Network Security Cheat sheet

Network Threat Facts

The following table summarizes some common threats to your network:

Threat	Description
Denial of Service (DoS)	A Denial of Service (DoS) attack impacts system availability by flooding the target system with traffic or requests or by exploiting a system or software flaw. The goal of a DoS attack is to make a service or device unavailable to respond to legitimate requests. There are several forms of DoS attacks: • In a regular DoS attack, a single attacker sends packets directly to a single target. • A Distributed Denial of Service (DDoS) attack uses zombies (slave computers) to multiply the number of attacks directed at the target. A DDoS attack allows the attacker to hide his identity. • A Distributed Reflective Denial of Service (DRDoS) uses an amplification network to increase the severity of the attack. Packets are sent to the amplification network addressed as coming from the target. The amplification network responds back to the target system.
Smurf	A *Smurf* attack is a form of DRDoS attack that spoofs the source address in ICMP packets. A Smurf attack requires an attacker system, an amplification network, and a victim computer or network. The attacker: • Sends ICMP packets to an amplification network or broadcast address. The packets spoof the source address to be that of the target. • The amplification network responds sending packets to the target (victim) site. • The victim has thousands of replies to packets sent by the attacker. Many personal firewalls block all ICMP protocol messages in response to these attacks.
Virus	A *virus* is a program that attempts to damage a computer system and replicate itself to other computer systems. A virus: • Requires a host to replicate and usually attaches itself to a host file or a hard drive sector. • Usually attaches to files with execution capabilities such as .doc, .exe, and .bat extensions. • Replicates each time the host is used. • Often focuses on destruction or corruption of data. • Often distributes via e-mail. Many viruses can e-mail themselves to everyone in your address book. • Examples: Stoned, Michelangelo, Melissa, I Love You.
Worm	A *worm* is a self-replicating program that uses the network to replicate itself to other systems. A worm does not require a host system to replicate. A worm can negatively impact network traffic just in the process of replicating itself. A worm: • Is usually introduced into the system through a vulnerability in a system or a network.

	• Infects one system and spreads to other systems on the network. • Can be designed to delete files or send e-mails. • Can install a backdoor in the infected computer, allowing an attacker to remotely access the system. • Example: Code Red.
Man-in-the-middle	A *man-in-the-middle* attack is used to intercept information passing between two communication partners. Man-in-the-middle attacks are commonly used to steal credit cards, online bank credentials, as well as confidential personal and business information. With a man-in-the-middle attack: • An attacker inserts himself in the communication flow between the client and server. The client is fooled into authenticating to the attacker. • Both parties at the endpoints believe they are communicating directly with the other, while the attacker intercepts and/or modifies the data in transit. The attacker can then authenticate to the server using the intercepted credentials.
Rogue access point	A *rogue access point* is an unauthorized access point added to a network or an access point that is configured to mimic a valid access point. Examples include: • An attacker or an employee with access to the wired network installs a wireless access point on a free port. The access port then provides a method for remotely accessing the network. • An attacker near a valid wireless access point installs an access point with the same (or similar) SSID. The access point is configured to prompt for credentials, allowing the attacker to steal those credentials or use them in a man-in-the-middle attack to connect to the valid wireless access point. • An attacker configures a wireless access point in a public location, then monitors traffic of those who connect to the access point. Monitoring of radio frequencies in your area is one way to prevent rogue access points.
Social engineering	Social engineering exploits human nature by convincing someone to reveal information or perform an activity. Social engineering relies on the attacker gaining or exploiting the trust of and individual. Examples of social engineering include: • *Dumpster diving* is the process of looking in the trash for sensitive information that has not been properly disposed of. • *Shoulder surfing* involves looking over the shoulder of someone while they work. • *Piggybacking* refers to an attacker entering a secured building by following an authorized employee. • *Eavesdropping* refers to an unauthorized person listening to conversations of employees or other authorized personnel discussing sensitive topics. • *Masquerading* refers to convincing personnel to grant access to sensitive information or protected systems by pretending to be someone who is authorized and/or requires that access. o The attacker usually poses as a member of senior management or technical support. o A scenario of distress is fabricated to the user to convince them that their actions are necessary. • *Phishing* uses an e-mail and a spoofed Web site to gain sensitive information. In a phishing attack: o A fraudulent message that appears to be legitimate is sent to a target.

	○ The message requests the target to visit a Web site which also appears to be legitimate. ○ The fraudulent Web site requests the victim to provide sensitive information such as the account number and password. • Hoax viruses are spoofed e-mails that ask for information or ask for tasks to be performed (such as delete a file or go to a Web site and enter sensitive information). Hoax virus e-mails are a form of a phishing attack.

Countermeasures Facts

A *countermeasure* is an action or a control put into place that eliminates or reduces the effects of a threat or attack. The following list identifies common countermeasures that protect against a wide variety of attacks.

- Identify potential attacks and put policies and procedures into place designed to reduce the threat from the attack.
  - A *policy* is a general statement about the role of security in the organization. A policy describes what is and is not allowed.
  - A *procedure* is a detailed, specific step-by-step instruction for a process. For example, a procedure might detail what to do when an unknown person requests access, or the steps to take in response to an identified threat.
- Implement user training and awareness programs to communicate policies and procedures, and to educate users about potential threats and valid responses to those threats. End users often represent the biggest weakness to the security of your organization. Training seeks to strengthen users so they respond appropriately.
- Apply patches and updates to systems, particularly updates related to security issues. Updates often eliminate the vulnerability associated with a specific attack.
- Implement strong physical security to control access to your facilities and your network. Physical security controls include:
  - Keeping doors and windows locked.
  - Requiring identification or key cards before entry is permitted.
  - Escorting visitors at all times.
  - Keeping devices with sensitive information out of view of public users.
  - Keeping the server room locked. Locking computers to racks or tables to prevent theft.

The following table lists specific countermeasures for various types of attacks.

Attack Types	Countermeasures
Automated attacks	Specific countermeasures to prevent automated attacks (such as Denial of Service attacks) include: • Implement firewall filters to block traffic type associated with known attacks. • Update software on network devices to eliminate known vulnerabilities. • Block ICMP messages. • Communicate with your Internet service provider when an attack starts so they

	can implement controls to block offending traffic.
Malware	*Malware* includes various software, such as viruses and worms, that can harm your system. Specific countermeasures to protect against malware include: • Deploy anti-virus software. Be sure to update the virus definition files regularly. • Educate users. • Block attachments at network borders, in particular those containing executable code (.exe, .bat, .doc files with macros). • Prevent the download of software from the Internet. • Enforce strict software installation policies. • Remove removable drives (floppy and CD-ROM drives) to prevent unauthorized software entering a system. • Scan new files before they are added to your system. Implement software that scans e-mail attachments before delivery.
Man-in-the-middle attacks	Countermeasures for man-in-the-middle attacks are: • Use encrypted communication protocols, such as IPSec. • Use certificates. • Perform mutual authentication.
Social engineering	The most effective countermeasure for social engineering is employee awareness training on how to recognize social engineering schemes and how to respond appropriately. Specific countermeasures include: • Train employees to demand proof of identity over the phone and in person. • Define values for types of information, such as dial-in numbers, user names, passwords, network addresses, etc. The greater the value, the higher the security around those items should be maintained. • If someone requests privileged information, have employees find out why they want it and whether they are authorized to obtain it. • Verify information contained in e-mails and use bookmarked links instead of links in e-mails to go to company Web sites. • Dispose of sensitive documents securely, such as shredding or incinerating. • Dispose of disks and devices securely by shredding floppy disks or overwriting disks with all 1's, all 0's, then all random characters. • Verify information from suspicious e-mails by visiting two or more well-known malicious code threat management Web sites. These sites can be your antivirus vendor or a well-known and well-regarded Internet security watch group.

Firewall Facts

A *firewall* is a device or software running on a device that inspects network traffic and allows or blocks traffic based on a set of rules.

- A *network-based* firewall inspects traffic as it flows between networks. For example, you can install a network-based firewall on the edge of your private network that connects to the Internet to protect against attacks from Internet hosts.
- A *host-based* firewall inspects traffic received by a host. Use a host-based firewall to protect against attacks when there is no network-based firewall, such as when you connect to the Internet from a public location.
- Firewalls use filtering *rules*, sometimes called *access control lists* (ACLs), to identify allowed and blocked traffic. A rule identifies characteristics of the traffic, such as:
    - The interface the rule applies to
    - The direction of traffic (inbound or outbound)
    - Packet information such as the source or destination IP address or port number
    - The action to take when the traffic matches the filter criteria
- Firewalls do not offer protection against all attacks (such as spoofed e-mail messages).

The following table explains different firewall types.

Firewall Type	**Characteristics**
Packet filtering firewall	A *packet filtering firewall* makes decisions about which network traffic to allow by examining information in the IP packet header such as source and destination addresses, ports, and service protocols. A packet filtering firewall: • Uses access control lists (ACLs) or filter rules to control traffic. • Operates at OSI layer 3 (Network layer). • Offers high performance because it only examines addressing information in the packet header. • Can be implemented using features that are included in most routers. • Is a popular solution because it is easy to implement and maintain, has a minimal impact on system performance, and is fairly inexpensive. A packet filtering firewall is considered a *stateless* firewall because it examines each packet and uses rules to accept or reject each packet without considering whether the packet is part of a valid and active session.
Circuit-level proxy	A *circuit-level proxy* or *gateway* makes decisions about which traffic to allow based on virtual circuits or sessions. A circuit-level gateway: • Operates at OSI Layer 5 (Session layer). • Keeps a table of known connections and sessions. Packets directed to known sessions are accepted. • Verifies that packets are properly sequenced. • Ensures that the TCP three-way handshake process occurs only when appropriate. • Does not filter packets. Rather it allows or denies sessions. A circuit-level proxy is considered a *stateful* firewall because it keeps track of the state of a session. A circuit-level proxy can filter traffic that uses dynamic ports because the firewall matches the session information, and not the port numbers, for filtering. In general, circuit-level proxies are slower than packet filtering firewalls, although if only the session state is being used for filtering, a circuit-level gateway can be faster after the initial session information has been identified.
Application level gateway	An application level gateway is a firewall that is capable of filtering based on information contained within the data portion of a packet. An application level gateway:

	• Examines the entire content (not just individual packets). • Operates at OSI Layer 7 (Application layer). • Understands or interfaces with the application-layer protocol. • Can filter based on user, group, and data such as URLs within an HTTP request. • Is the slowest form of firewall because entire messages are reassembled at the Application layer. One example of an application level gateway is a *proxy* server. A proxy server is a device that stands as an intermediary between a secure private network and the public. Proxies can be configured to: • Control both inbound and outbound traffic. • Increase performance by caching heavily accessed content. Content is retrieved from the proxy cache instead of being retrieved from the original server. • Filter content. • Shield or hide a private network. • Restrict access by user or by specific Web sites.

A common method of using firewalls is to identify various network *zones*. Each zone identifies a collection of users who have similar access needs. Firewalls are configured at the edge of these zones to filter incoming and outbound traffic. For example, you can define a zone that includes all hosts on your private network protected from the Internet. Or you can define a zone within your network for controlled access to specific servers that hold sensitive information.

A *demilitarized zone* (DMZ), also called a *screened subnet*, is a buffer network (or subnet) that sits between the private network and an untrusted network (such as the Internet).

- The DMZ is created using the following configurations:
    - Configure two firewall devices: one connected to the public network and one connected to the private network.
    - Configure a single device with three network cards: one connected to the public network, one connected to the private network, and one connected to the screened subnet.
    - Configure a single device with two network cards: one connected to the public network, and another connected to a private subnet containing hosts that are accessible from the private network. Configure proxy ARP so the public interface of the firewall device responds to ARP requests for the public IP address of the device.
- Publicly-accessible resources (servers) are placed inside the screened subnet. Examples of publicly-accessible resources include Web, FTP, or e-mail servers.
- Packet filters on the outer firewall allow traffic directed to the public resources inside the DMZ. Packet filters on the inner firewall prevent unauthorized traffic from reaching the private network.
- If the firewall managing traffic into the DMZ fails, only the servers in the DMZ are subject to compromise. The LAN is protected by default.
- When designing the outer firewall packet filters, a common practice is to close all ports, opening only those ports necessary for accessing the public resources inside the DMZ.

Common Ports

Network ports are logical connections, provided by the TCP or UDP protocols at the Transport layer, for use by protocols in the upper layers of the OSI model. The TCP/IP protocol stack uses port numbers to determine what protocol incoming traffic should be directed to. Some characteristics of ports are listed below:

- Ports allow a single host with a single IP address to run network services. Each port number identifies a distinct service.
- Each host can have over 65,000 ports per IP address.
- Port use is regulated by the Internet Corporation for Assigning Names and Numbers (ICANN).

ICANN specifies three categories for ports.

- *Well known* ports range from 0 to 1023 and are assigned to common protocols and services.
- *Registered* ports range from 1024 to 49151 and are assigned by ICANN to a specific service.
- *Dynamic* (also called *private* or *high*) ports range from 49,152 to 65,535 and can be used by any service on an ad hoc basis. Ports are assigned when a session is established, and released when the session ends.

The following table lists the well known ports that correspond to common Internet services.

Port(s)	Service
20 TCP 21 TCP	File Transfer Protocol (FTP)
22 TCP and UDP	Secure Shell (SSH)
23 TCP	Telnet
25 TCP	Simple Mail Transfer Protocol (SMTP)
53 TCP and UDP	Domain Name Server (DNS)
67 UDP 68 UDP	Dynamic Host Configuration Protocol (DHCP)
69 UDP	Trivial File Transfer Protocol (TFTP)
80 TCP	HyperText Transfer Protocol (HTTP)
110 TCP	Post Office Protocol (POP3)
119 TCP	Network News Transport Protocol (NNTP)
123 UDP	Network Time Protocol (NTP)
143 TCP and UDP	Internet Message Access Protocol (IMAP4)
161 TCP and UDP 162 TCP and UDP	Simple Network Management Protocol (SNMP)
389 TCP and UDP	Lightweight Directory Access Protocol
443 TCP and UDP	HTTP with Secure Sockets Layer (SSL)

Note: To protect a server, ensure that only the necessary ports are opened. For example, if the server is only being used for e-mail, then shut down ports that correspond to FTP, DNS, and HTTP (among others).

VPN Facts

A virtual private network (VPN) is a network that uses encryption to allow IP traffic to travel securely over the TCP/IP network. A VPN is used primarily to support secured communications over an untrusted network.

- VPNs work by using a *tunneling* protocol that encrypts packet contents and wraps them in an unencrypted packet.
- Tunnel endpoints are devices that can encrypt and decrypt packets. When you create a VPN, you establish a security association between the two tunnel endpoints. These endpoints create a secure, virtual communication channel. Only the destination tunnel endpoint can unwrap packets and decrypt the packet contents.
- Routers use the unencrypted packet headers to deliver the packet to the destination device. Intermediate routers along the path cannot (and do not) read the encrypted packet contents.
- A VPN can be used over a local area network, across a WAN connection, over the Internet, and even between a client and a server over a dial-up connection through the Internet.
- VPNs can be implemented in the following ways:
    - With a *host-to-host* VPN, two hosts establish a secure channel and communicate directly. With this configuration, both devices must be capable of creating the VPN connection.
    - With a *site-to-site* VPN, routers on the edge of each site establish a VPN with the router at the other location. Data from hosts within the site are encrypted before being sent to the other site. With this configuration, individual hosts are unaware of the VPN.
    - With a *remote access* VPN, a server on the edge of a network (called a VPN *concentrator*) is configured to accept VPN connections from individual hosts. Hosts that are allowed to connect using the VPN connection are granted access to resources on the VPN server or the private network.

The following table compares the common VPN tunneling protocols.

Protocol	**Description**
Point-to-Point Tunneling Protocol (PPTP)	PPTP was one of the first VPN protocols. Developed by Microsoft, PPTP: • Uses standard authentication protocols, such as Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP). • Supports TCP/IP only. • Encapsulates other LAN protocols and carries the data securely over an IP network. • Uses Microsoft's MPPE for data encryption. • Is supported by most operating systems and servers. • Uses TCP port 1723.
Layer Two Tunneling Protocol (L2TP)	L2TP is an open standard for secure multi-protocol routing. L2TP: • Supports multiple protocols (not just IP). • Uses IPSec for encryption. • Is not supported by older operating systems. • Uses TCP port 1701 and UDP port 500.
Internet Protocol Security (IPSec)	IPSec provides authentication and encryption, and can be used in conjunction with L2TP or by itself as a VPN solution. IPSec includes two protocols that provide different features. • Authentication Header (AH) provides authentication features. Use AH to

	enable authentication with IPSec. • Encapsulating Security Payload (ESP) provides data encryption. Use ESP to encrypt data. **Note:** If you use only AH, data is *not* encrypted. IPSec can be used to secure the following types of communications: • Host-to-host communications within a LAN. • VPN communications through the Internet, either by itself or in conjunction with the L2TP VPN protocol. • Any traffic supported by the IP protocol including Web, e-mail, Telnet, file transfer, and SNMP traffic as well as countless others. IPSec uses either digital certificates or pre-shared keys.
Secure Sockets Layer (SSL)	The SSL protocol has long been used to secure traffic generated by other IP protocols such as HTTP, FTP, and e-mail. SSL can also be used as a VPN solution, typically in a remote access scenario. SSL: • Authenticates the server to the client using public key cryptography and digital certificates. • Encrypts the entire communication session. • Uses port 443, a port that is often already opened in most firewalls. Implementations that use SSL for VPN tunneling include Microsoft's SSTP and Cisco's SSL VPN.

You should be aware that ports must be opened in firewalls to allow VPN protocols. For this reason, using SSL for the VPN often works through firewalls when other solutions do not. In addition, some NAT solutions do not work well with VPN connections.

Switch Security Facts

The following table lists switch features that can be implemented to increase network security:

Feature	**Description**
Virtual LAN (VLAN)	A virtual LAN (VLAN) is a logical grouping of computers based on switch port. • VLAN membership is configured by assigning a switch port to a VLAN. • A switch can have multiple VLANs configured on it, but each switch port can only be a member of a single VLAN (with one exception described below). • VLANs can be defined on a single switch, or configured on multiple interconnected switches. With multiple switches, each switch can be configured with the same VLANs, and devices on one switch can communicate with devices on other switches as long as they are on the same VLAN. • A *trunk* port is used to connect two switches together. o Typically, Gigabit Ethernet ports are used for trunk ports, although any port can be a trunking port.

	o A trunk port is a member of all VLANs, and carries traffic between the switches. o When trunking is used, frames that are sent over a trunk port are tagged by the first switch with the VLAN ID so that the receiving switch knows to which VLAN the frame belongs. o The *trunking protocol* describes the format that switches use for tagging frames with the VLAN ID. o Because end devices do not understand the VLAN tags, the tag is removed from the frame by the switch before the frame is forwarded to the destination device. o VLAN tagging is only used for frames that travel between switches on the trunk ports. • In a typical configuration with multiple VLANs and a single or multiple switches, workstations in one VLAN will not be able to communicate with workstations in other VLANs. To enable inter-VLAN communication, you will need to use a router (or a Layer 3 switch). • Using VLANs, the switch can be used to create multiple IP broadcast domains. Each VLAN is in its own broadcast domain, with broadcast traffic being sent only to members of the same VLAN.
MAC filtering/port security	With switch port security, the devices that can connect to a switch through the port are restricted. • Port security uses the MAC address to identify allowed and denied devices. • You can specify only a single MAC address that is allowed, or allow multiple addresses per port. • With automatic configuration, the next device to connect to the port is allowed, while additional devices are denied. • On the switch, MAC addresses are stored in RAM in a table, and are associated with the port. • A *port violation* occurs when an unauthorized device tries to connect. When a violation occurs, you can drop all frames from the unauthorized device or shut down the port, disabling all communications through that port.
Port authentication (802.1x)	Port authentication is provided by the 802.1x protocol, and allows only authenticated devices to connect to the LAN through the switch. Authentication uses usernames and passwords, smart cards, or other authentication methods. • When a device first connects, the port is set to an unauthorized state. Ports in unauthorized states can only be used for 802.1x authentication traffic. • The process begins by the switch sending an authentication request to the device. • The device responds with authentication credentials, which are forwarded by the switch to the authentication device (such as a RADIUS server). • After the server authenticates the device or the user, the switch port is placed in an authorized state, and access to other LAN devices is allowed. • When a device disconnects, the switch places the port in the unauthorized state.

Be aware of the following when implementing switch security:

- Creating VLANs with switches offers the following administrative benefits.
    - You can create virtual LANs based on criteria other than physical location (such as workgroup, protocol, or service)
    - You can simplify device moves (devices are moved to new VLANs by modifying the port assignment)
    - You can control broadcast traffic based on logical criteria (only devices in the same VLAN receive broadcast traffic)
    - You can control security (isolate traffic within a VLAN)
- When you use switches to create VLANs, you will still need routers to:
    - Route data in to and out of the local area network
    - Route data between VLANs
    - Apply firewall filtering rules to traffic
- VLANs are commonly used with Voice over IP (VoIP) to distinguish voice traffic from data traffic. Traffic on the voice VLAN can be given a higher priority to ensure timely delivery.
- MAC filtering uses the MAC address of a device to drop or forward frames through the switch. Port authentication requires that the user or device authenticates before frames are forwarded through the switch.

Authentication Facts

Authentication is the process of submitting and checking credentials (such as username and password) to validate or prove user identity. The authentication protocol identifies how credentials are submitted, protected during transmission, and validated.

Instead of a simple username and password, some authentication protocols use *certificates* and *digital signatures* for proof of identity.

- A *certificate* is a digital document that identifies a user or a computer. The certificate includes a *subject* name that is the name of a user or a computer.
- Certificates are obtained from a Public Key Infrastructure (PKI). A PKI is a system that provides for a trusted third party to vouch for user identities.
- A PKI is made up of Certification Authorities (CAs), also called *certificate authorities*. A CA is an entity trusted to issue, store, and revoke certificates. A CA:
    - Accepts certificate requests.
    - Verifies the information provided by the requester.
    - Creates and issues the certificate to the requester.
    - Revokes certificates (revoked certificates are not valid).
    - Publishes a list of revoked certificates known as the *certificate revocation list* (CRL).
- You can obtain certificates from a public CA (such as VeriSign), or install your own PKI and CAs to issue certificates to users and computers in your organization.
- Computers accept as valid any certificate issued by a trusted CA. By default, most computers trust well-known public CAs. If you configure your own PKI, you also configure each computer in your organization to trust your own CAs. **Note**: If you want a certificate to be trusted by users outside of your organization, obtain a certificate from a third-party CA.
- A *digital signature* is a digital document that is altered in such a way so that it can only have come from the subject identified in the certificate. A certificate obtained from a PKI is signed by the CA that issued the certificate (the digital signature of the issuing CA is included within the certificate).
- A computer that receives a certificate verifies the issuing CA's signature, and accepts the identity of the user or computer if the CA is trusted.

The following table describes various authentication protocols and their use.

Protocol	Description

Challenge Handshake Authentication Protocol (CHAP)	CHAP is a three-way handshake (challenge/response) authentication protocol used for remote access connections. Both devices are configured with a password called a *shared secret*. For unique user authentication, this value is associated with a user account. Authentication using a challenge/response mechanism occurs as follows: 1. The server generates a message called a *challenge* message and sends this to the client. 2. The client uses the shared secret to hash the challenge message and returns this value along with the username to the server (the *response*). 3. The server uses its copy of the shared secret for that user to perform the same hash. The server compares its hashed value with the hashed value received from the client. With CHAP, plaintext versions of the password are never sent, only the hashed challenge message is sent between devices.
Microsoft Challenge Handshake Authentication Protocol (MS-CHAP)	MS-CHAP is Microsoft's proprietary, challenge-response authentication method used for remote access connections. • MS-CHAP encrypts the shared secret on each system so that it is not saved in plain text. • MS-CHAP provides a mechanism for changing the password over the remote connection. • MS-CHAP v2 allows for *mutual authentication*, where the server authenticates to the client. MS-CHAP is considered more secure than CHAP, and should be used if supported on both devices.
Extensible Authentication Protocol (EAP)	EAP allows the client and server to negotiate the characteristics of authentication. • An EAP authentication scheme is called an EAP *type*. Both the client and authenticator have to support the same EAP type for authentication to function. • When a connection is established, the client and server negotiate the authentication type that will be used, based on the allowed or required authentication types configured on each device. • The submission of authentication credentials occurs based on the rules defined by the authentication type. • EAP is used to allow authentication using smart cards, biometrics (user physical characteristics), and certificate-based authentication.
Kerberos	Kerberos is used for both authentication and authorization to services. Kerberos grants *tickets* (also called a security *token*) to authenticated users and to authorized resources. The process of using tickets to validate permissions is called *delegated authentication*. Kerberos uses the following components: • An authentication server (AS) accepts and processes authentication requests. • A service server (SS) is a server that provides or holds network resources. • A ticket granting server (TGS) grants tickets that are valid for specific

	resources on specific servers. Keberos works as follows: 1. The client sends an authentication request to the authentication server. 2. The authentication server validates the user identity and grants a ticket granting ticket (TGT). The TGT validates the user identity and is good for a specific ticket granting server. 3. When the client needs to access a resource, it submits its TGT to the TGS. The TGS validates that the user is allowed access, and issues a client-to-server ticket. 4. The client connects to the service server and submits the client-to-server ticket as proof of access. 5. The SS accepts the ticket and allows access. Tickets are valid during the entire session and do not need to be re-requested. Windows Active Directory uses Kerberos for user authentication and for controlling resource access. Kerberos requires that all servers within the process have synchronized clocks to validate tickets.
802.1x	802.1x authentication is an authentication method used on a LAN to allow or deny access based on a port or connection to the network. • 802.1x is used for port authentication on switches and authentication to wireless access points. • 802.1x requires an authentication server for validating user credentials. This server is typically a RADIUS server. • Authentication credentials are passed from the client, through the access point device, on to the authentication server. • The access point enables or disables traffic on the port based on the authentication status of the user. • Authenticated users are allowed full access to the network; unauthenticated users only have access to the RADIUS server. • 802.1x is based on EAP and can use a wide variety of methods for authentication (such as usernames and passwords, certificates, or smart cards).

Secure Protocol Facts

When many protocols were created, they were designed with little or no security controls. An unsecured protocol is one that does not provide authentication or encryption, or that uses plaintext for passing authentication protocols or data. Security services (authentication and encryption) are often added to new or existing protocols using one of the following secure protocols:

- Secure Sockets Layer (SSL)
- Transport Layer Security (TLS)
- Secure Shell (SSH)

The following table compares unsecure and secure protocols:

Unsecure Protocol	Secure Protocol	Description
Hypertext Transfer Protocol (HTTP)	HTTP over SSL (HTTPS)	HTTPS is a secure form of HTTP that uses SSL to encrypt data before it is transmitted.
Telnet Remote SHell (RSH)	Secure SHell (SSH)	SSH allows for secure interactive control of remote systems. SSH uses RSA public key cryptography for both connection and authentication. SSH is also a protocol that can be used to provide security services for other protocols.
File Transfer Protocol (FTP) Remote Copy Protocol (RCP)	Secure FTP (SFTP) FTP over SSL (FTPS) Secure Copy Protocol (SCP)	Both SFTP and SCP are secure file copy protocols that use SSH for security. SSH provides authentication and encryption. FTPS uses SSL to encrypt data.
Simple Network Management Protocol (SNMPv1/2)	SNMPv3	The original version of SNMP has several vulnerabilities including: • No authentication of devices. Any device configured with the correct community name can send messages. Any message sent from a device with the correct community name is received and processed. • Information sent in plain text. • The SNMP manager is able to send messages to devices that result in the device taking an action. This feature is often disabled because of the security risk it poses. SNMP v2 added some security features, but most security comes with SNMP v3. SNMP v3 adds the following: • Authentication for agents and managers • Encryption of SNMP information • Message integrity to ensure that data is not altered in transit

Detection and Prevention Facts

An intrusion detection system (IDS) is a special network device that can detect attacks and suspicious activity. There are several ways to describe typical detection systems:

Method	Variations
Response capability	An intrusion detection system can be classified by how it responds when a threat is detected: • A *passive* IDS monitors, logs, and detects security breaches but takes no action to stop or prevent the attack. A passive IDS can send an alert, but it is the network administrator's job to interpret the degree of the threat and to respond accordingly.

	• An *active* IDS (also called an *intrusion protection system* or IPS) performs the functions of an IDS, but can also *react* when security breaches occur. An IPS: o Can automate responses that may include dynamic policy adjustment and reconfiguration of supporting network devices to block the offending traffic. o Performs behaviors that can be seen by anyone watching the network. Usually these actions are necessary to block malicious activities or discover the identity of an intruder. Updating filters and performing reverse lookups are common behaviors of an active IDS.
Recognition method	The recognition method defines how the system distinguishes attacks and threats from normal activity. • *Signature* recognition, also referred to as *pattern* matching or *dictionary* recognition, looks for patterns in network traffic and compares it to known attack patterns called *signatures*. o IDS signatures are written and updated by the IDS vendor in response to identified vulnerabilities. o Signature-based recognition cannot detect unknown attacks; they can only detect attacks identified by published signature files. For this reason, it is important to update signature files on a regular basis. • *Anomaly* recognition, also referred to as *behavior* or *heuristic*, monitors traffic to define a standard activity pattern as "normal". o *Clipping levels* or *thresholds* are defined that identify deviations from the norm. o When the threshold is reached, an alert is generated or action taken. o Anomaly-based systems can recognize and respond to some unknown attacks (attacks that do not have a corresponding signature file). o Anomaly-based recognition systems can be fooled by incremental changes within the clipping level that cause the changed state to become the normal level of activity, thus allowing a higher level of irregularity to go unnoticed.
Detection scope	Systems can be classified based on where the system runs and the scope of threats it looks for. • A *host-based* IDS is installed on a single host and monitors all traffic coming in to the host. o The IDS is typically unaware of other devices on the network, but can be detected and could be the target of an attack itself. o The IDS may rely on auditing and logging capabilities of the operating system. o A host-based IDS can analyze encrypted traffic. o Anti-virus software is the most common form of a host-based IDS. • A network-based IDS is a dedicated device installed on the network. It analyzes all traffic on the network. o The IDS is typically implemented as part of a firewall device. o The IDS is typically unaware of individual hosts on the network. It cannot be detected by attacking systems. o It is particularly suited to detecting and blocking port scanning and DoS attacks. o It cannot analyze encrypted traffic.

In addition to implementing an IDS or IPS, you can also catch threats to your network by performing regular monitoring with common network tools.

- Use a packet sniffer to examine network traffic, looking for traffic of a specific type that should not be on your network or for traffic types associated with known attacks.
- Use a port scanner to check for open ports on a system or a firewall. Compare the list of opened ports with the list of ports allowed by your network design and security policy.
  - Close all unused ports.
  - Investigate the cause of incorrectly opened ports. Make sure that administrators do not open ports unnecessarily, or verify that the system does not have malware installed which could have opened ports for its own purposes.
- Run security scanning software on each system to detect malware or other security vulnerabilities (such as opened ports, weak passwords, or missing operating system patches).
- Keep operating systems and applications up to date with the latest patches. Download the recent signature files to protect against attacks.
- Monitor system logs for unusual activity that could indicate an attempted (or successful) attack. Check firewall logs to identify the type of traffic that has been blocked to identify past attempted attacks. If possible, take additional measures to block unwanted traffic before it reaches your network.

Detection and Prevention Facts

An intrusion detection system (IDS) is a special network device that can detect attacks and suspicious activity. There are several ways to describe typical detection systems:

Method	Variations
Response capability	An intrusion detection system can be classified by how it responds when a threat is detected: • A *passive* IDS monitors, logs, and detects security breaches but takes no action to stop or prevent the attack. A passive IDS can send an alert, but it is the network administrator's job to interpret the degree of the threat and to respond accordingly. • An *active* IDS (also called an *intrusion protection system* or IPS) performs the functions of an IDS, but can also *react* when security breaches occur. An IPS: o Can automate responses that may include dynamic policy adjustment and reconfiguration of supporting network devices to block the offending traffic. o Performs behaviors that can be seen by anyone watching the network. Usually these actions are necessary to block malicious activities or discover the identity of an intruder. Updating filters and performing reverse lookups are common behaviors of an active IDS.
Recognition method	The recognition method defines how the system distinguishes attacks and threats from normal activity. • *Signature* recognition, also referred to as *pattern* matching or *dictionary* recognition, looks for patterns in network traffic and compares it to known attack patterns called *signatures*. o IDS signatures are written and updated by the IDS vendor in response to identified vulnerabilities. o Signature-based recognition cannot detect unknown attacks; they can only

	detect attacks identified by published signature files. For this reason, it is important to update signature files on a regular basis. • *Anomaly* recognition, also referred to as *behavior* or *heuristic*, monitors traffic to define a standard activity pattern as "normal". o *Clipping levels* or *thresholds* are defined that identify deviations from the norm. o When the threshold is reached, an alert is generated or action taken. o Anomaly-based systems can recognize and respond to some unknown attacks (attacks that do not have a corresponding signature file). o Anomaly-based recognition systems can be fooled by incremental changes within the clipping level that cause the changed state to become the normal level of activity, thus allowing a higher level of irregularity to go unnoticed.
Detection scope	Systems can be classified based on where the system runs and the scope of threats it looks for. • A *host-based* IDS is installed on a single host and monitors all traffic coming in to the host. o The IDS is typically unaware of other devices on the network, but can be detected and could be the target of an attack itself. o The IDS may rely on auditing and logging capabilities of the operating system. o A host-based IDS can analyze encrypted traffic. o Anti-virus software is the most common form of a host-based IDS. • A network-based IDS is a dedicated device installed on the network. It analyzes all traffic on the network. o The IDS is typically implemented as part of a firewall device. o The IDS is typically unaware of individual hosts on the network. It cannot be detected by attacking systems. o It is particularly suited to detecting and blocking port scanning and DoS attacks. o It cannot analyze encrypted traffic.

In addition to implementing an IDS or IPS, you can also catch threats to your network by performing regular monitoring with common network tools.

- Use a packet sniffer to examine network traffic, looking for traffic of a specific type that should not be on your network or for traffic types associated with known attacks.
- Use a port scanner to check for open ports on a system or a firewall. Compare the list of opened ports with the list of ports allowed by your network design and security policy.
    - Close all unused ports.
    - Investigate the cause of incorrectly opened ports. Make sure that administrators do not open ports unnecessarily, or verify that the system does not have malware installed which could have opened ports for its own purposes.
- Run security scanning software on each system to detect malware or other security vulnerabilities (such as opened ports, weak passwords, or missing operating system patches).
- Keep operating systems and applications up to date with the latest patches. Download the recent signature files to protect against attacks.
- Monitor system logs for unusual activity that could indicate an attempted (or successful) attack. Check firewall logs to identify the type of traffic that has been blocked to identify past attempted

attacks. If possible, take additional measures to block unwanted traffic before it reaches your network.