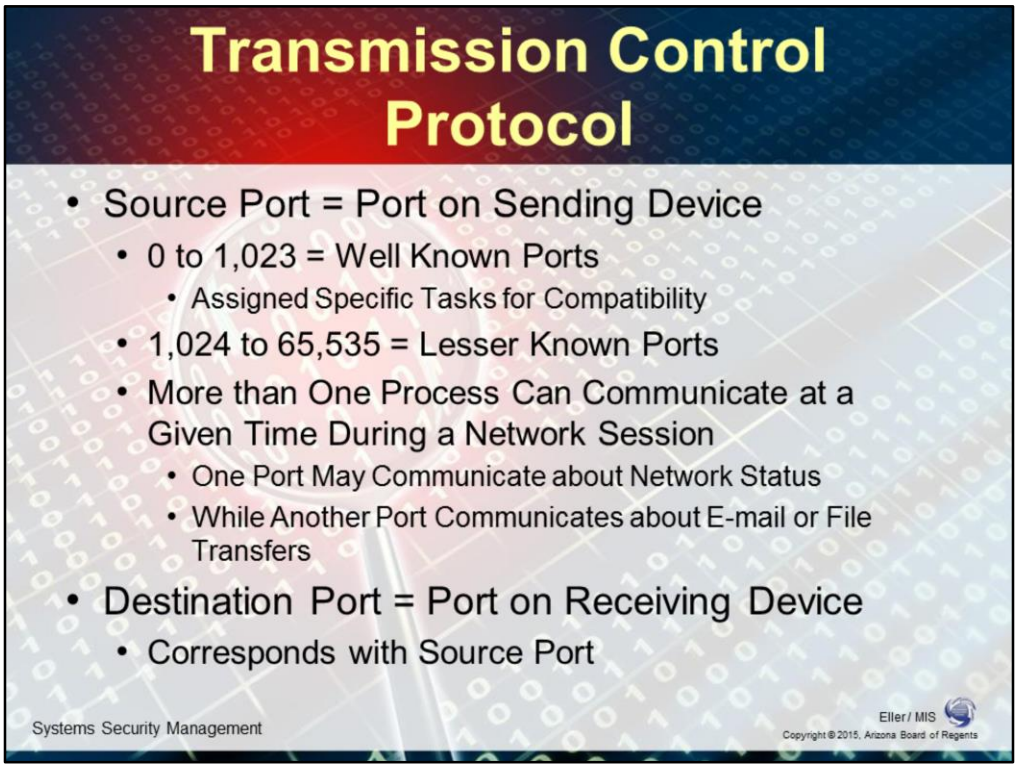
The Transmission Control Protocol, or TCP, is a transport protocol that is designed to establish communication sessions between applications. TCP provides for reliable end-to-end delivery of data across the Internet, monitoring accurate receipt of frames, controlling data flow, and by sequencing and acknowledging frames. TCP enhances security through a connection-oriented services approach to communications, ensuring only the right data is received by the right destination.



Transmission Control Protocol (continued)

Part of the responsibility of TCP involves accurate sequencing of data frames. TCP sequencing specifies the order of frames during data transmission. Each sequence number is placed in the TCP frame header, and they also are used to indicate the amount of data contained within the frame.

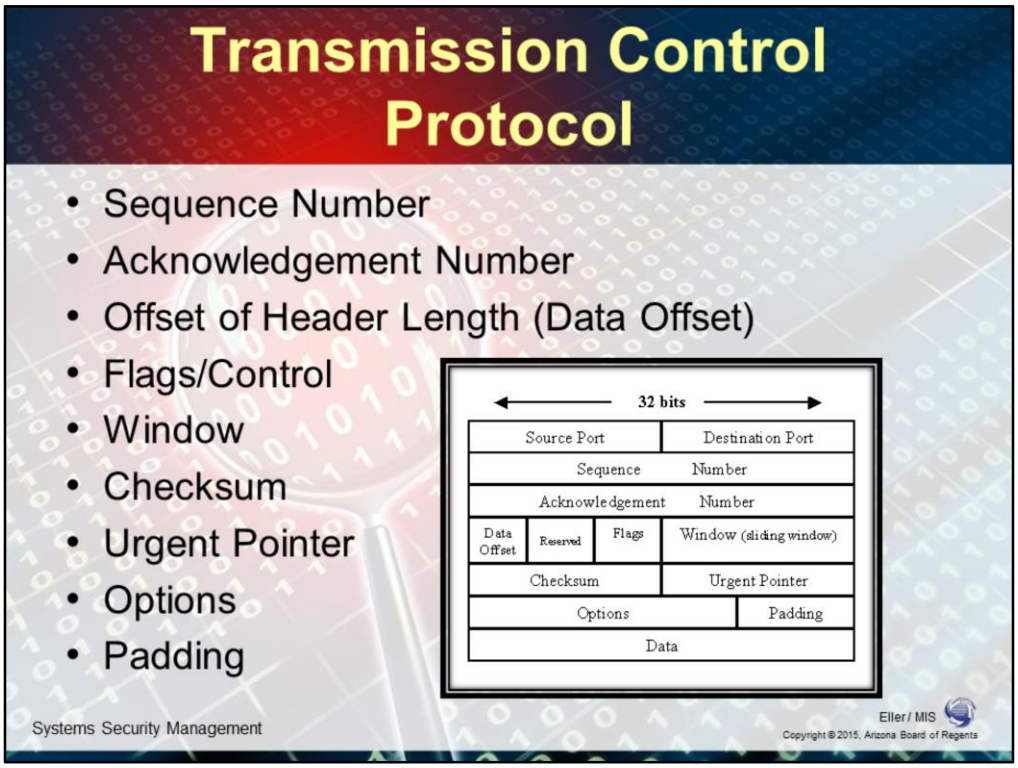
TCP sessions are started by a three-way handshake between client workstations and servers on the network. During this communication, the client creates a TCP frame header with the synchronize (SYN) flag set. This also sends a random number the server should use as the beginning sequence number for transmission.



Transmission Control Protocol (continued)

TCP communications is begun by specifying a source port, or the TCP port number used by the sending device. TCP ports numbered from zero to 1,023 are considered to be the well-known data ports and are assigned to specific tasks for application compatibility. TCP ports numbered from 1,024 to 65,535 are referred to as lesser known ports. This is because any application can utilize ports in this range for network communication. In addition, multiple ports and processes can communicate at a given time during a network session. One port may communicate about the status of the network while another port communicates about e-mail or file transfers in progress.

On the other end of the communication session is a defined destination port. This is a TCP port that is defined on the receiving device during communication and it corresponds directly with the source port.



Transmission Control Protocol (continued)

The TCP frame header actually contains a significant amount of information. Each frame that is sent during a communications session will contain a TCP frame header. The following information is transmitted in the header:

Sequence Number – Each frame in a transmission is assigned a 32-bit sequence number, which enables TCP to ensure all frames are received. Also used to identify duplicate frames & place frames back in proper order.

Acknowledgement Number – After checking the sequence number, TCP sends back an acknowledgement showing the frame was received. If an acknowledgement is not sent, the frame is retransmitted.

Offset of Header Length – Offset value indicates the length of the header, so that the start of the data portion of the frame can be quickly determined.

Flags/Control – Two of the flags in this frame area are used to show the beginning (SYN) and the end (FIN) of the complete data stream. Other flags are for control information, for example to reset the connection or to show that the urgent pointer field is in effect.

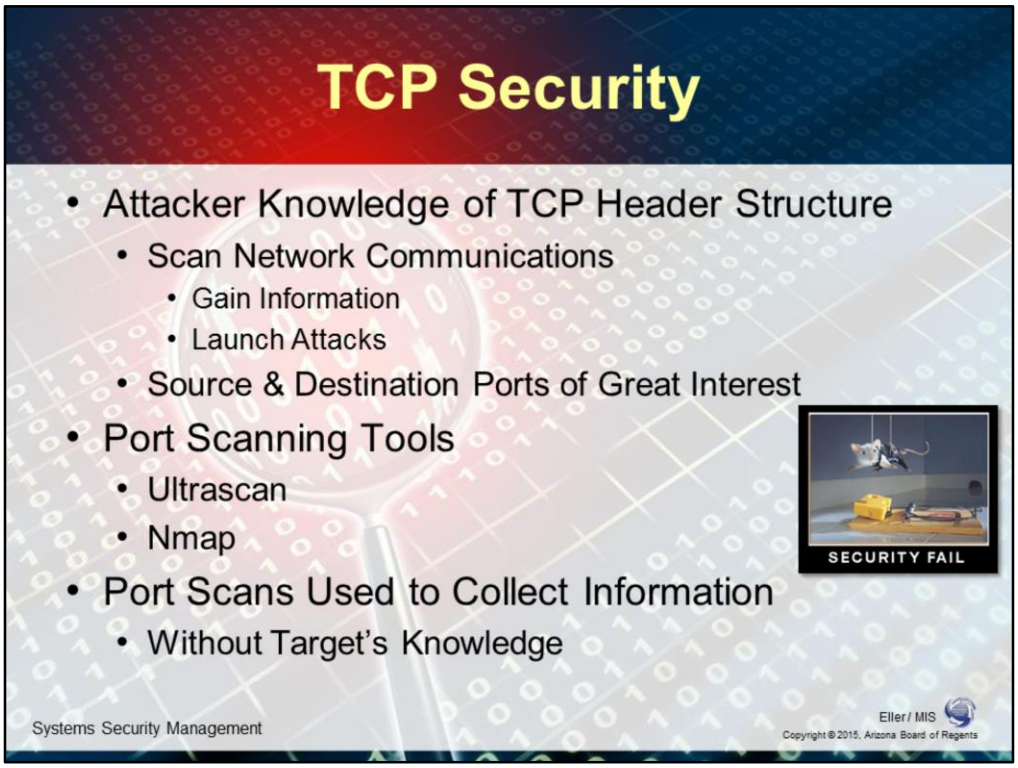
Window – This information works in conjunction with flow control. The window consists of the number of bytes that can be transmitted before the sender receives an acknowledgement of receipt. When the window size is reached, flow control is turned on to stop transmission until acknowledgement is received.

Checksum – The checksum is a 16-bit cyclic redundancy check (CRC) that is computed by adding the length of all header fields plus the length of the data payload field. The CRC checksum is placed in the frame by the sending station. The recipient calculates the checksum and compares its calculation with the value of the checksum field. If they are different, the frame is discarded, and the receiving station requests the frame be sent again. Added to the front of the checksum value are the source and destination addresses, which are the same as those contained in the IP header of the frame as a check that the frame was sent to the right destination.

Urgent Pointer – Provides a warning to the receiver that urgent data is coming, and points to the end of the urgent data within the sequence of the transmission of frames. Its purpose is to provide advance information about how much data is still to be received in a connected sequence of one or more frames.

Options – This area in the frame can hold additional information and flags about a transmission.

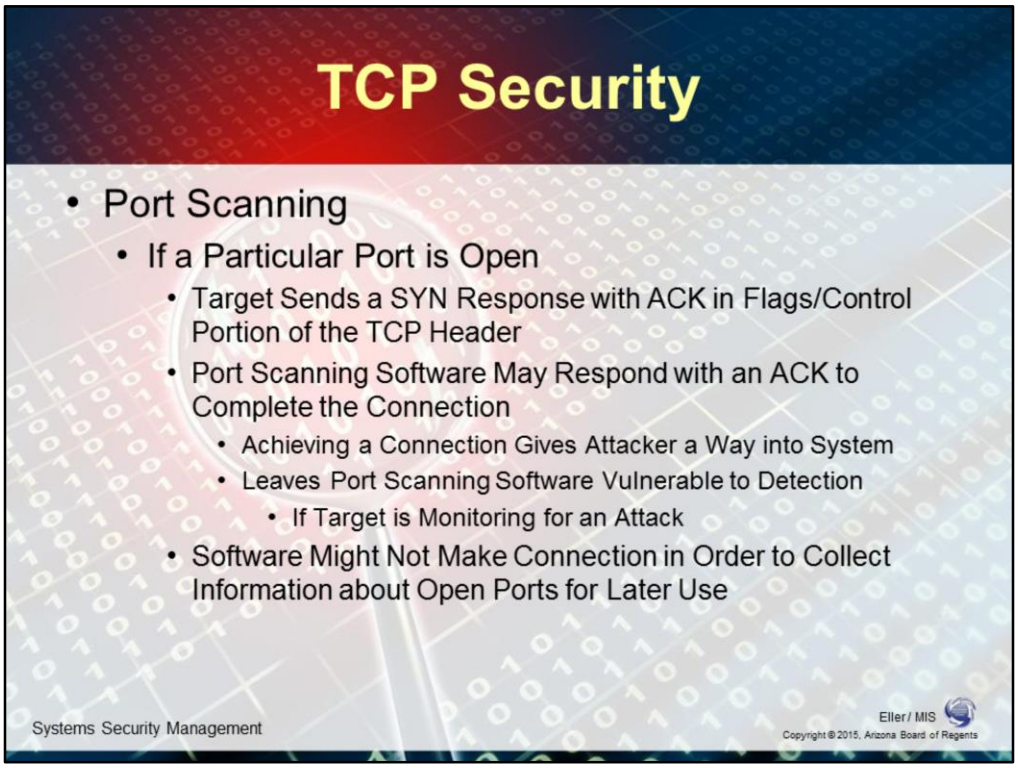
Padding – Padding area is used when there is too little or no optional data to complete the required header length, which must be divisible by 32.



TCP Security

It is absolutely vital for SysAdmins to ensure security when using TCP communications. The reason for this is because it is a well known protocol that is part of the backbone for Internet communications, so attackers will have complete knowledge of the TCP header structure. This allows attackers to scan network communications in order to gain information and launch attacks. The focus for most attackers is the header information pertaining to source and destination ports because these ports are generally left open in firewalls in order to allow for communication to occur.

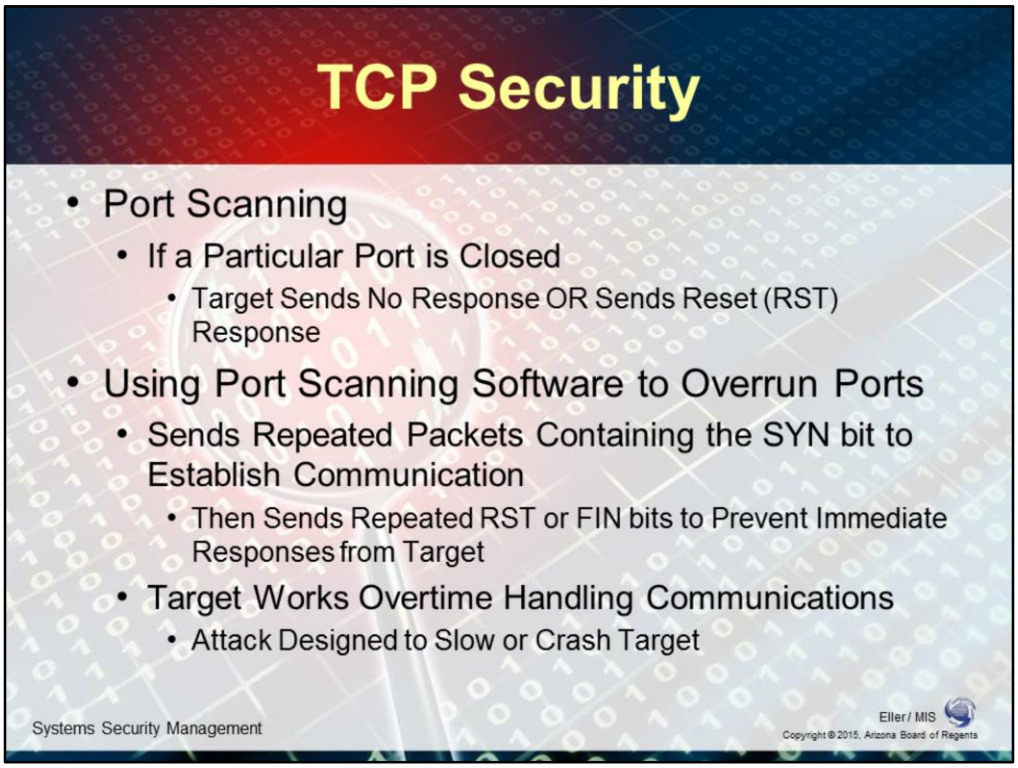
This header information can be easily intercepted through the use of port scanning tools such as Ultrascan and Nmap. In addition, port scans are used to collect information without the knowledge of the target or source.



TCP Security (continued)

With port scanning, the attacker is looking for ports which may be open in order to determine the best way to breach the system. In the event a specific port being scanned is open, the target will send a synchronize (SYN) response with acknowledgement (ACK) in the Flags/Control portion of the TCP frame header. The port scanning software may then respond with an ACK to complete the connection.

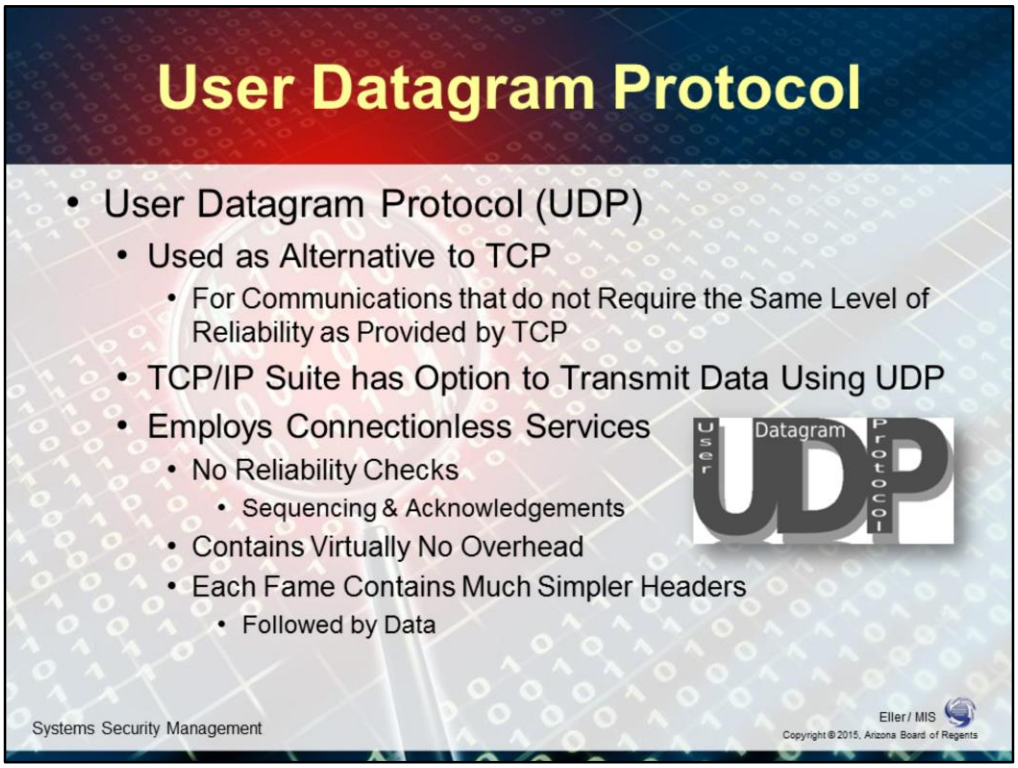
Achieving a connection with the target gives the attacker a way into the system; however, this leaves the port scanning software vulnerable to detection if the target is configured to monitor for attacks. In some cases, the port scanning software may not make a connection with the target system in order to simply collect information about the open ports for later use in an attack.



TCP Security (continued)

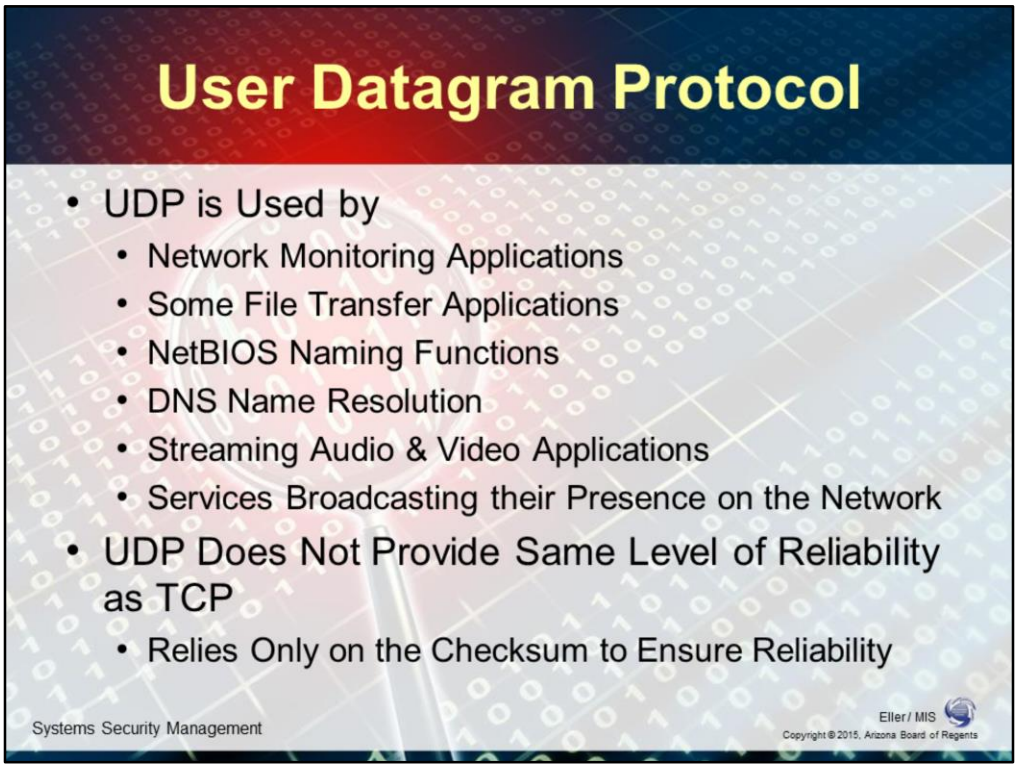
In the event a particular port is closed, the target will either send no response at all to the attacker or will send a reset (RST) response in order to close the connection.

Some attacks will actually use port scanning software in an effort to overrun any open ports. This means the attacker will send repeated packets containing the SYN bit in the header in order to establish communication. The attacker then sends repeated RST or FIN bits in an effort to prevent immediate responses from the target system. The target is then working overtime trying to handle all of the communications requests, allowing the attacker to either slow the system significantly or even cause the target system to crash. This is known as a DoS (or Denial of Service) attack.



User Datagram Protocol

The User Datagram Protocol, or UDP, is commonly used as an alternative to TCP, specifically for communications that do not require the same level of reliability as that provided by TCP. The TCP/IP suite has the option to transmit data using UDP, employing connectionless services. This means there are no reliability checks, including sequencing or acknowledgements. As a benefit, this means UDP contains virtually zero overhead when processing frames on the source and target. Finally, each frame will contain much simpler headers followed by the actual data.

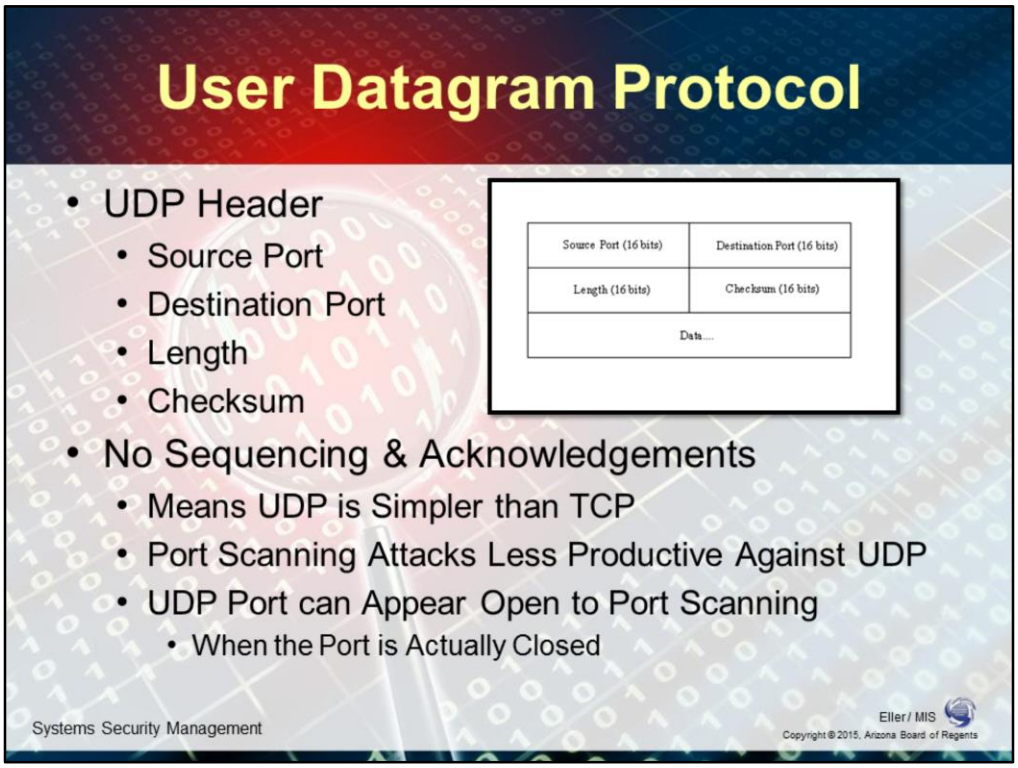


User Datagram Protocol (continued)

UDP is commonly used for communications by:

- Network Monitoring Applications
- Some File Transfer Applications
- NetBIOS Naming Functions
- DNS Name Resolution
- Streaming Audio & Video Applications
- Services Broadcasting their Presence on the Network

Again, UDP does not provide the same level of reliability as TCP as it relies only on the data checksum to ensure the reliability of communication.

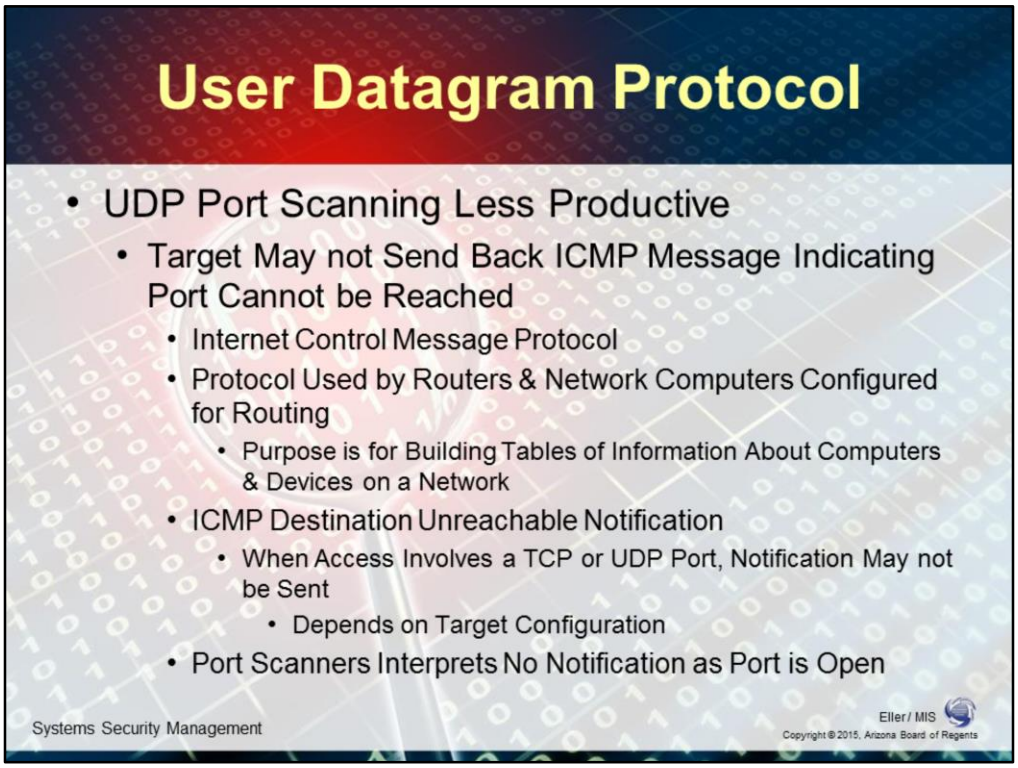


User Datagram Protocol (continued)

As with the TCP frame header, the UDP frame header contains the following four pieces of information:

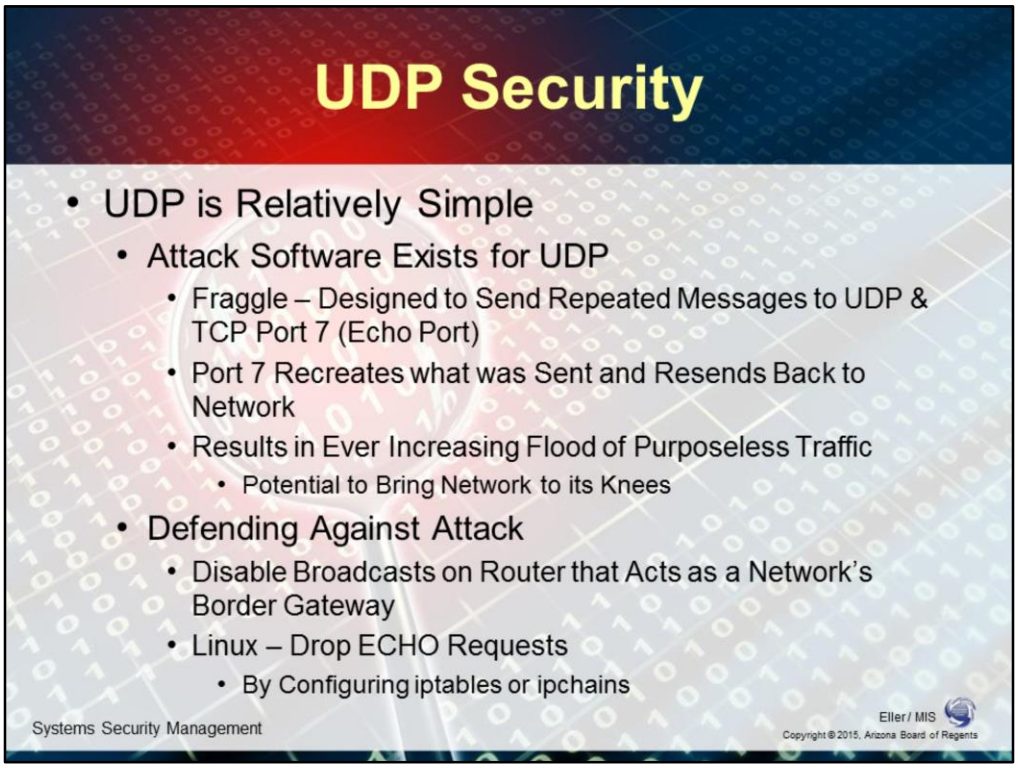
- Source Port
- Destination Port
- Header Length
- Checksum

As you can see this is considerably less information than what is contained in a TCP frame header. There are no sequencing and acknowledgements, making UDP much simpler than TCP. This also has the effect of making port scanning attacks less productive against UDP targets as there is less information to glean. In addition, a UDP port on a system can appear open to port scanning software when the port is actually closed, making it even less effective.



User Datagram Protocol (continued)

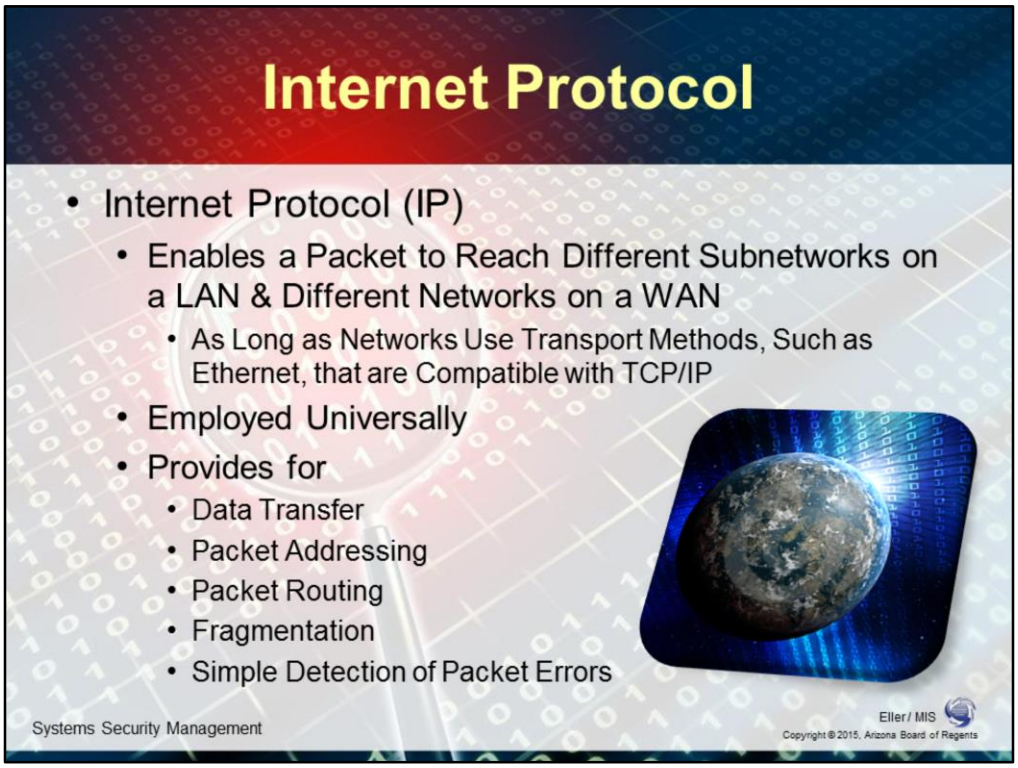
So, in general, UDP port scanning is less productive for an attacker. The target system may not send back an Internet Control Message Protocol, or ICMP, message indicating the port cannot be reached. This protocol is used by routers and network computers configured for routing. The purpose of this protocol is to allow the routers to build tables of information about computers and other devices on the network. Some routers may send an ICMP destination unreachable notification if the target of a UDP scan cannot be found. When target access involves a TCP or UDP port, the notification may not be sent back to the attacker, it depends on the target configuration. Because of this, many UDP port scanners will interpret no notification as the port is open when it is not.



UDP Security

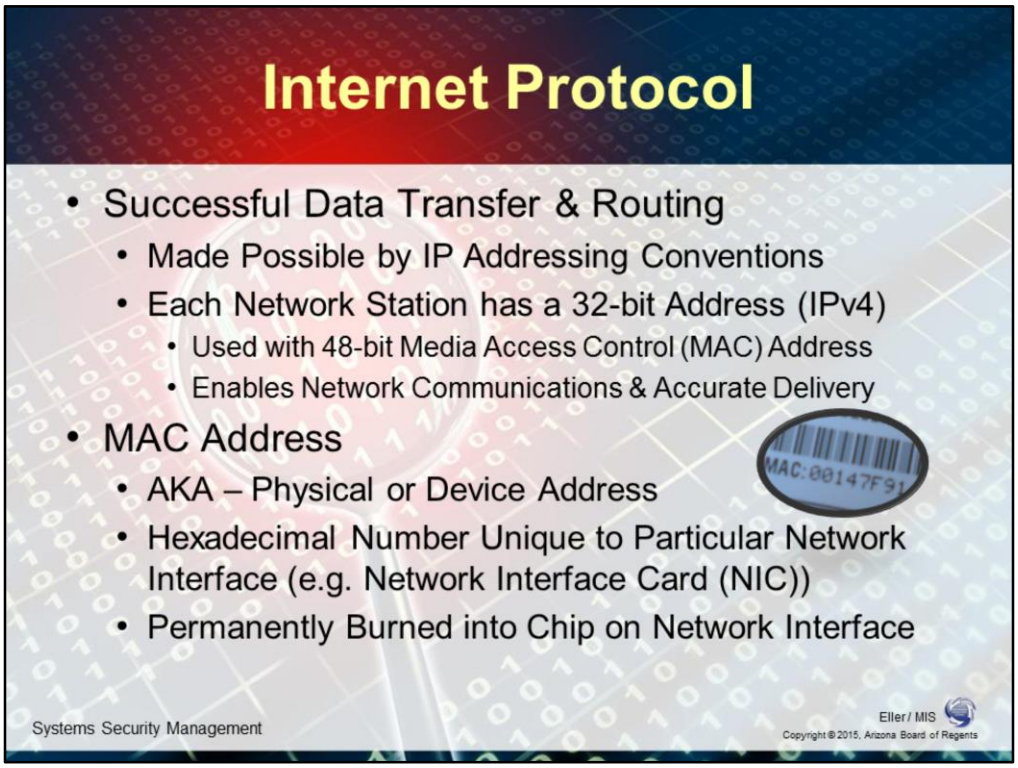
As you can see, UDP is a very simple network protocol; however, despite this, attack software does exist to take advantage of UDP. One such UDP scanner software is called Fraggle. This software is designed to send repeated messages to UDP and TCP port 7 (the echo port). Port 7 on the target system will recreate what was sent to it and resends that information to the entire network. This results in an ever increasing flood of purposeless traffic which has the potential to bring a network to its knees.

In order to defend against this attack, a SysAdmin should disable broadcasting on routers which act as a network's border gateway. Linux systems can be setup to drop ECHO requests from these ports by configuring iptables or ipchains.



Internet Protocol

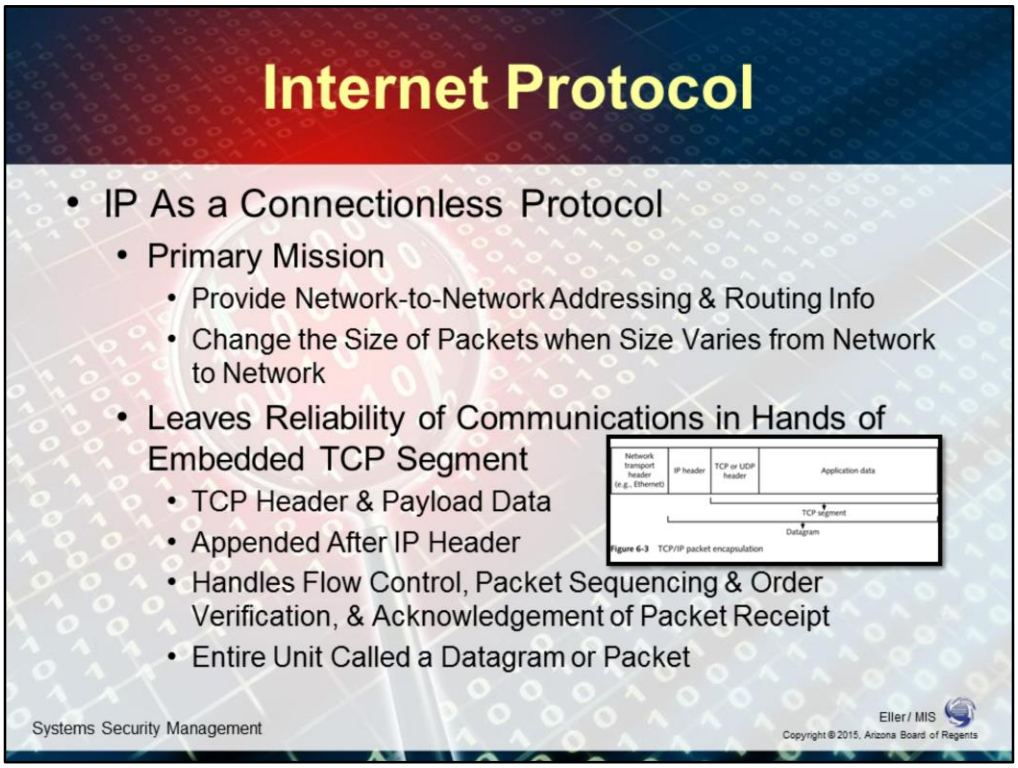
The Internet Protocol, or IP, is designed to enable a data packet to reach different subnetworks on a LAN or different networks on a WAN as long as the different networks use transport methods, such as Ethernet, that are compatible with TCP/IP. Internet Protocol is employed universally, providing for data transfer, packet addressing, packet routing, fragmentation, and a simple detection of packet errors.



Internet Protocol (continued)

The Internet Protocol allows for successful data transfer and routing, which is made possible by IP Addressing conventions. Each station on the network will have a 32-bit IP Address (referred to as IPv4) that is used in conjunction with a 48-bit Media Access Control (MAC) Addresses. These two addresses combined enables network communications and accurate delivery of data packets.

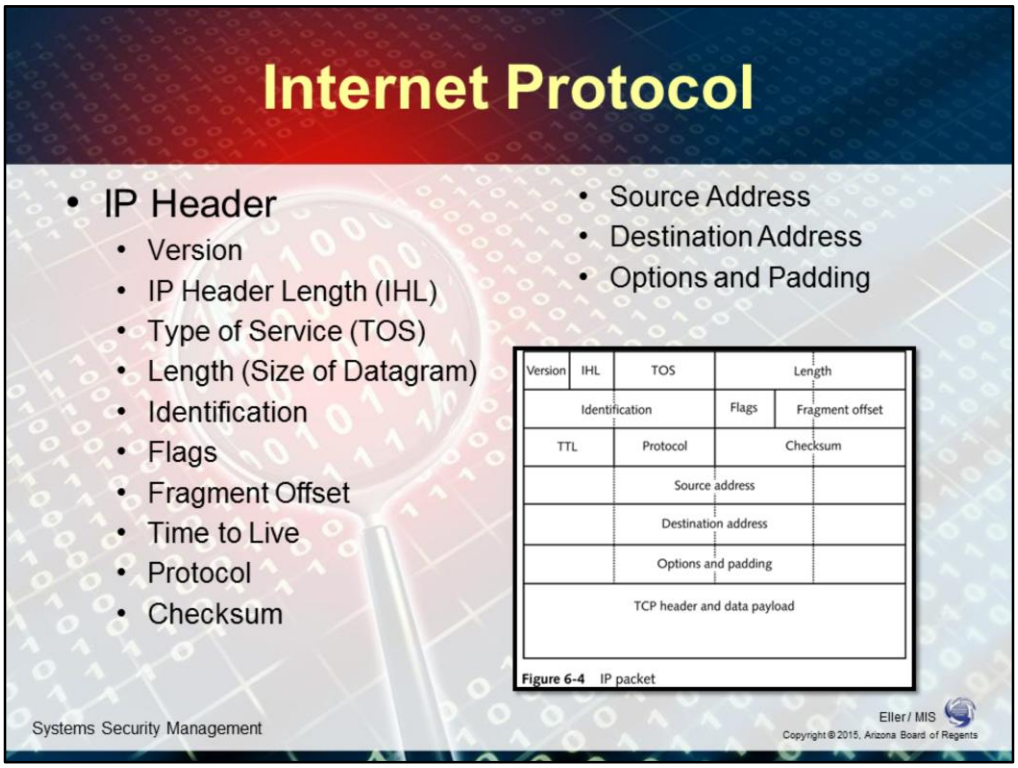
The MAC Address is also called the Physical or Device Address, and is a hexadecimal number that is unique to a particular network interface, such as a Network Interface Card, or NIC. This address is permanently burned into a chip on the network interface in order to identify the device.



Internet Protocol (continued)

When using the Internet Protocol as a connectionless protocol, the primary mission is to provide network-to-network addressing and routing information. This way the network routers can change the size of network packets when transmitting from network to network.

This then leaves the reliability of communications in the hands of the embedded TCP segment. The TCP segment contained in the IP data packet includes the TCP frame header plus the payload data, both of which are appended after the IP header. This information handles the flow control, packet sequencing and order verification, and an acknowledgement of packet receipts. This makes the entire combined unit of data what is referred to as a datagram or packet.



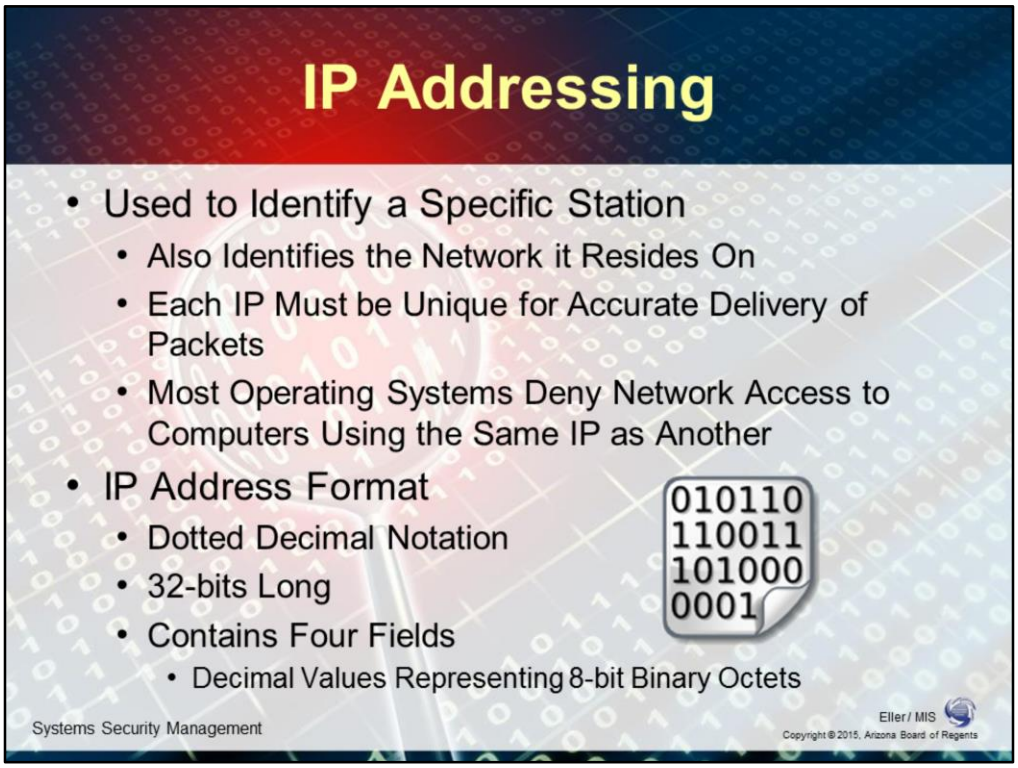
Internet Protocol (continued)

The IP Header is actually quite complex by itself.

- **Version** (always set to the value 4 in the current version of IP)
- **IP Header Length** (number of 32 -bit words forming the header, usually five)
- **Type of Service (ToS)**, now known as Differentiated Services Code Point (DSCP) (usually set to 0, but may indicate particular Quality of Service need from the network, the DSCP defines the way routers should queue packets while they are waiting to be forwarded).
- **Size of Datagram** (in bytes, this is the combined length of the header and the data)
- **Identification** (16-bit number which together with the source address uniquely identifies this packet - used during reassembly of fragmented datagrams)
- **Flags** (a sequence of three flags (one of the 4 bits is unused) used to control whether routers are allowed to fragment a packet (i.e. the Don't Fragment, DF, flag), and to indicate the parts of a packet to the receiver)
- **Fragmentation Offset** (a byte count from the start of the original sent packet, set by any router which performs IP router fragmentation)
- **Time To Live** (Number of hops /links which the packet may be routed over, decremented by most routers - used to prevent accidental routing loops)
- **Protocol** (Service Access Point (SAP) which indicates the type of transport packet being carried (e.g. 1 = ICMP; 6 = TCP; 17= UDP).

- **Header Checksum** (inserted by the sender and updated whenever the packet header is modified by a router - Used to detect processing errors introduced into the packet inside a router or bridge where the packet is not protected by a link layer cyclic redundancy check. Packets with an invalid checksum are discarded by all nodes in an IP network)
- **Source Address** (the IP address of the original sender of the packet)
- **Destination Address** (the IP address of the final destination of the packet)
- **Options** (not normally used, but, when used, the IP header length will be greater than five 32-bit words to indicate the size of the options field)

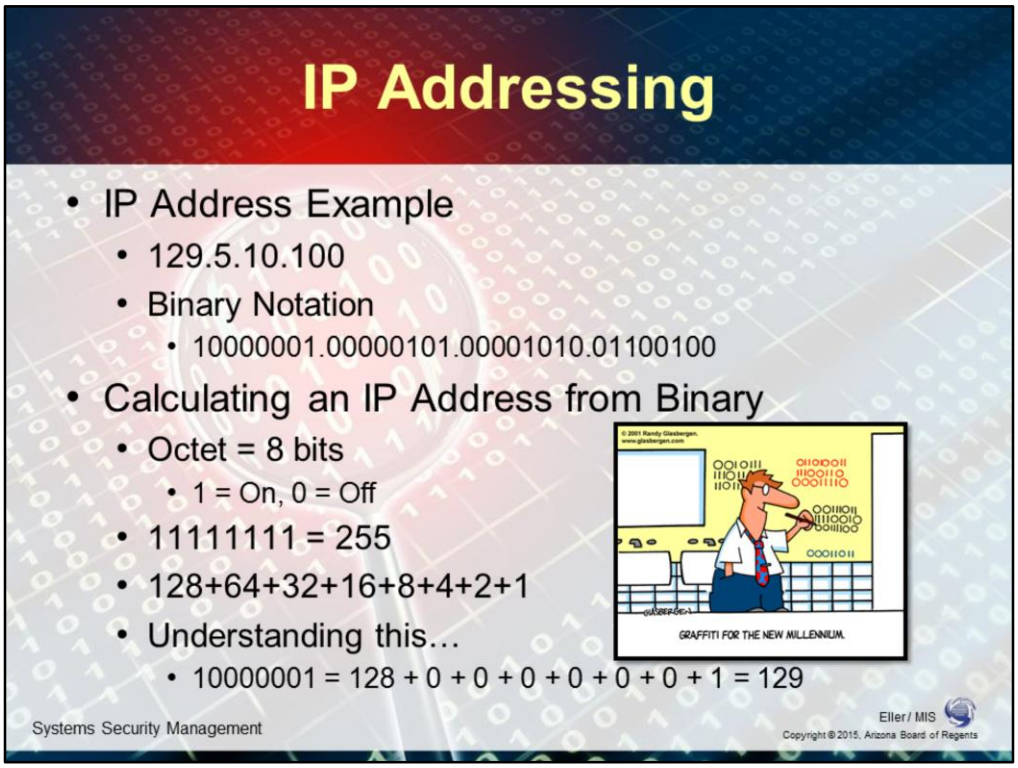
At the end of the IP header is the embedded TCP or UDP header for the data being transmitted.



IP Addressing

IP Addresses, or IPs, are used to identify specific stations on a network as well as the network the station resides on. Each IP must be unique in order to provide for accurate delivery of data packets. Most operating systems will actively deny network access to computers using the same IP as another system already on the network.

IP Addresses are formatted using a dotted decimal notation that is 32-bits long. IPs contain four fields and the decimal values represent 8-bit binary octets. Let me explain…

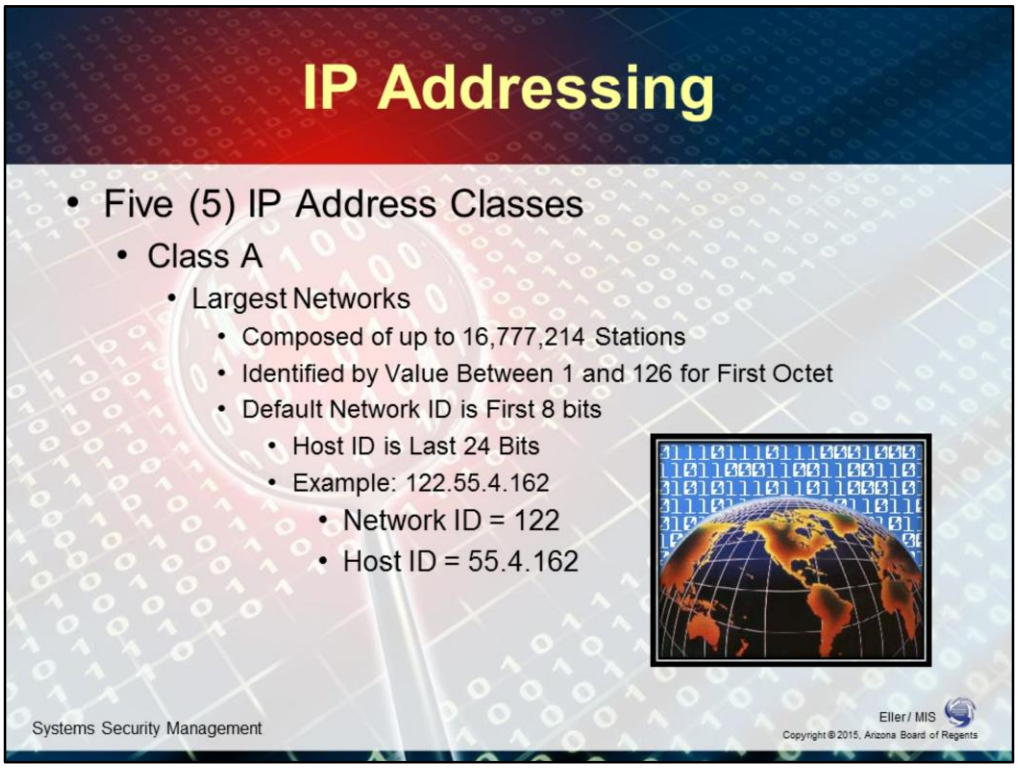


IP Addressing (continued)

Let's say you have the IP Address 129.5.10.100. This is the dotted decimal notation of the IP. When converting this number to binary notation (or ones and zeros) this IP Address will be 10000001.00000101.00001010.01100100. So, how do you figure this conversion out?

Calculating an IP Address from Binary

As mentioned there are four decimals separated by dots. Each decimal is called an octet because the decimal is representative of eight (8) ones and zeros. In the octet, a one represents the on position, while the zero represents the off position. If you were to take an octet made up entirely of ones, or 11111111, this would equal a total decimal value of 255. This is calculated by adding the decimal value of the position of each bit. In other words, a one in the left-most position would be equal to a decimal value of 128. A one in the second position would be equal to 64, the third position would be 32, and so on. Understanding this, an octet of 10000001 would require one to add the value of each bit in the octet. This would yield a total value of 128+0+0+0+0+0+0+1 or 129.

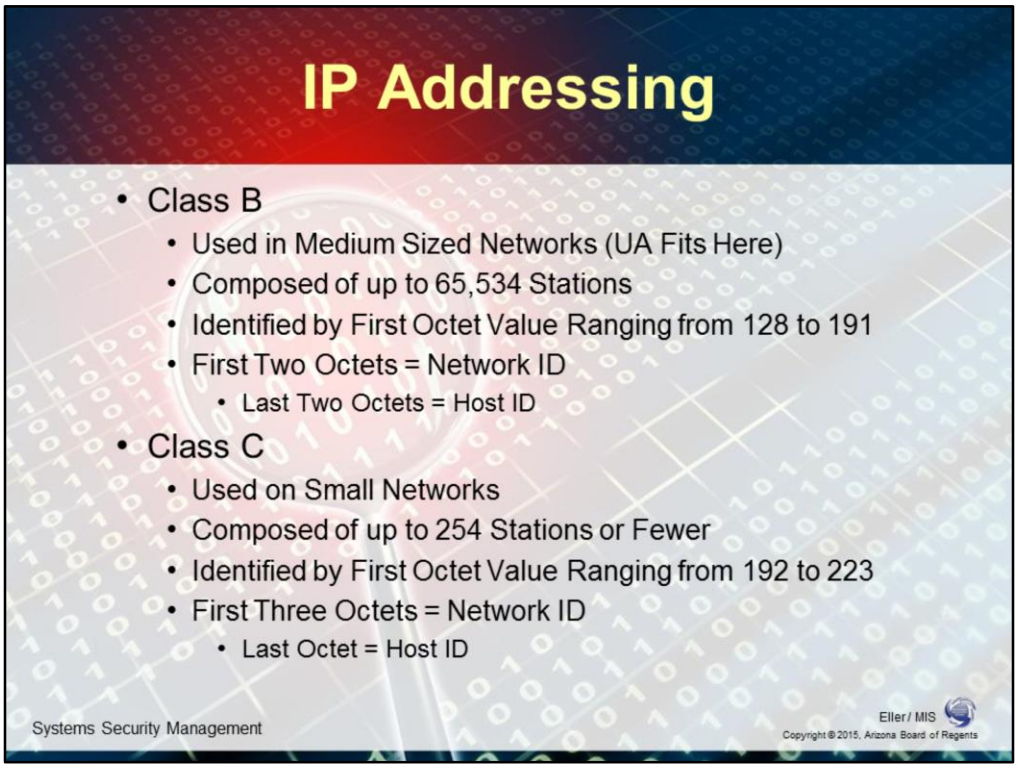


IP Addressing (continued)

There are five IP Address classes used in network communications: Class A through Class E.

Class A

Class A networks are the largest networks, composed of up to 16,777,214 total stations. These networks are identified by a value between 1 and 126 for the first octet of the address. In Class A networks, the default network ID is the first 8 bits, while the host ID is the last 24 bits in the IP address. So, if your IP address is 122.55.4.162, the Network ID is 122 while the Host ID is 55.4.162.



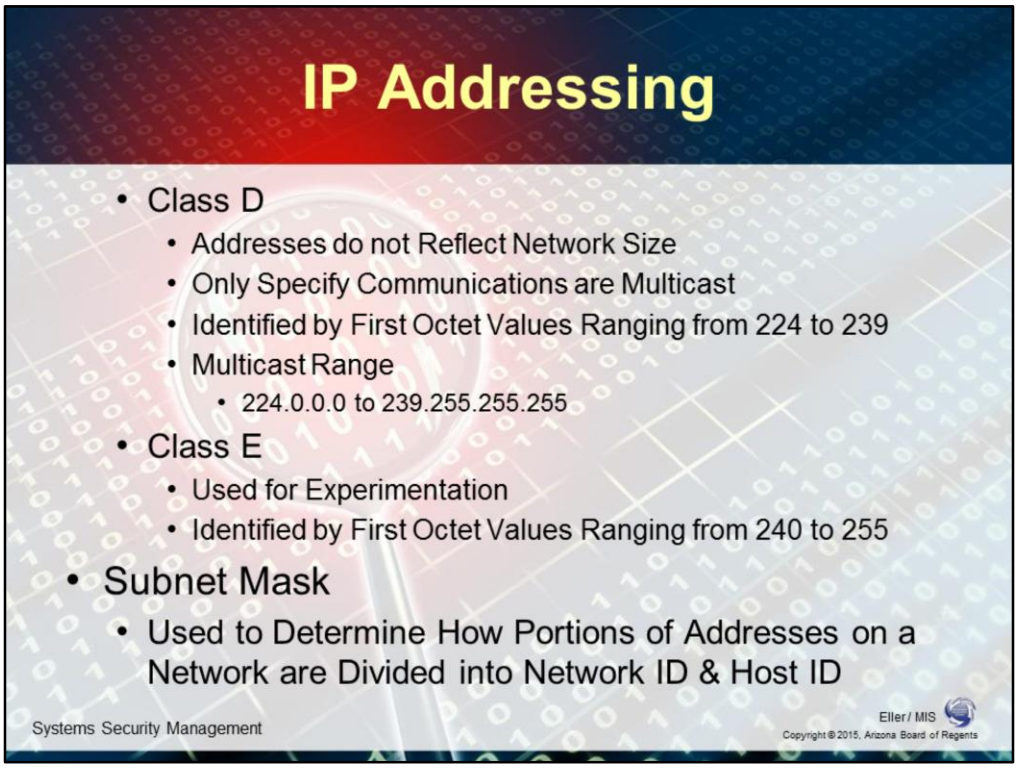
IP Addressing (continued)

Class B

Class B networks are used in medium sized networks, this is where the University of Arizona fits. This class of IPs is composed of up to 65,534 total stations that are identified by the first octet value being in the range of 128 to 191. In this instance, the first two octets will be the Network ID and the last two octets will make up the Host ID.

Class C

Class C networks are used by small networks that are comprised of up to 254 total addresses or fewer. These networks are identified by the first octet value ranging from 192 to 223 and the first three octets will make up the Network ID with the remaining octet making up the Host ID.



IP Addressing (continued)

Class D

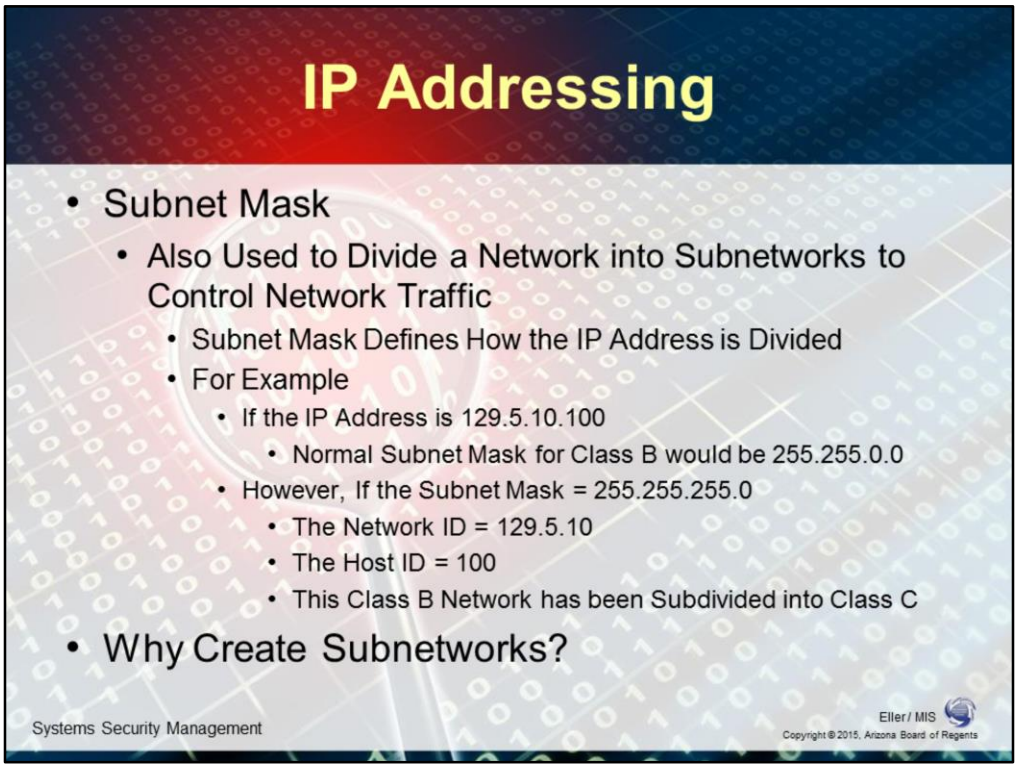
Class D network addresses will not reflect the actual size of the network, as they only specify the communications are multicast-type communication. Class D networks are identified by the first octet values ranging from 224 to 239, with the multicast range extending from 224.0.0.0 to 239.255.255.255.

Class E

Class E network addresses are only used for experimentation purposes. These networks are identified by the first octet values ranging from 240 to 255.

Subnet Masks

In some cases, it may be necessary for Network Admins to break a network up into multiple pieces. The Subnet Mask is used to determine how different portions of Addresses on a network are divided into the Network ID and Host ID.



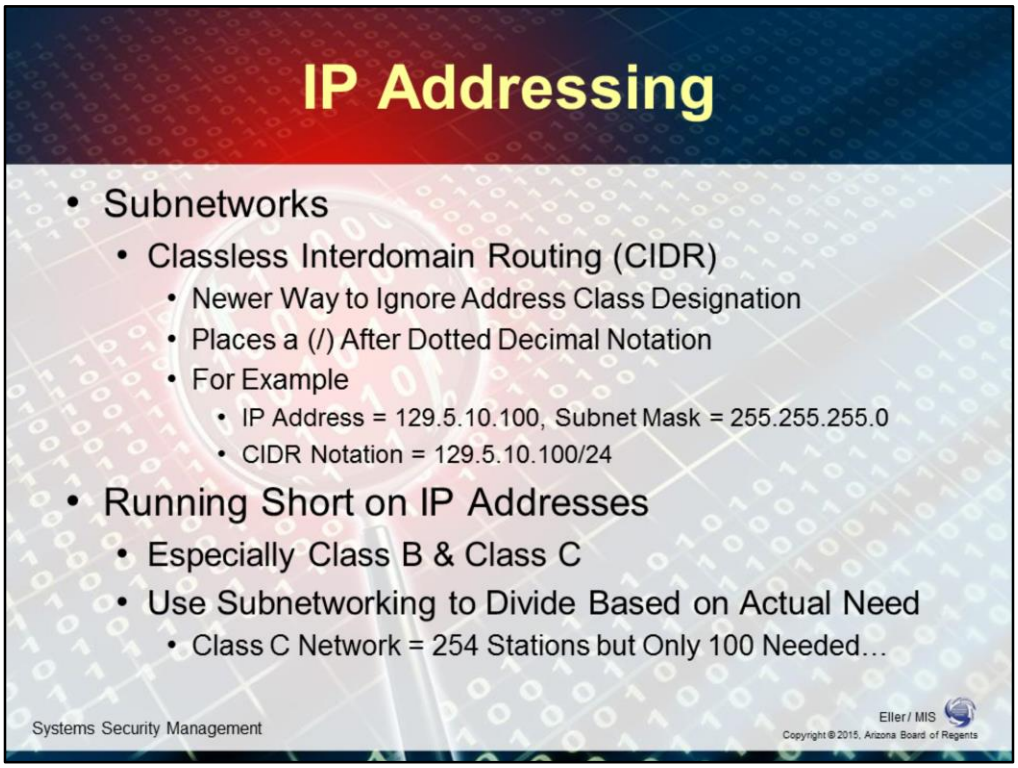
IP Addressing (continued)

Subnet Mask (continued)

Subnet Masks are also used to divide a network into subnetworks in order to help control network traffic by defining how the IP Address is divided. For example:

If your IP Address is 129.5.10.100, this would be a normal IP Address on a Class B network. The expected Subnet Mask for this IP would be 255.255.0.0. However, if the Subnet Mask were changed to 255.255.255.0 then this would effectively change the Network ID to 129.5.10 and the Host ID would be 100. This means the Class B network has been subdivided into a Class C network in order to control traffic flow.

So why create subnetworks? The answer is simply to better segment a larger network class for control purposes.



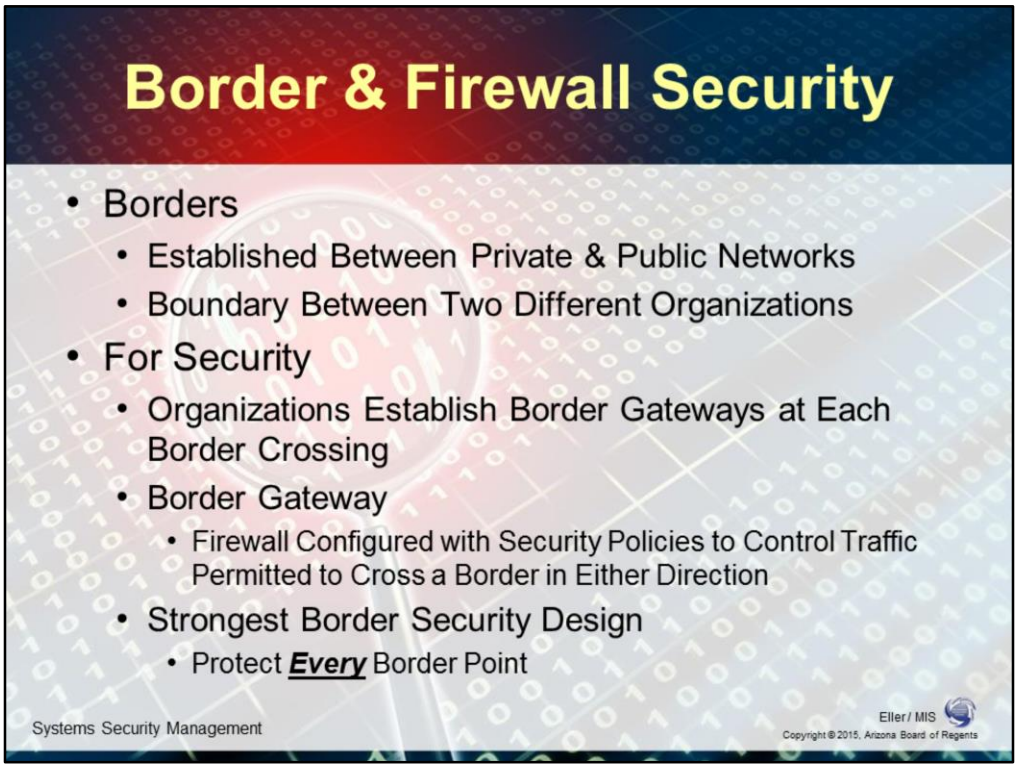
IP Addressing (continued)

Subnetworks

A newer way of defining a Subnet Mask is through the use of Classless Interdomain Routing, or CIDR. This allows Network Managers to ignore Address Class designations. CIDR notation involves placing a forward slash (/) after the dotted decimal notation, with the number after the slash indicating the number of bits in the Subnet Mask. so for example:

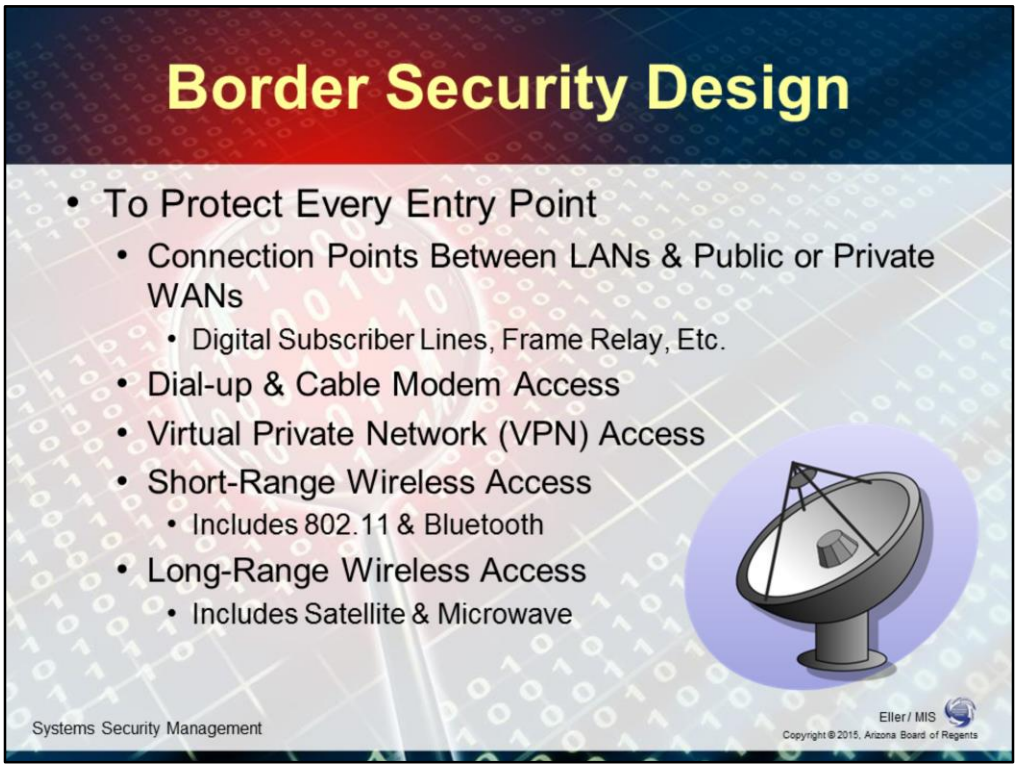
If your IP address is 129.5.10.100 with a Subnet Mask of 255.255.255.0, the CIDR notation will be 129.5.10.100/24.

The Internet is running short on IP Addresses, especially addresses that are designated as Class B and Class C. Because of this, many network managers are taking their existing network address ranges and using subnetworking techniques to divide the networks based on actual need. So, for example, they might have a Class C network with 254 possible stations, but may only need 100 total stations, so the Admin could subdivide the network to free up additional IP Addresses for other uses.



Border and Firewall Security

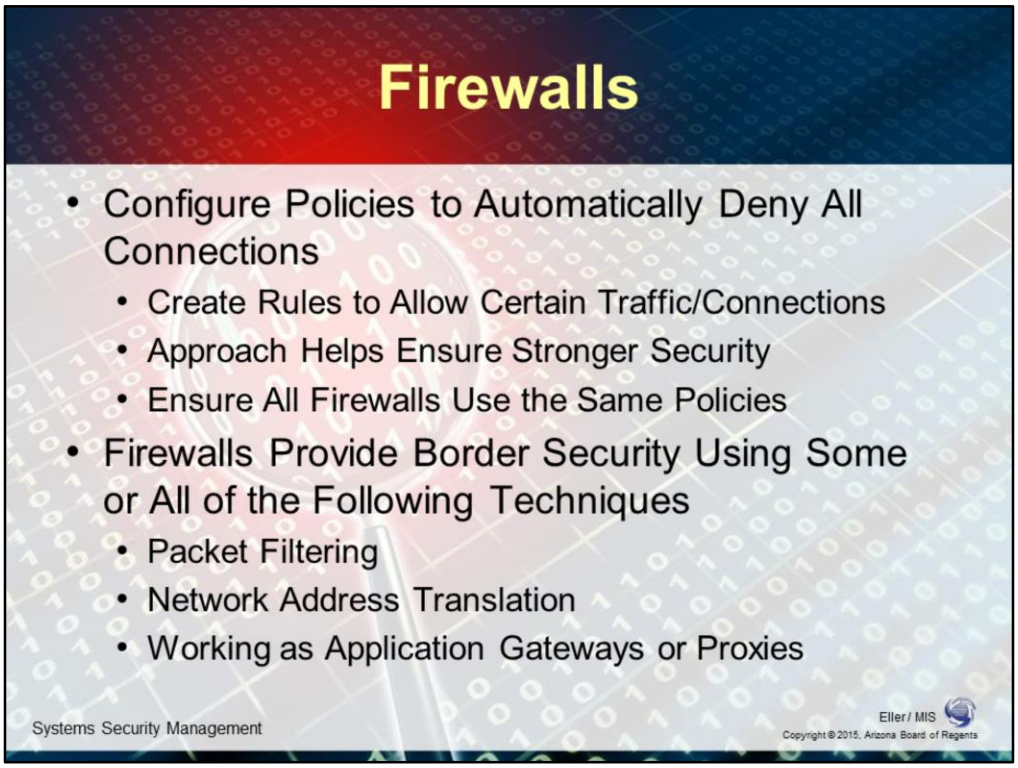
A network border is the boundary that is established between private and public networks or the boundary between two different organizations. For security purposes, organizations should establish border gateways at each border crossing. A border gateway is a firewall configured with security policies to control traffic that is permitted to cross a border in either direction. Remember, the strongest border security design is one that protects EVERY border point.



Border Security Design

It is important to protect every entry point to a network. If a single entry point were to be left open, the network will be vulnerable to attack. This includes connection points between LANs and public or private WANs, such as digital subscriber lines and frame relays, among others. Border security should also include protecting the network when dial-up and cable modem access is used or when virtual private networks, or VPNs, are used.

Other avenues considered to be a network border include both short-range and long-range wireless access. Short-range access would include technologies such as 802.11 networks and Bluetooth. Long-range access would include satellite or microwave communications.



Firewalls

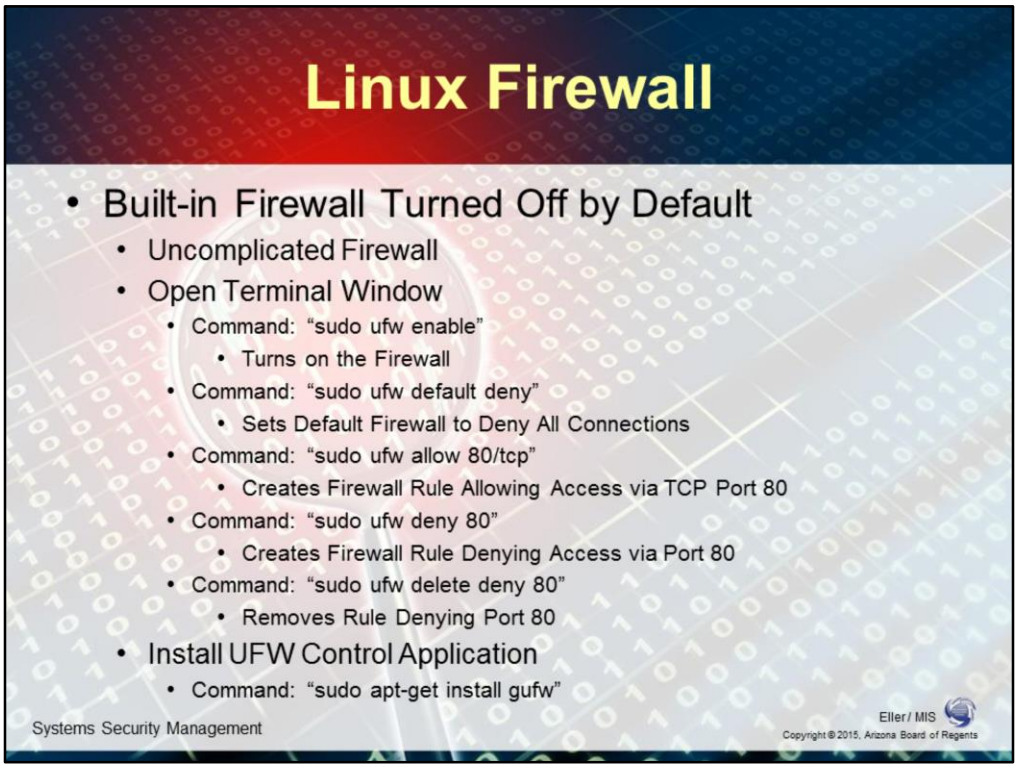
Firewalls are hardware or software designed to block access to a network. A SysAdmin can configure a firewall to enact policies that will automatically deny all connections to a network or system. From this point a SysAdmin can create various rules designed to allow or deny certain network traffic or connection types. This approach helps to ensure stronger security, and it is very important when implementing multiple firewalls that they all are configured to use the same policies.

Firewalls help to provide border security through the use of several security techniques, including: packet filtering, network address translation, and configuring the firewall to work as an application gateway or proxy.

Windows Firewall

In Windows Server, the built-in Windows Firewall is turned on by default. For Windows Server 2008 and 2008 R2, the Firewall is available in two locations: a basic configuration in the Control Panel called Windows Firewall, and an Advanced Configuration available under Administrative Tools called Windows Firewall with Advanced Security. The latter option will allow you to configure firewall rules for domain-level access, private network access, and public network access.

As an alternative, if you are not comfortable with configuring a firewall, you can run the Security Configuration Wizard in Administrative Tools. This is a wizard which will guide you through a security configuration. It will analyze the server and recommend security settings based on the services you have installed. The wizard will then give you an opportunity to make modifications to the recommendations and apply the changes to secure the server.

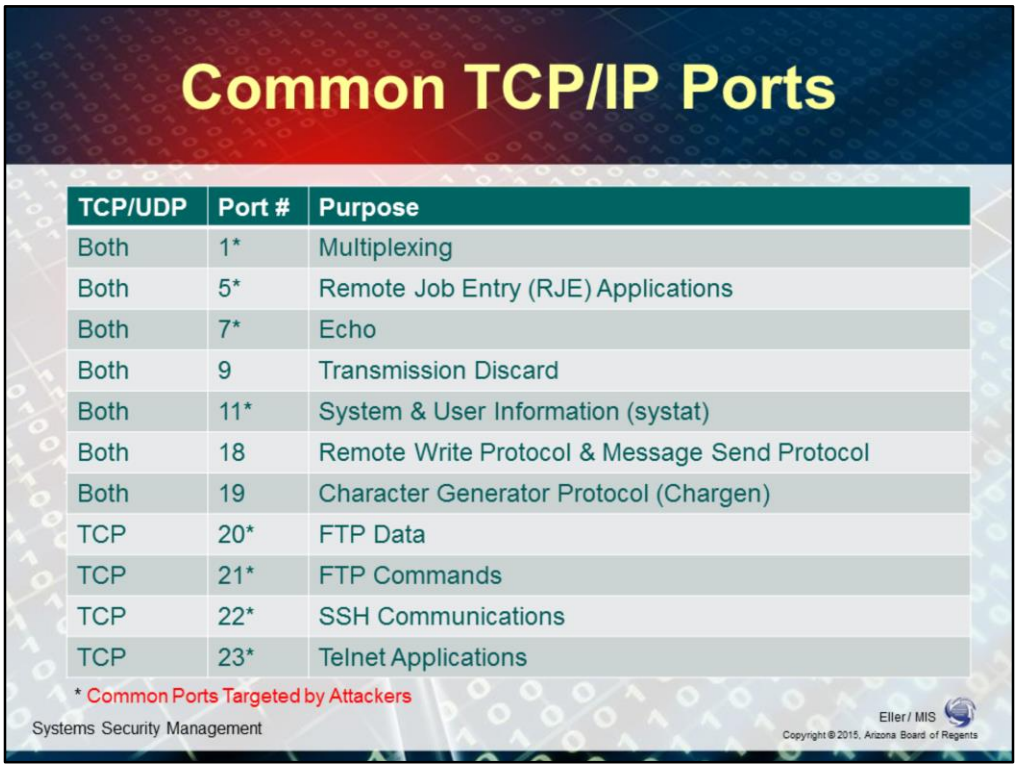


Linux Firewall

Whether or not your Linux installation has a firewall and if it does whether or not it is turned on by default will depend entirely upon the version of Linux you are running. One of the most common built-in firewalls for Linux is called the Uncomplicated Firewall and it is turned off by default on most instances. In order to configure this firewall, you will need to open a terminal window and type a series of commands.

- The command "sudo ufw enable" will turn on the firewall.
- The command "sudo ufw default deny" will set the default behavior of the firewall to deny all connection attempts.
- The command "sudo ufw allow 80/tcp" will create a firewall rule that allows access to the system via TCP port 80.
- The command "sudo ufw deny 80" will create a firewall rule that denies access to both TCP and UDP port 80.
- The command "sudo ufw delete deny 80" will remove the firewall rule that denies access to both TCP and UDP port 80.

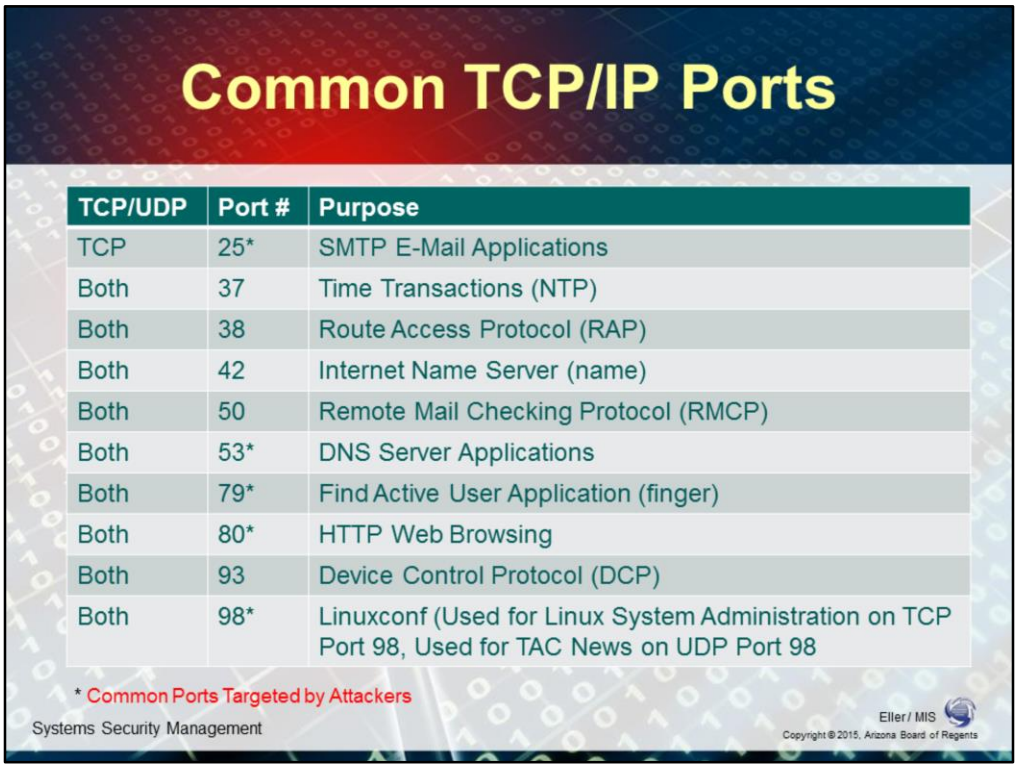
You can also install an application to help you control the Uncomplicated Firewall through a graphical user interface. This is done with the command "sudo apt-get install gufw."



Common TCP/IP Ports

TCP/UDP	Port #	Purpose
Both	1*	Multiplexing
Both	5*	Remote Job Entry (RJE) Applications
Both	7*	Echo
Both	9	Transmission Discard
Both	11*	System & User Information (systat)
Both	18	Remote Write Protocol & Message Send Protocol
Both	19	Character Generator Protocol (Chargen)
TCP	20*	FTP Data
TCP	21*	FTP Commands
TCP	22*	SSH Communications
TCP	23*	Telnet Applications

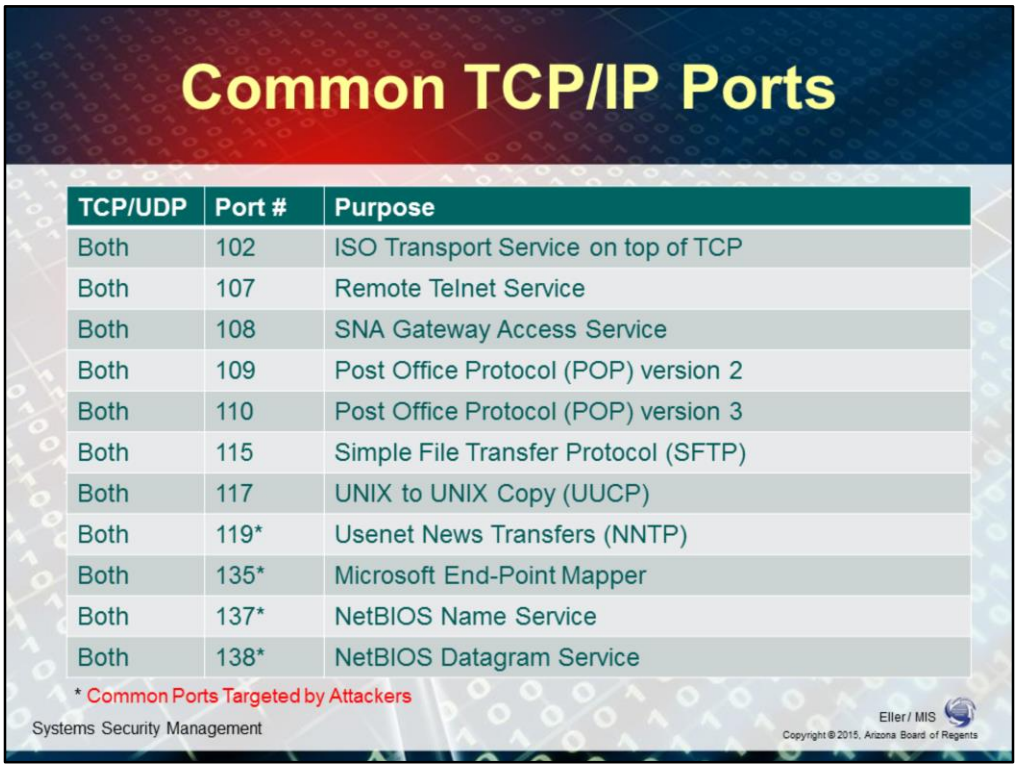
Out of these ports, the most common ports for attackers to target include TCP/UDP ports 1, 5, 7, 11, and TCP ports 20-23.



Common TCP/IP Ports (continued)

TCP	25*	SMTP E-Mail Applications
Both	37	Time Transactions (NTP)
Both	38	Route Access Protocol (RAP)
Both	42	Internet Name Server (name)
Both	50	Remote Mail Checking Protocol (RMCP)
Both	53*	DNS Server Applications
Both	79*	Find Active User Application (finger)
Both	80*	HTTP Web Browsing
Both	93	Device Control Protocol (DCP)
Both	98*	Linuxconf (Used for Linux System Administration on TCP Port 98, Used for TAC News on UDP Port 98

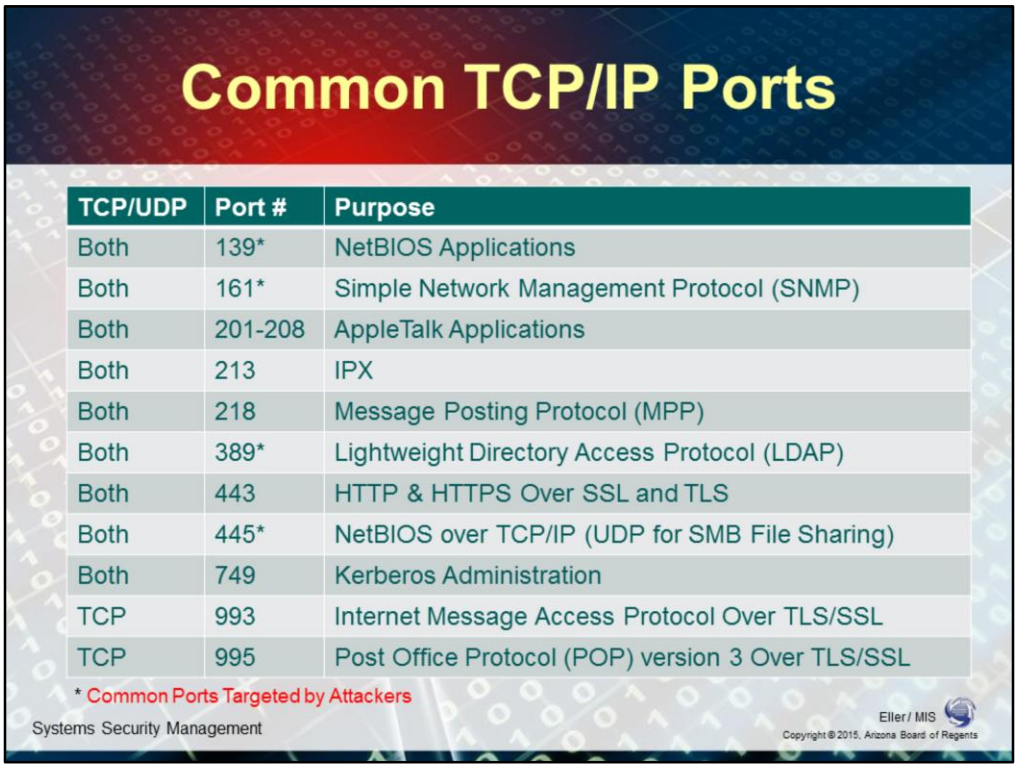
Out of these ports, the most common ports for attackers to target include TCP Port 25 and TCP/UDP ports 53, 79, 80, and 98.



Common TCP/IP Ports (continued)

TCP/UDP	Port #	Purpose
Both	102	ISO Transport Service on top of TCP
Both	107	Remote Telnet Service
Both	108	SNA Gateway Access Service
Both	109	Post Office Protocol (POP) version 2
Both	110	Post Office Protocol (POP) version 3
Both	115	Simple File Transfer Protocol (SFTP)
Both	117	UNIX to UNIX Copy (UUCP)
Both	119*	Usenet News Transfers (NNTP)
Both	135*	Microsoft End-Point Mapper
Both	137*	NetBIOS Name Service
Both	138*	NetBIOS Datagram Service

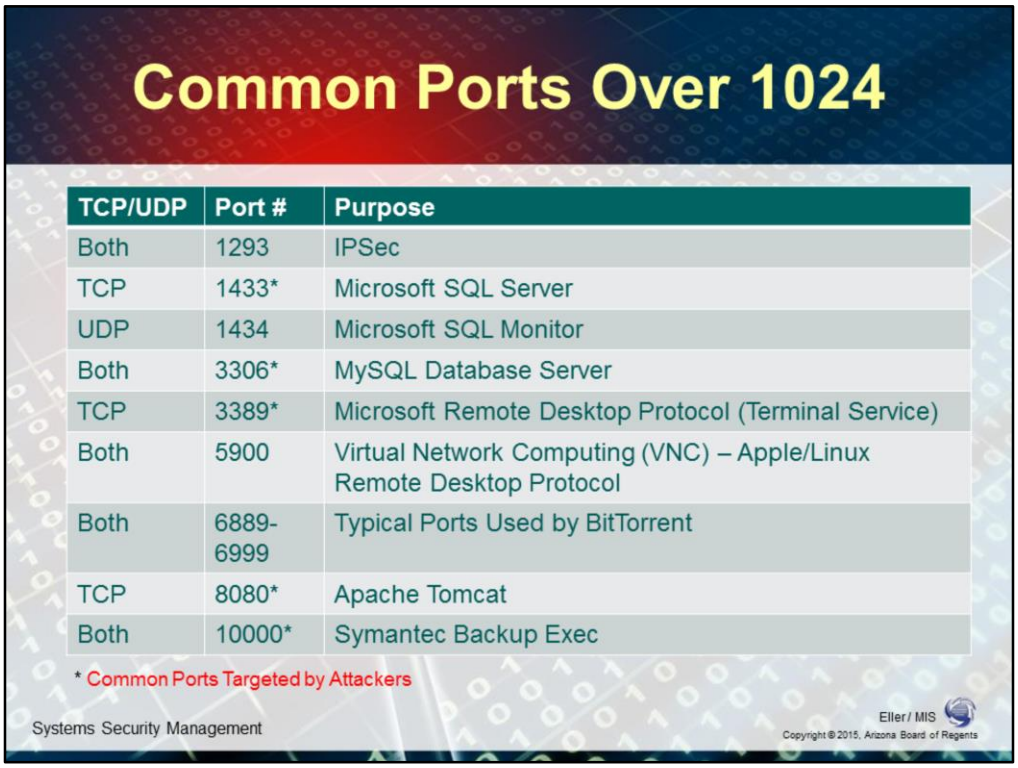
Out of these ports, the most common ports for attackers to target include TCP/UDP ports 119, 135, 137, and 138.



Common TCP/IP Ports (continued)

TCP/UDP	Port #	Purpose
Both	139*	NetBIOS Applications
Both	161*	Simple Network Management Protocol (SNMP)
Both	201-208	AppleTalk Applications
Both	213	IPX
Both	218	Message Posting Protocol (MPP)
Both	389*	Lightweight Directory Access Protocol (LDAP)
Both	443	HTTP & HTTPS Over SSL and TLS
Both	445*	NetBIOS over TCP/IP (UDP for SMB File Sharing)
Both	749	Kerberos Administration
TCP	993	Internet Message Access Protocol Over TLS/SSL
TCP	995	Post Office Protocol (POP) version 3 Over TLS/SSL

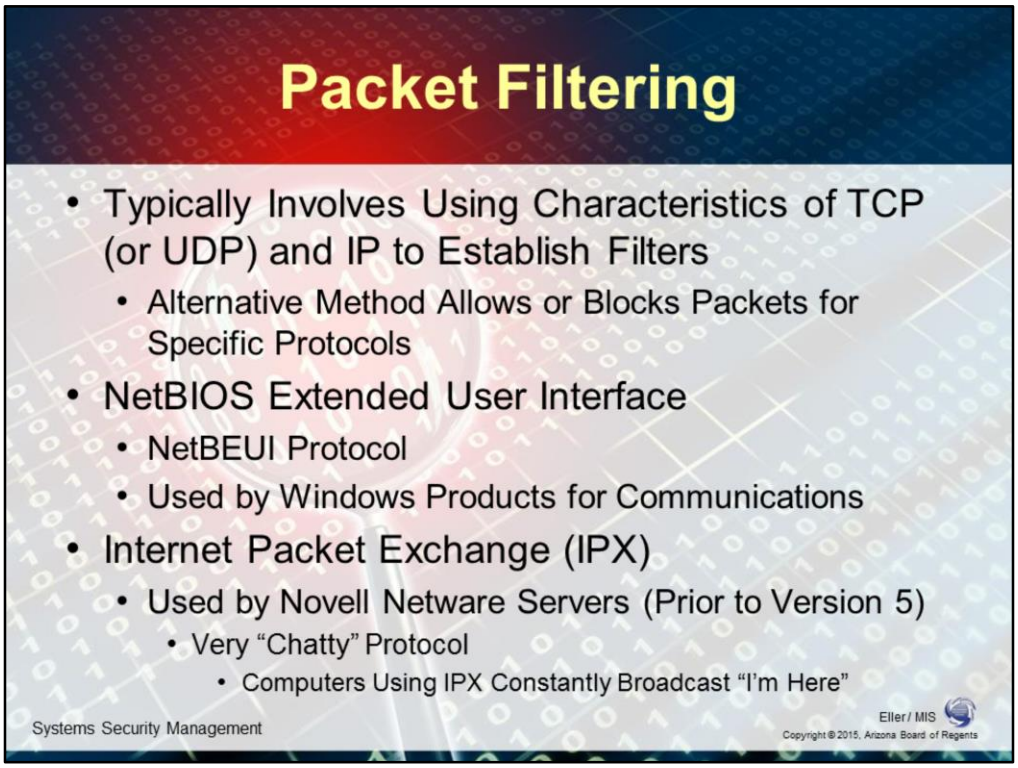
Out of these ports, the most common ports for attackers to target include TCP/UDP ports 139, 161, 389, and 445.



Common TCP/IP Ports over 1024

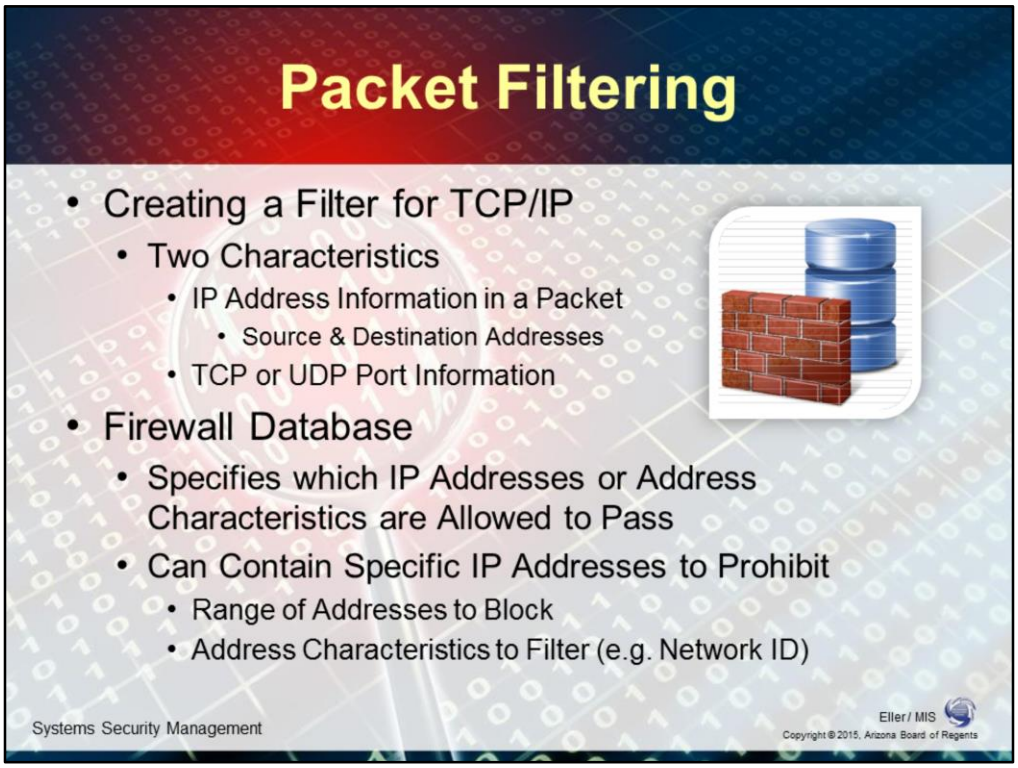
TCP/UDP	Port #	Purpose
Both	1293	IPSec
TCP	1433*	Microsoft SQL Server
UDP	1434	Microsoft SQL Monitor
Both	3306*	MySQL Database Server
TCP	3389*	Microsoft Remote Desktop Protocol (Terminal Service)
Both	5900	Virtual Network Computing (VNC) – Apple/Linux Remote Desktop Protocol
Both	6889-6999	Typical Ports Used by BitTorrent
TCP	8080*	Apache Tomcat
Both	10000*	Symantec Backup Exec

Out of these ports, the most common ports for attackers to target include TCP Ports 1433, 3389, and 8080 and TCP/UDP ports 3306, and 10000.



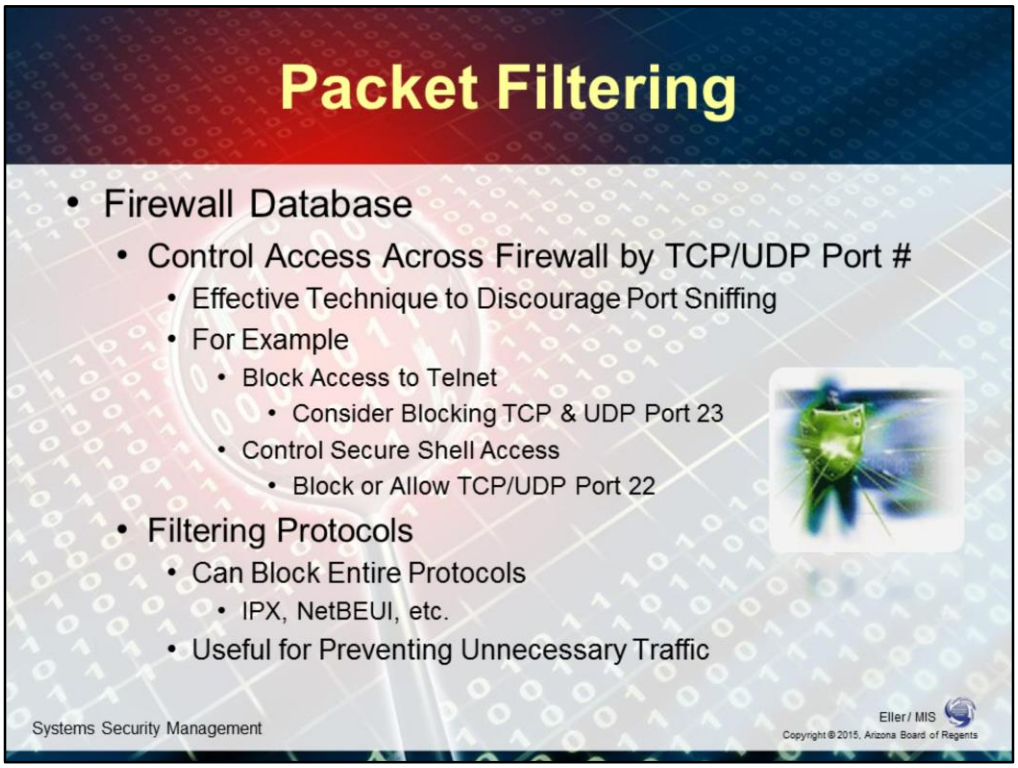
Packet Filtering

Packet filtering typically involves using the characteristics of TCP or UDP, and IP in order to establish filters designed to examine packets for specific information. An alternative method for packet filtering involves allowing or blocking packets for specific protocols. For example, Microsoft developed the NetBIOS Extended User Interface, or NetBEUI, protocol, designed for use by Windows systems for network communications. Novell Netware servers, prior to version 5, used a proprietary protocol called the Internet Packet Exchange, or IPX. This latter protocol was considered to be a very "chatty" protocol because computers using IPX would constantly broadcast "I'm here" to the entire network. Packet filtering could be used to block either of these protocols, among others.



Packet Filtering (continued)

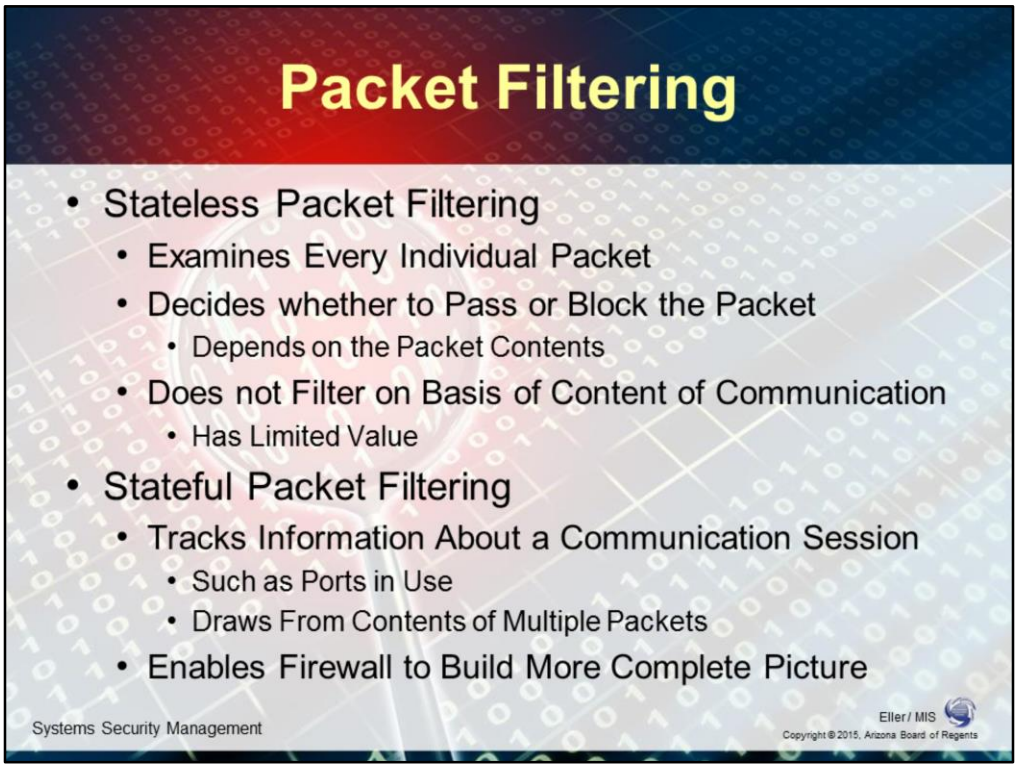
When creating a filter for TCP/IP, there are two characteristics which are needed: the IP address information in the packet (source and destination) and the TCP or UDP port information. Packet filters will normally reside in a firewall database, router, or intrusion detection system, and they will specify which IP addresses, or address characteristics, are allowed to pass. Filters can contain specific IP addresses to prohibit, such as a range of addresses to block, or the address characteristics that need to be filtered, such as Network ID.



Packet Filtering (continued)

The database can also control access across a firewall via TCP or UDP port number. This is an effective technique used to discourage port sniffing by attackers. For example, if you want to block access to the Telnet protocol, you should consider blocking TCP and UDP port 23. If you want to control secure shell access, you can block or allow TCP and UDP port 22.

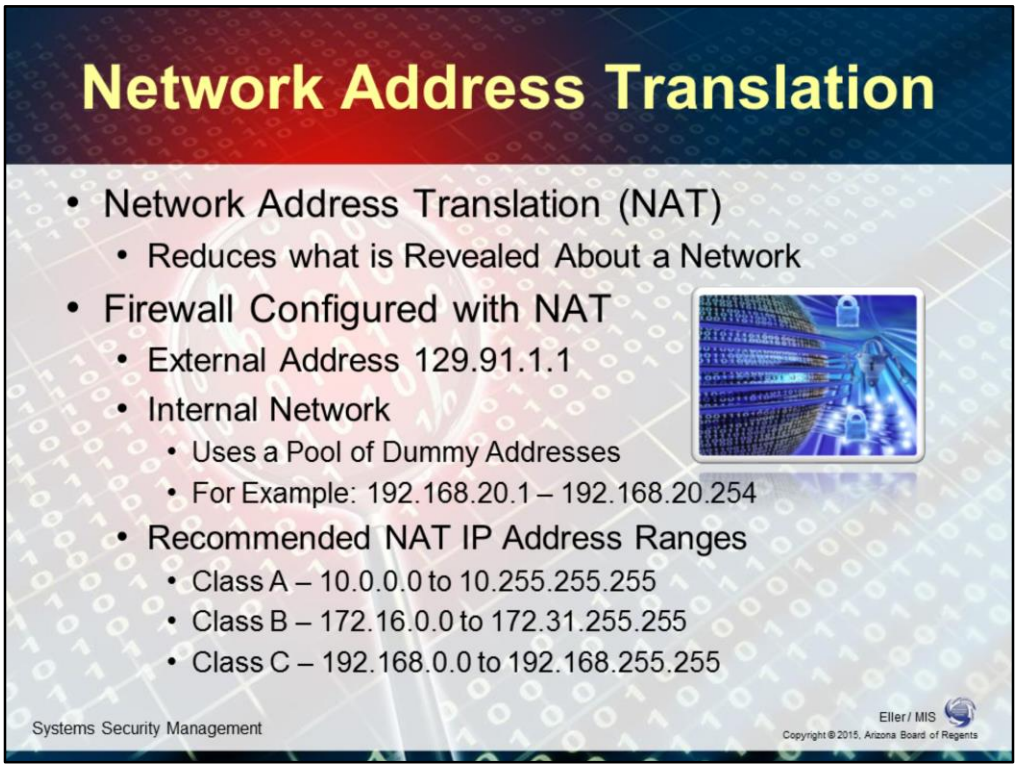
With protocol filtering you can block entire protocols, such as IPX and NetBEUI. This can be useful for preventing unnecessary traffic on the network.



Packet Filtering (continued)

Another method is called Stateless Packet Filtering. Stateless filtering examines every individual packet and decides whether to pass the packet on or block the packet depending on the packet's contents. Stateless filtering does not filter on the basis of the content of the overall communication, so it has limited value.

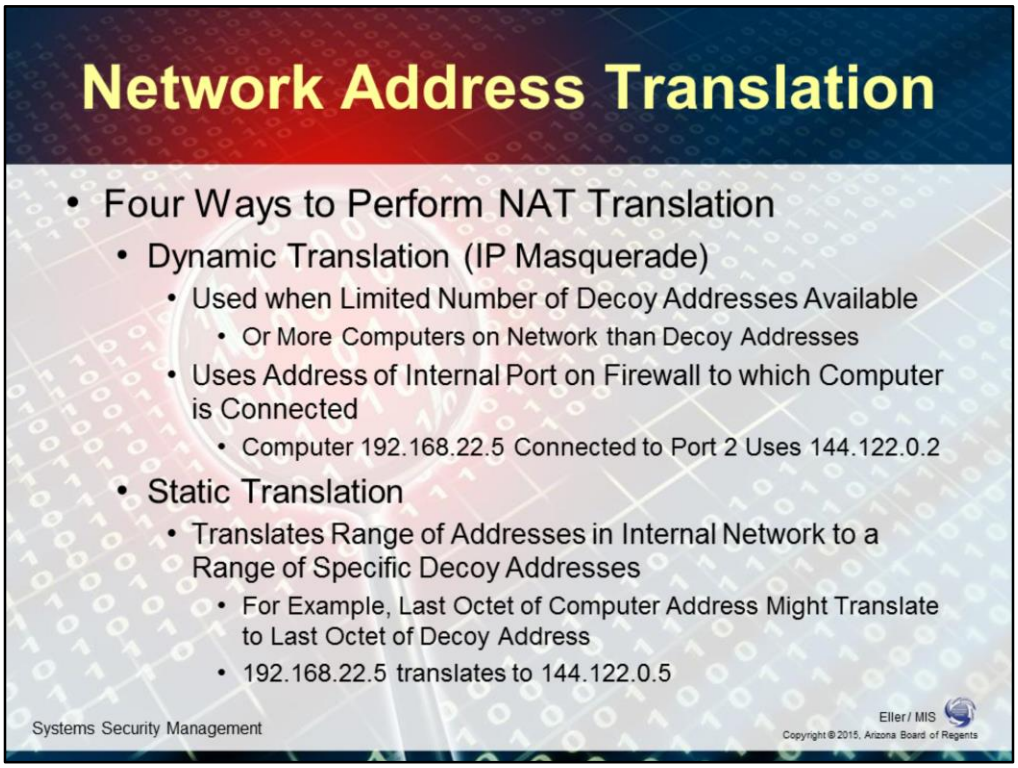
With Stateful Packet Filtering, the firewall will track information about an entire communications session, such as the ports in use, drawing information from the contents of multiple packets. This type of filtering enables the firewall to build a more complete picture of the network communications.



Network Address Translation

Network Address Translation, or NAT, is a security method designed to reduce what can be revealed about a network. Typically an organization will deploy a firewall on the network that can be configured with NAT. As far as the external network is concerned, the firewall will be configured with a specific IP address, such as 129.91.1.1. The internal network behind the firewall will be configured to use a pool of dummy IP addresses, for example, devices behind the firewall might use IPs in the range of 192.168.20.1 through 192.168.20.254.

Depending on the size of the internal network, there are three ranges of IP addresses which can be used. If a Class A network is needed, the SysAdmin can use addresses in the range 10.0.0.0 through 10.255.255.255. If a smaller, Class B network is needed internally, a SysAdmin can use addresses in the range of 172.16.0.0 through 172.31.255.255. Finally, if the internal network only needs to be a Class C network, the SysAdmin can use addresses in the range of 192.168.0.0 through 192.168.255.255. Any of these dummy networks can be subdivided if desired, just like a regular IP network.



Network Address Translation (continued)

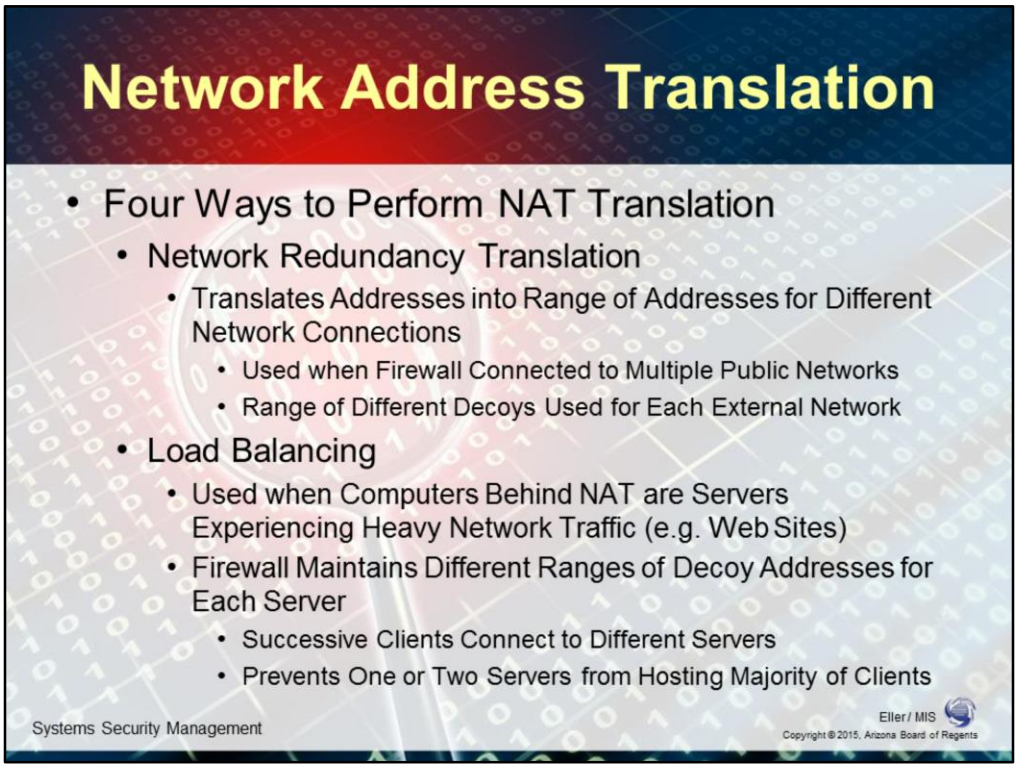
There are four ways to perform NAT translation on a network: dynamic translation, static translation, network redundancy translation, and load balancing.

Dynamic Translation

Dynamic translation, or IP masquerade, is used when there are a limited number of decoy addresses available, or there are more computers on a network than decoy addresses are available. This type of NAT uses the address of the internal port on the firewall to which the computer is connected. For example, a computer with IP address 192.168.22.5 is connected to port 2 on the firewall, using IP 144.122.0.2.

Static Translation

Static translation is designed to translate a range of addresses in the internal network to a range of specific decoy addresses. For example, the last octet of the computer address might translate to the last octet of the decoy address, so 192.168.22.5 would translate to 144.122.0.5.



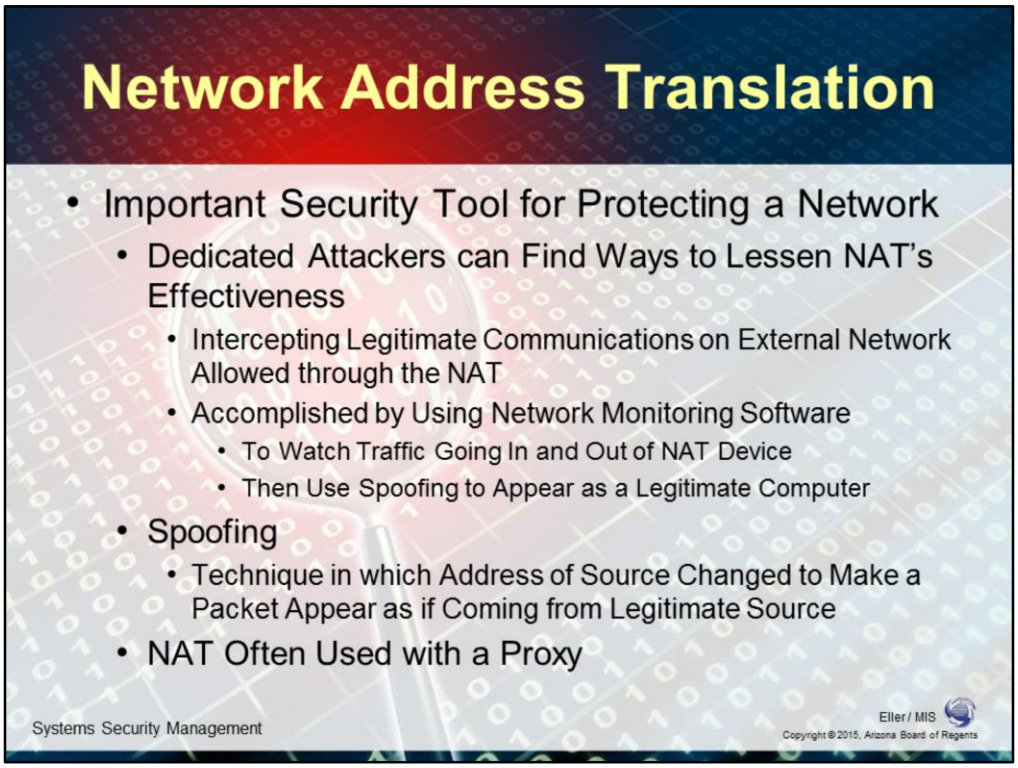
Network Address Translation (continued)

Network Redundancy Translation

Network redundancy translation is designed to translate addresses into a range of addresses for different network connections. Typically this is used when a firewall is connected to multiple public networks or a range of different decoys is used for each external network.

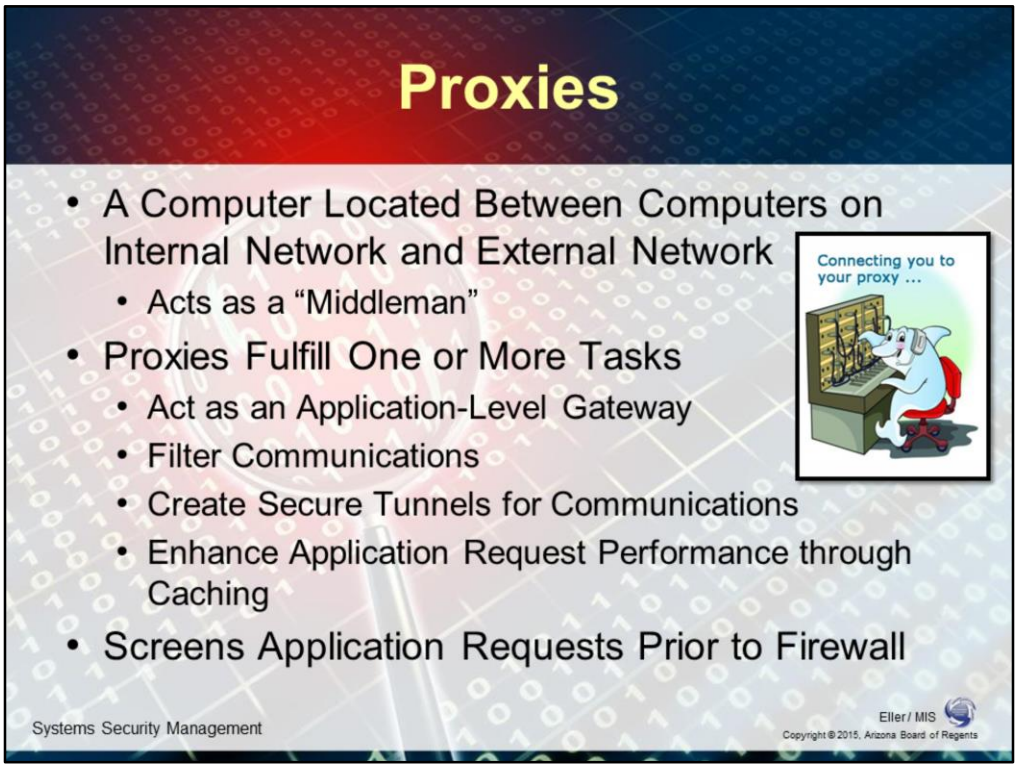
Load Balancing

Load balancing is used when computers behind a NAT are servers which are experiencing a heavy network load, like a web site. The firewall will maintain different ranges of decoy addresses for each server, so successive clients will connect to different servers in order to prevent one of two servers from hosting a majority of the clients.



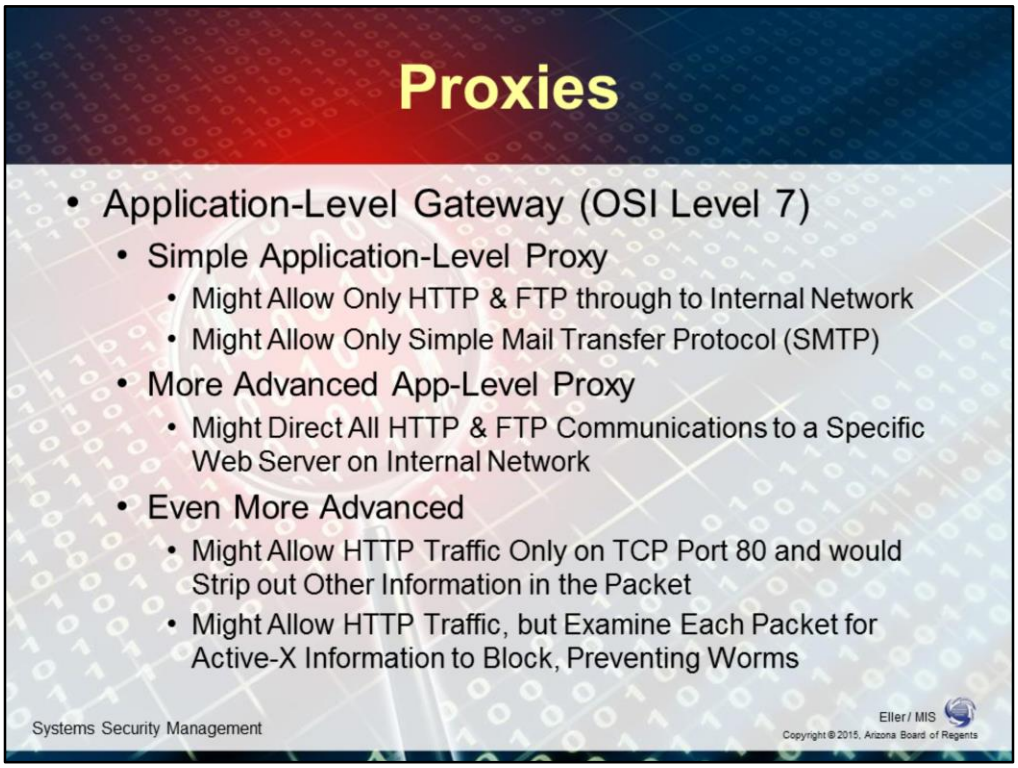
Network Address Translation (continued)

NAT is a very important security tool for protecting a network; however, dedicated attackers can still find ways to lessen the effectiveness of a NAT. For example, an attacker can intercept legitimate communications on an external network that is allowed through the NAT. This can also be accomplished by using network monitoring software to watch traffic going in and out of a NAT device, then using spoofing to appear as a legitimate computer. Spoofing is a technique in which the address of the source is changed to make a packet appear as if it is coming from a legitimate source. Oftentimes, NAT will be used in conjunction with a Proxy server for more effective security.



Proxies

A proxy is a computer located between computers on an internal network and external network which acts as a middleman in all network communications. Proxies are designed to fulfill one or more specific tasks, such as acting as an application-level gateway (from the OSI model), to filter communications, to create secure tunnels for communication, or enhancing application request performance through caching. The proxy will screen application requests prior to sending data through a firewall.

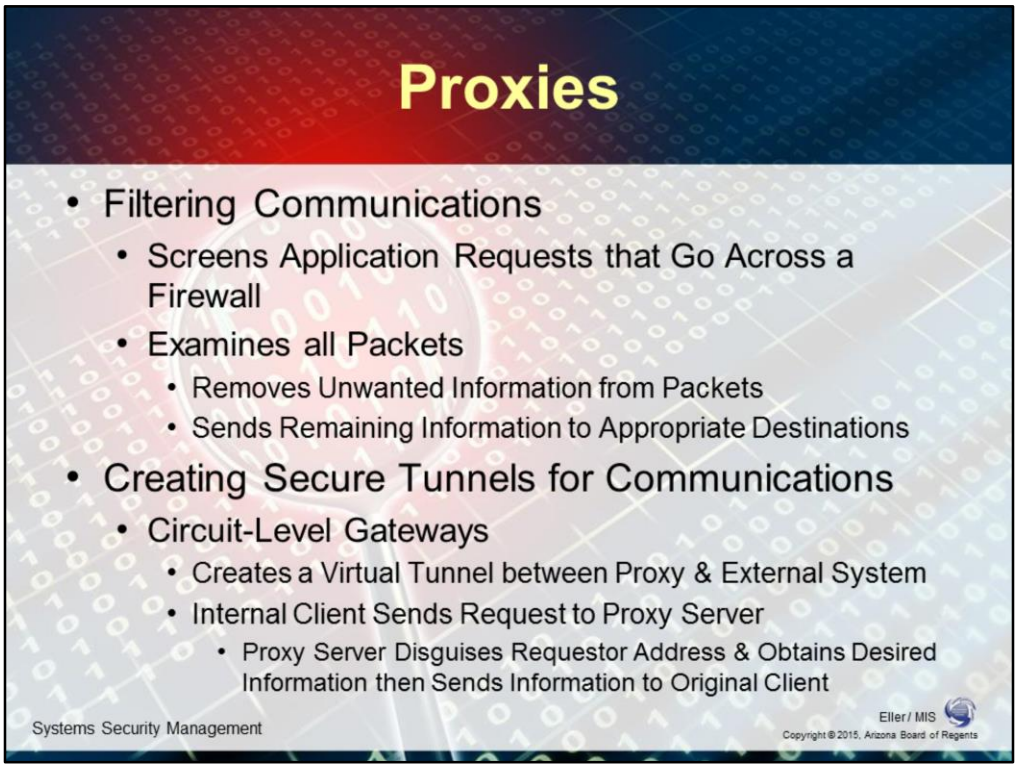


Proxies (continued)

When acting as an Application-Level Gateway, or OSI Model Level 7, a proxy can be configured in several different ways. When configured as a simple application-level proxy, it might allow only HTTP and FTP traffic through to the internal network, or it might only allow the simple mail transfer protocol, or SMTP, through. In other words, it will only allow specific traffic through to the internal network.

With a more advanced application-level proxy, it might actually direct all HTTP or FTP communications to a specific web server on the internal network. This is a more directed means of sending traffic where it needs to go rather than allowing it into the entire network.

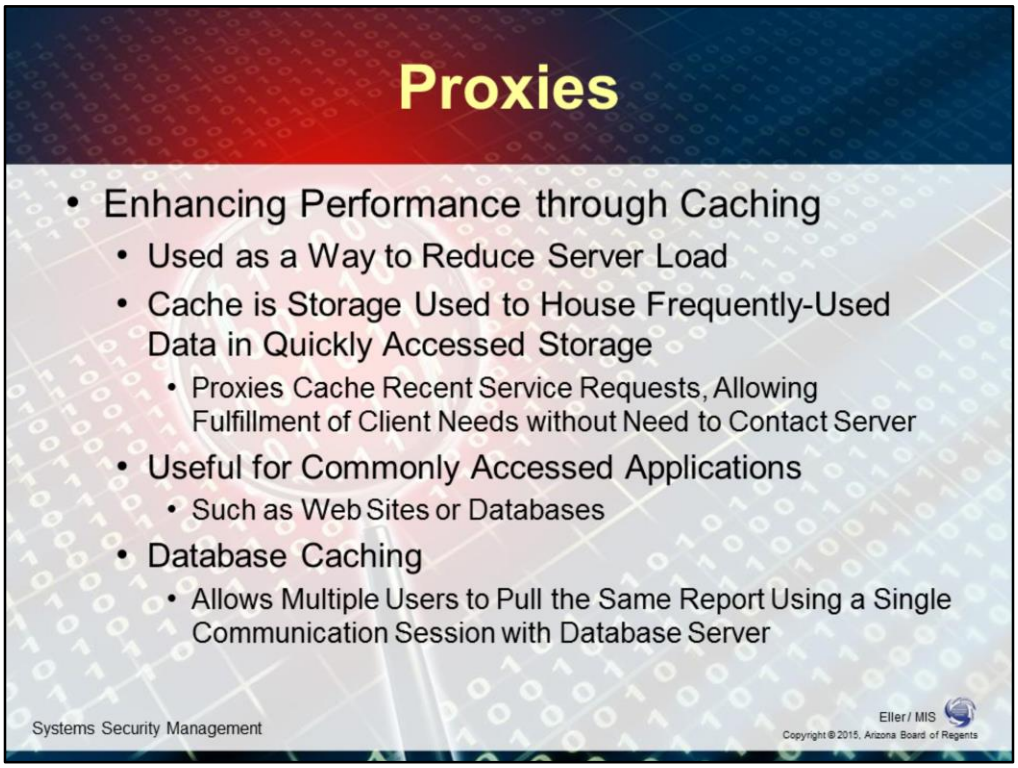
An even more advanced proxy might only allow HTTP traffic that is sent over TCP port 80, and would remove all other information contained within the packet. Alternatively, it might allow all HTTP traffic, but examine each packet for Active-X information in order to block or prevent the spread of worms.



Proxies (continued)

When a proxy is configured to filter communications, this server will screen application requests that are sent across a firewall. All packets are examined in order to remove unwanted information from the packets while sending the remaining information to the appropriate destinations.

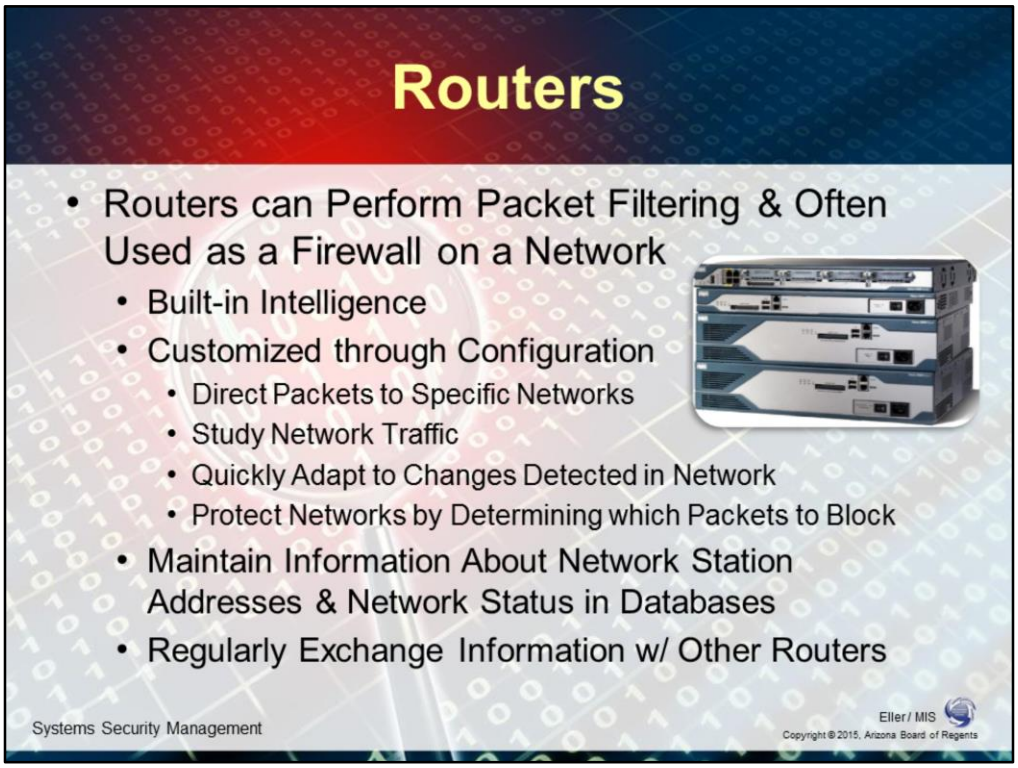
When proxies are used to create secure tunnels for communications, this is referred to as a circuit-level gateway. This type of gateway creates a virtual tunnel between the proxy and the external system. The internal client sends a request to the proxy server and the proxy disguises the requestor address, obtains the desired information, then sends the information to the original client who requested it.



Proxies (continued)

Proxies are also used to help enhance network performance through caching. This technique is used as a means of reducing server load. The cache is storage used to house frequently-used data in quickly accessed storage. The proxy will cache recent service requests, allowing the fulfillment of client needs without the need to contact the target server again. This is very useful for commonly accessed applications such as web sites or databases. With regard to database caching, this allows multiple users to pull the same report(s) using a single communication session with the database server.

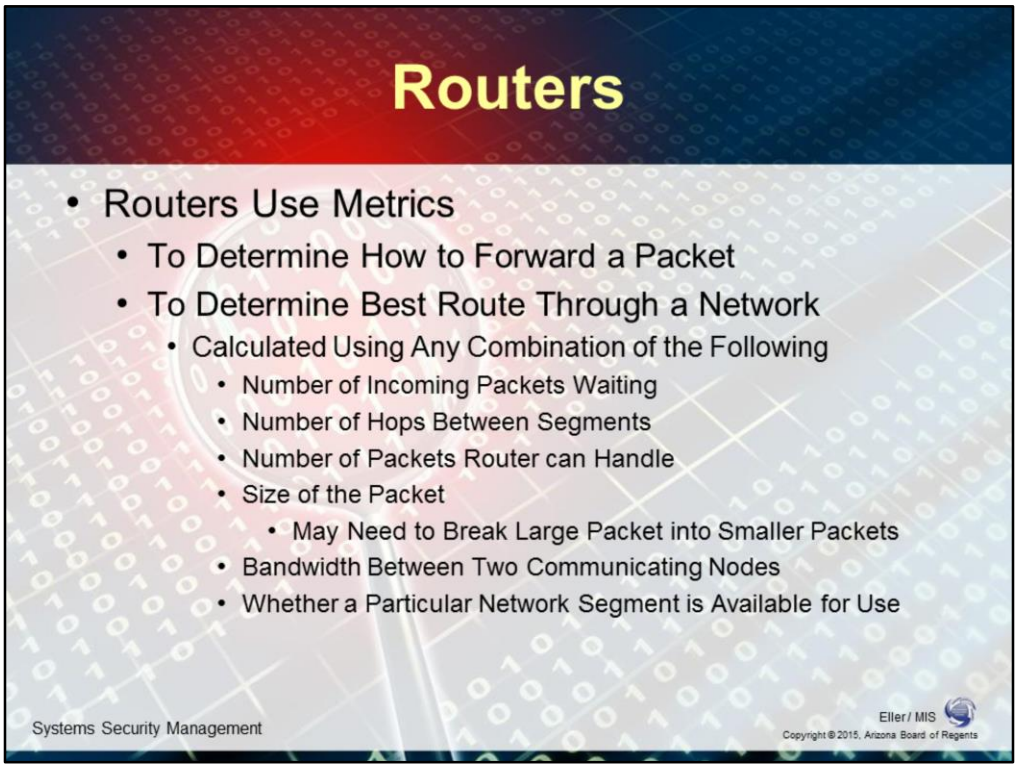
A caveat to using proxies for caching is it is possible for the proxy to cache bad information, such as a site under maintenance. The maintenance pages will remain cached on the proxy for however long the proxy is configured to keep the information. If this information is kept too long, this will hinder user's ability to effectively use and access network resources. There needs to be a balance between how long this information is cached and when the proxy sends a new request to the server for a copy of the data.



Routers

Routers are used on a network to send data packets from one network to another as the packets are sent to their ultimate destination. In addition to this, routers can also perform packet filtering and are often used as a firewall on a network. Routers are designed with built-in intelligence software which can be customized through configuration. Configuration settings can be used to direct packets to specific networks (such as VLANs), to study network traffic, to quickly adapt to changes detected within the network, or to protect networks by determining which packets to block. Routers maintain information about network station addresses, both IPs and MAC Addresses, and the network status in databases. These databases are then used to regularly exchange information with other routers on nearby networks.

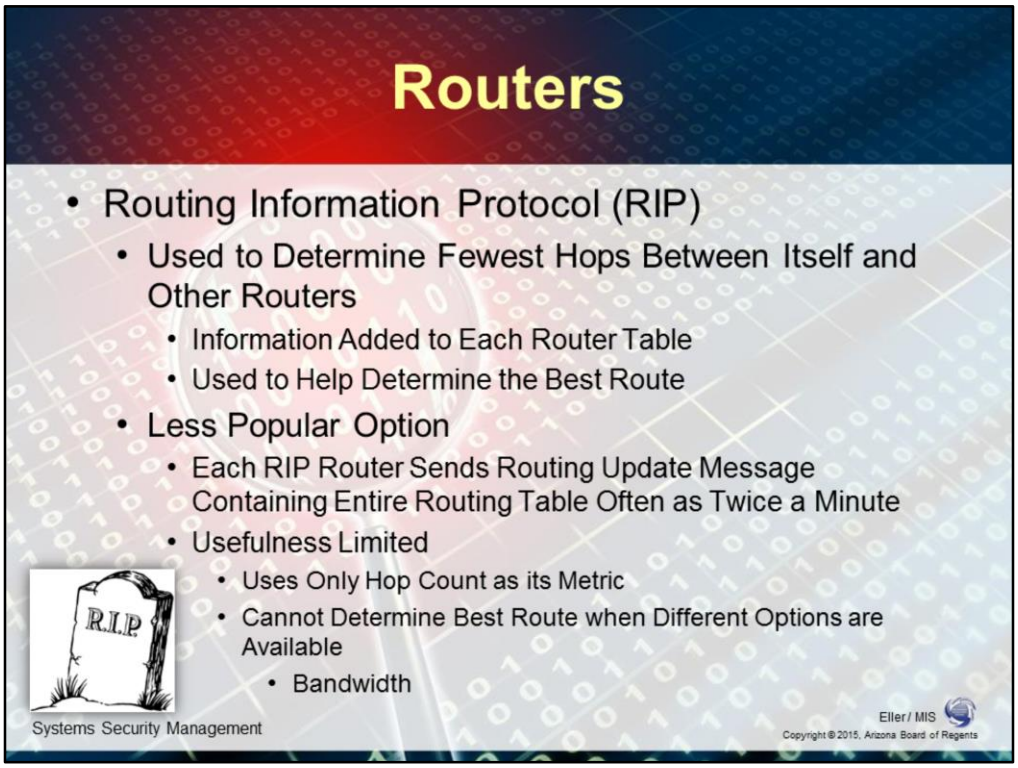
Routers function at Layer 3 of the OSI Model: the Network layer. This is what gives them the ability to do packet filtering, unlike switches that operate at Layer 2, where IP information is not used, only MAC addresses. (NOTE – There are certain “Layer 3 switches” that offer similar functionality, but those are beyond the scope of this course. Most switches just operate at Layer 2.)



Routers (continued)

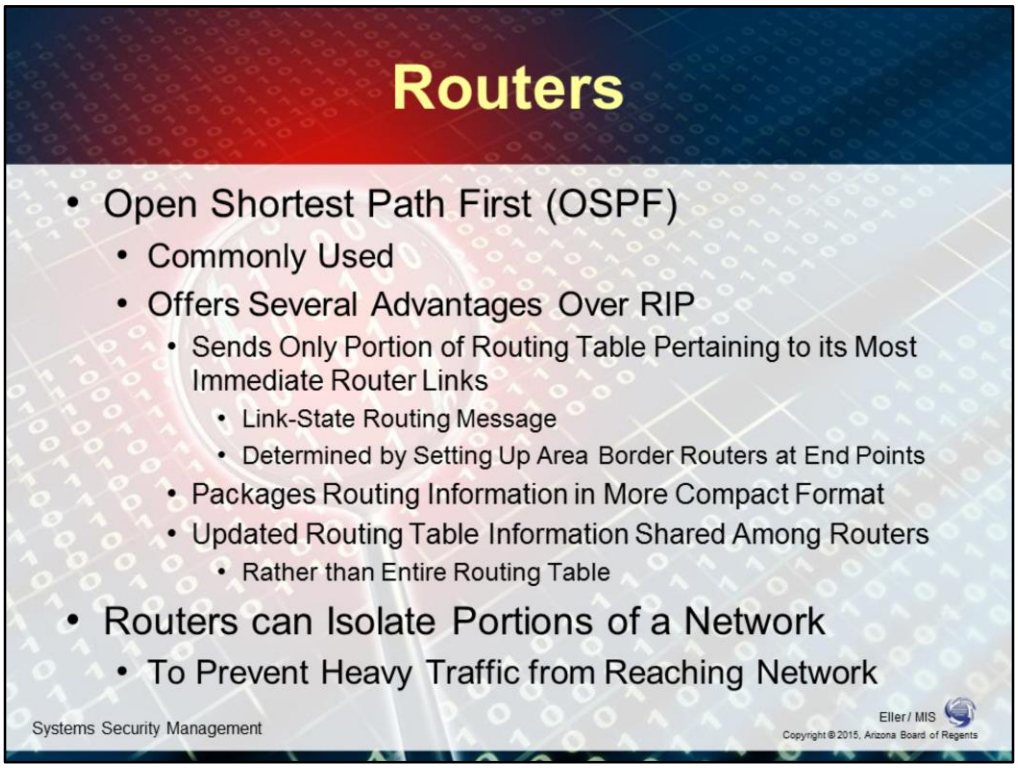
In order to determine how to forward packets or what the best route for forwarding packets will be, a router will collect and analyze metrics. These metrics can be calculated using any combination of the following:

- Number of Incoming Packets Waiting
- Number of Hops Between Segments
- Number of Packets Router can Handle
- Size of the Packet (May Need to Break Large Packet into Smaller Packets)
- Bandwidth Between Two Communicating Nodes
- Whether a Particular Network Segment is Available for Use



Routers (continued)

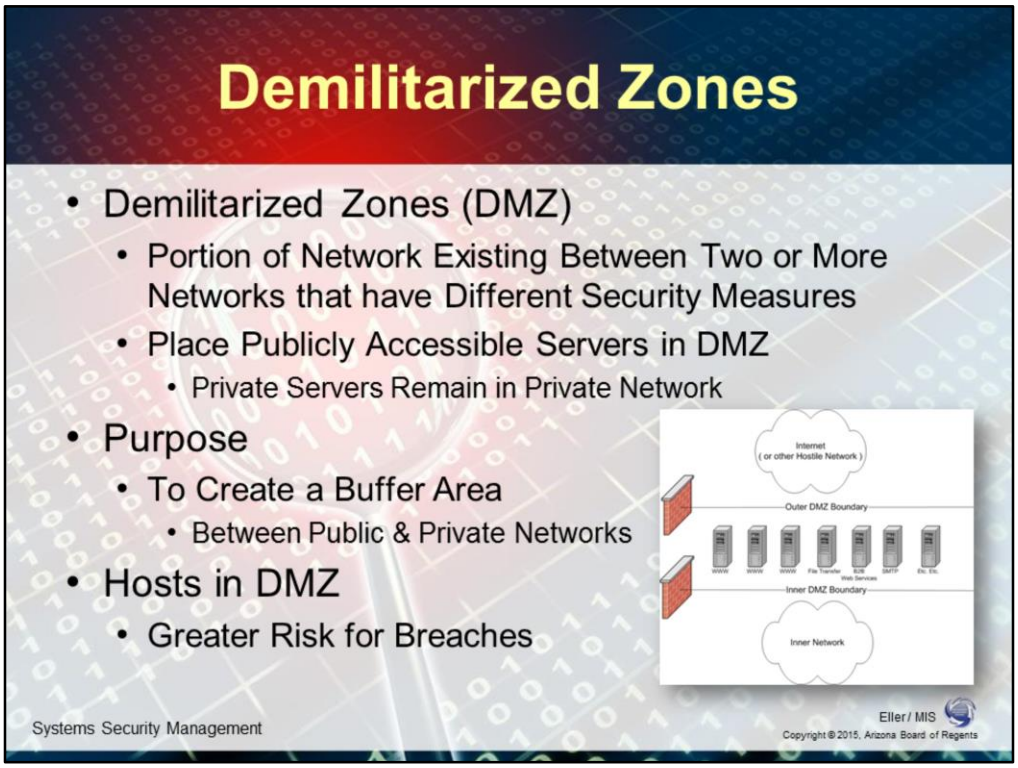
The Routing Information Protocol, or RIP, is used to determine the fewest number of hops between a router on the internal network and other routers nearby. This information is added to each router table and is used to determine the best route for sending data packets. This protocol is a less popular option for gathering metrics because each RIP router sends a routing update message containing the entire routing table as often as twice a minute. This makes its usefulness limited, as the hop count is the only metric RIP uses. RIP also cannot determine the best route when different options are available, such as bandwidth.



Routers (continued)

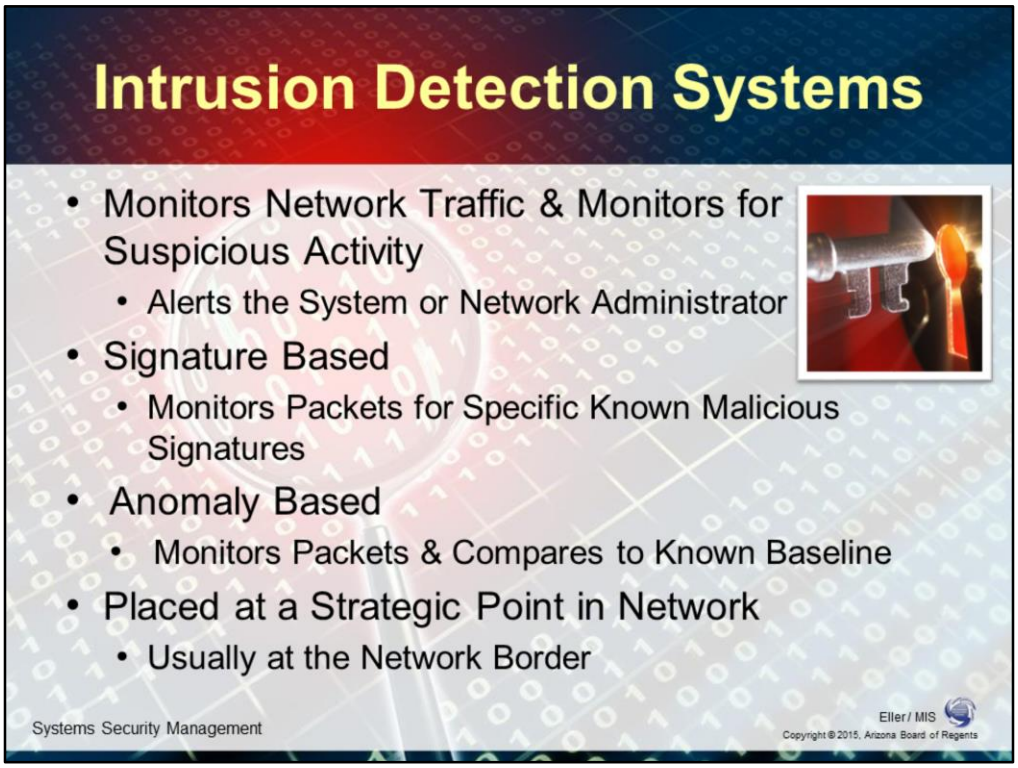
Open Shortest Path First, or OSPF, is the most commonly used means for a router to gather metrics. OSPF offers several advantages over RIP. For example, OSPF sends only a portion of the routing table that pertains to its most immediate router links instead of sending the entire routing table. It does this through link-state routing messages that are determined by setting up area border routers at network end points. OSPF also packages the routing information in a more compact format, allowing the data to be sent to other routers quickly and efficiently. OSPF also sends only updated routing table information to be shared among other routers, rather than the entire routing table.

Finally, routers can also be configured to isolate portions of a network as a means of preventing heavy traffic from reaching the network intact.



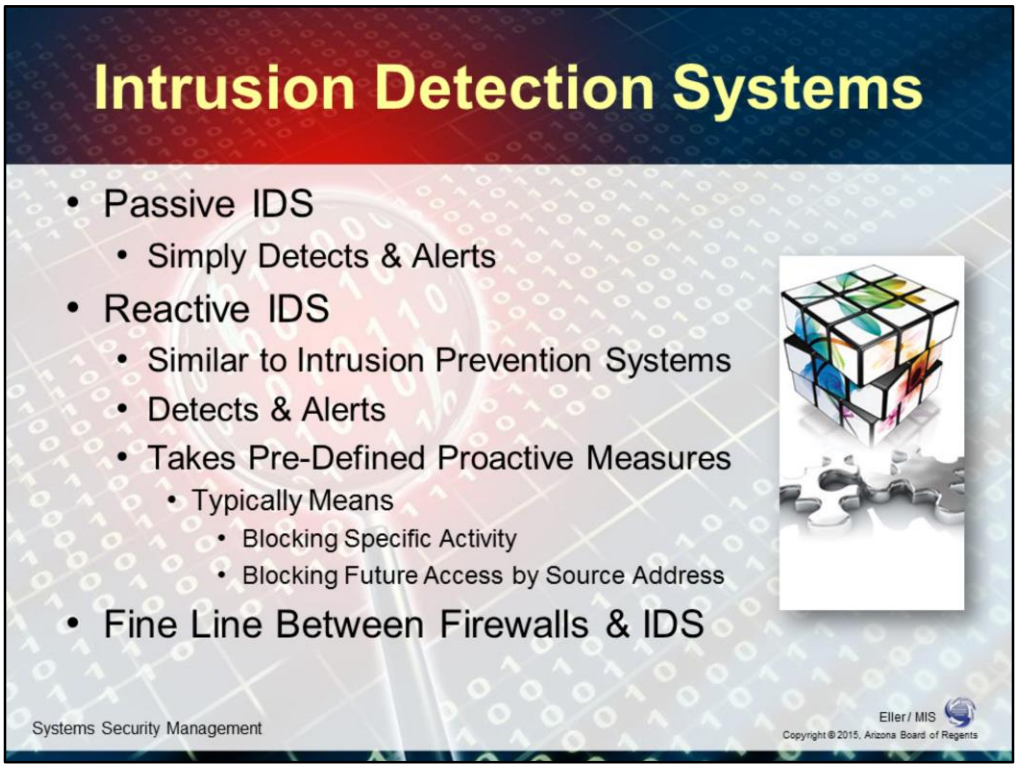
Demilitarized Zones

A demilitarized zone, or DMZ, is a term describing a portion of a computer network which exists between two or more networks that have different security measures. For example, an organization may choose to place publically accessible web servers and other resources available to the public within a DMZ, allowing public access to these resources, while also preventing the public from accessing resources within the company's internal network. The purpose of a DMZ is to create a buffer area that lies between public and private networks in order to protect internal resources; however, any systems hosted within the DMZ are at a much greater risk for system breach than those in the private network.



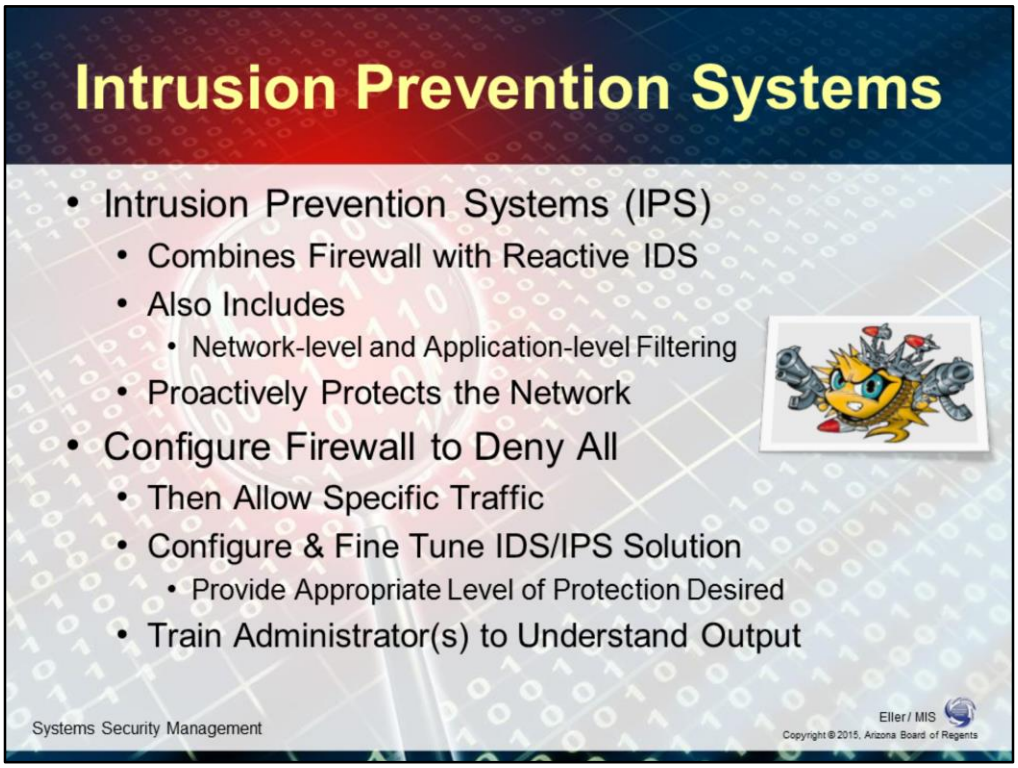
Intrusion Detection Systems

An Intrusion Detection System, or IDS, is designed to monitor network traffic in an effort to detect suspicious activity. Once suspicious activity is detected, the system is designed to notify the SysAdmin for further analysis. A signature-based IDS monitors packets for specific, known malicious signatures, such as data patterns exhibited by worms or spyware. An anomaly-based IDS monitors packets and simply compares them to a known, good baseline. Usually IDS devices are placed at a strategic point on the network, such as the network border, so all packets coming in and out of the network are monitored.



Intrusion Detection Systems (continued)

There are two primary types of IDS devices: passive and reactive. Passive IDS devices simply detect and alert. Reactive IDS devices are similar to Intrusion Prevention Systems. As with passive systems, reactive devices will detect and alert; however, they will also take pre-defined, proactive security measures in an effort to block specific activity or block future access to the network by the source address. There is a fine line between a firewall and an IDS.



Intrusion Prevention Systems

An Intrusion Prevention System, or IPS, combines a firewall with a reactive IDS and includes both network-level and application-level filtering of packets in order to proactively protect the network. As with a normal firewall, the IPS firewall should be configured to deny all traffic, then only allow specific traffic into the network. The SysAdmin can then configure and fine-tune the IDS/IPS solution to provide the desired level of protection. With this type of system, it is important to train the administrators to understand the system's output and react appropriately.

In the next module, we will discuss wireless networking, including:

- Wireless Networking Basics
- Attacks on Wireless Networks
- Why Wireless?
- Radio Wave Technologies
- IEEE 802.11 Radio Wave Networking
- Bluetooth Radio Wave Networking
- Anatomy of Attacks on Wireless Networking
- Wireless Security Measures
- Wireless Policy

References

Bradley, T. (2010). Introduction to Intrusion Detection Systems. *About.com: Internet/Network Security*. Retrieved from http://netsecurity.about.com/cs/hackertools/a/aa030504.htm.

List of TCP and UDP Port Numbers. (2009). Wikipedia, the Free Encyclopedia. Retrieved from http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers.

Palmer, M. (2004). Guide to Operating System Security, 1st Edition. *Thomson Course Technology*. Canada.