1. **How to configure the firewall?**

Firewall Configurations

In addition to the various types of firewalls, there are various configuration options. The type of firewall tells you how it will evaluate traffic and hence decide what to allow and not to allow. The configuration gives you an idea of how that firewall is set up in relation to the network it is protecting. Some of the major configurations / implementations for firewalls include the following:

* Network host-based firewall
* Dual-homed host
* Router-based firewall
* Screened host

Each of these is discussed in the following sections.

**Network Host-Based Firewalls**

* A network host-based firewall is a **software solution** installed on an existing machine with an existing operating system.
* The most significant concern in using this type of firewall is that no matter how good the firewall solution is, it is contingent (depend on) upon the underlying operating system.
* In such a situation, it is absolutely critical that the machine hosting the firewall have a **hardened operating system.**

Steps:

1. **Install the Firewall Software:** Ensure the firewall software is installed and up-to-date.
2. **Define Rules and Policies:** Specify what traffic is allowed or blocked based on criteria like **IP addresses, ports, and protocols.**

* Windows Firewall: Use the Windows Firewall with Advanced Security to create inbound and outbound rules.
* Linux iptables: Use commands like iptables -A INPUT -p tcp --dport 22 -j ACCEPT to allow SSH traffic.

1. **Enable the Firewall:** Ensure the firewall is enabled and set to start on boot.

* Windows: Go to Control Panel -> System and Security -> Windows Firewall -> Turn Windows Firewall on or off.
* Linux: Use systemctl enable iptables and systemctl start iptables.

1. **Test the Configuration:** Verify that the firewall rules are working as expected using tools like **nmap** or **netcat**.

**Dual-Homed Host**

* A dual-homed host is a firewall running on a server with at least two network interfaces.
* The server acts as a router between the network and the interfaces to which it is attached.
* To make this work, the automatic routing function is disabled, meaning that an IP packet from the Internet is not routed directly to the network.
* You can choose what packets to route and how to route them.
* Systems inside and outside the firewall can communicate with the dual-homed host but cannot communicate directly with each other.

Steps:

1. **Setup the Host:** Install and configure two network interfaces.

* Interface 1: Connect to the internal network.
* Interface 2: Connect to the external network (e.g., the internet).

1. **Install Firewall Software:** Use firewall software that supports dual-homed configurations, such as iptables on Linux or pfSense.
2. **Configure Forwarding and Filtering:**

* Disable IP Forwarding: Typically, disable IP forwarding to prevent packets from being routed directly between interfaces unless explicitly allowed.
* Linux: echo 0 > /proc/sys/net/ipv4/ip\_forward
* Set Up Rules: Define rules to control traffic between the two interfaces.
* iptables: iptables -A FORWARD -i eth0 -o eth1 -m state --state NEW, ESTABLISHED, RELATED -j ACCEPT

1. **Harden the Host:** Ensure the host itself is secure by disabling unnecessary services, applying updates, and using strong authentication.

**Router-Based Firewall**

As was previously mentioned, you can implement firewall protection on a router. In larger networks with multiple layers of protection, this is commonly the first layer of protection.

Although you can implement various types of firewalls on a router, the most common type used is packet filtering.

If you use a broadband connection in your home or small office, you can get a packet-filtering firewall router to replace the basic router provided to you by the broadband company.

In recent years, router-based firewalls have become increasingly common and are in fact the most common type of firewall used today.

Steps:

1. **Access Router Configuration:** Log in to the router's web interface or command-line interface.
2. **Navigate to Firewall Settings:** Look for sections like "Firewall," "Security," or "Access Control."
3. **Define Rules:** Create rules to allow or block traffic based on criteria such as source/destination IP addresses, ports, and protocols.

* Example: Allow HTTP traffic: Source IP - Any, Destination IP - Any, Protocol - TCP, Port - 80.

1. **Apply and Save Configuration:** Apply the changes and ensure the configuration is saved.
2. **Test the Firewall:** Use network scanning tools to test if the rules are effective.

**Screened Host**

A screened host is really a combination of firewalls. In this configuration, you use a combination of a bastion host and a screening router.

The screening router adds security by allowing you to deny or permit certain traffic from the bastion host. It is the first stop for traffic, which can continue only if the screening router lets it through.

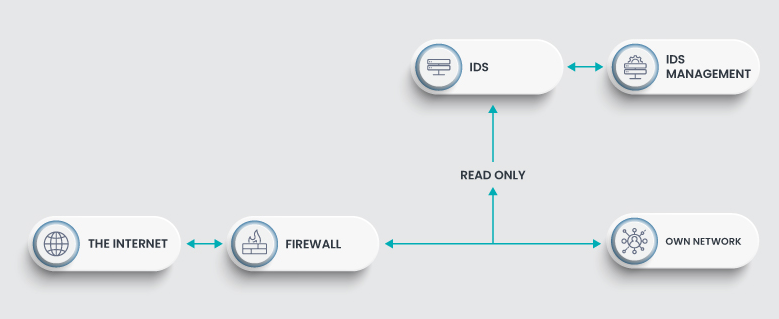
Steps:

1. **Setup the Network:** Configure a DMZ with a firewall separating it from both the internal network and the internet.
2. **Place the Host in the DMZ:** Ensure the screened host is within the DMZ.
3. **Configure the Firewall:**

* DMZ to External Network: Allow necessary services (e.g., web server ports) from the internet to the DMZ.
* DMZ to Internal Network: Restrict access to critical internal resources.
* Example Rule: Allow HTTP traffic from the internet to the web server in the DMZ, but block other traffic.

1. **Harden the Host:** Secure the screened host by disabling unnecessary services, keeping software up-to-date, and using strong passwords.
2. **Monitor and Test:** Regularly monitor traffic and test the configuration to ensure security.
3. **Elaborate Intrusion-Detection system in detail?**

* A system called an intrusion detection system (IDS) observes network traffic for malicious transactions and sends immediate alerts when it is observed.
* It is software that checks a network or system for malicious activities or policy violations.
* Each illegal activity or violation is often recorded either centrally using an SIEM system or notified to an administration.
* IDS monitors a network or system for malicious activity and protects a computer network from unauthorized access from users, including perhaps insiders.
* The intrusion detector learning task is to build a predictive model (i.e. a classifier) capable of distinguishing between ‘bad connections’ (intrusion/attacks) and ‘good (normal) connections’.



**Working of Intrusion Detection System (IDS)**

* An IDS (Intrusion Detection System) monitors the traffic on a computer network to detect any suspicious activity.
* It analyzes the data flowing through the network to look for patterns and signs of abnormal behavior.
* The IDS compares the network activity to a set of predefined rules and patterns to identify any activity that might indicate an attack or intrusion.
* If the IDS detects something that matches one of these rules or patterns, it sends an alert to the system administrator.
* The system administrator can then investigate the alert and take action to prevent any damage or further intrusion.

**Classification of Intrusion Detection System(IDS)**

Intrusion Detection System are classified into 5 types:

1. Network Intrusion Detection System (NIDS): Network intrusion detection systems (NIDS) are set up at a planned point within the network to examine traffic from all devices on the network. It performs an observation of passing traffic on the entire subnet and matches the traffic that is passed on the subnets to the collection of known attacks. Once an attack is identified or abnormal behavior is observed, the alert can be sent to the administrator. An example of a NIDS is installing it on the subnet where firewalls are located in order to see if someone is trying to crack the firewall.
2. Host Intrusion Detection System (HIDS): Host intrusion detection systems (HIDS) run on independent hosts or devices on the network. A HIDS monitors the incoming and outgoing packets from the device only and will alert the administrator if suspicious or malicious activity is detected. It takes a snapshot of existing system files and compares it with the previous snapshot. If the analytical system files were edited or deleted, an alert is sent to the administrator to investigate. An example of HIDS usage can be seen on mission-critical machines, which are not expected to change their layout.
3. Protocol-based Intrusion Detection System (PIDS): Protocol-based intrusion detection system (PIDS) comprises a system or agent that would consistently reside at the front end of a server, controlling and interpreting the protocol between a user/device and the server. It is trying to secure the web server by regularly monitoring the HTTPS protocol stream and accepting the related HTTP protocol. As HTTPS is unencrypted and before instantly entering its web presentation layer then this system would need to reside in this interface, between to use the HTTPS.
4. Application Protocol-based Intrusion Detection System (APIDS): An application Protocol-based Intrusion Detection System (APIDS) is a system or agent that generally resides within a group of servers. It identifies the intrusions by monitoring and interpreting the communication on application-specific protocols. For example, this would monitor the SQL protocol explicitly to the middleware as it transacts with the database in the web server.
5. Hybrid Intrusion Detection System: Hybrid intrusion detection system is made by the combination of two or more approaches to the intrusion detection system. In the hybrid intrusion detection system, the host agent or system data is combined with network information to develop a complete view of the network system. The hybrid intrusion detection system is more effective in comparison to the other intrusion detection system. Prelude is an example of Hybrid IDS.

**Benefits of using an IDS:**

* Improved threat detection: IDSs can identify a wider range of threats than traditional security measures.
* Faster response times: Early detection allows for quicker response to security incidents.
* Enhanced security posture: IDSs provide valuable insights into network activity and can help improve overall security posture.

**Limitations of IDS:**

* Can generate false positives: IDSs may sometimes flag normal activity as suspicious.
* Doesn't prevent attacks: An IDS detects threats, but it typically doesn't take action to stop them. This is the job of an Intrusion Prevention System (IPS).
* Requires ongoing maintenance: The IDS database of threats needs to be regularly updated to stay effective.

**3) Elaborate the concept of Digital certificates in detail? 317**

Another way to provide message integrity and message authentication (and some more security services as we see shortly) is a digital signature. A MAC uses a secret key to protect the digest; a digital signature uses a pair of private-public keys.

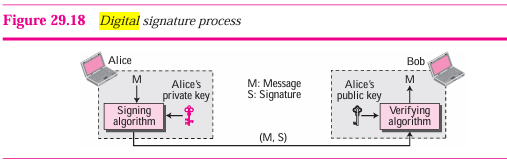
A digital signature uses a pair of private-public keys.

We are all familiar with the concept of a signature. A person signs a document to show that it originated from her or was approved by her. The signature is proof to the recipient that the document comes from the correct entity. When a customer signs a check, the bank needs to be sure that the check is issued by that customer and nobody else. In other words, a signature on a document, when verified, is a sign of authentication the document is authentic. Consider a painting signed by an artist. The signature on the art, if authentic, means that the painting is probably authentic.

When Alice sends a message to Bob, Bob needs to check the authenticity of the sender; he needs to be sure that the message comes from Alice and not Eve. Bob can ask Alice to sign the message electronically. In other words, an electronic signature can prove the authenticity of Alice as the sender of the message. We refer to this type of signature as a digital signature.

Process:

The fig shows the digital signature process. The sender uses a signing algorithm to sign the message. The message and the signature are sent to the receiver. The receiver receives the message and the signature and applies the verifying algorithm to the combination. If the result is true, the message is accepted; otherwise, it is rejected



A conventional signature is like a private “key” belonging to the signer of the document. The signer uses it to sign documents; no one else has this signature. The copy of the signature on file is like a public key; anyone can use it to verify a document, to compare it to the original signature.

In a digital signature, the signer uses her private key, applied to a signing algorithm, to sign the document. The verifier, on the other hand, uses the public key of the signer, applied to the verifying algorithm, to verify the document.

Note that when a document is signed, anyone, including Bob, can verify it because everyone has access to Alice’s public key. Alice must not use her public key to sign the document because then anyone could forge her signature.

Can we use a secret (symmetric) key to both sign and verify a signature? The answer is negative for several reasons. First, a secret key is known by only two entities (Alice and Bob, for example). So, if Alice needs to sign another document and send it to Ted, she needs to use another secret key. Second, as we will see, creating a secret key for a session involves authentication, which uses a digital signature. We have a vicious cycle. Third, Bob could use the secret key between himself and Alice, sign a document, send it to Ted, and pretend that it came from Alice.

A digital signature needs a public-key system. The signer signs with her private key; the verifier verifies with the signer’s public key

We should make a distinction between private and public keys as used in digital signatures and public and private keys as used in a cryptosystem for confidentiality. In the latter, the private and public keys of the receiver are used in the process. The sender uses the public key of the receiver to encrypt; the receiver uses his own private key to decrypt. In a digital signature, the private and public keys of the sender are used. The sender uses her private key; the receiver uses the sender’s public key.

A cryptosystem uses the private and public keys of the receiver: a digital signature uses the private and public keys of the sender.

X.509 is an international standard for the format and information contained in a digital certificate. X.509 is the most common type of digital certificate in the world. It is a digital document that contains a public key signed by the trusted third party that is known as a certificate authority, or CA.

The following are the basic items in an X.509 certificate, though there can be other optional information:

* Version: This is the version of X.509 that this certificate complies with.
* Certificate holder’s public key: This is the primary way of getting someone’s public key from his X.509 certificate.
* Serial number: This is a unique identifier for this certificate.
* Certificate holder’s distinguished name: This is often a domain name or an email address associated with a certificate.
* Certificate’s validity period: One year is the most common validity period.
* Unique name of certificate issuer: This is the certificate authority that issued this certificate.
* Digital signature of issuer: This field and the next are used to verify the certificate.
* Signature algorithm identifier: This identifies the digital signature algorithm used.

**4) Explain the following:**

**1) Snort**

Several vendors supply IDSs, and each has unique strengths and weaknesses. Which system is best for your environment depends on many factors, including the network environment, security level required, budget constraints, and skill level of the person who will be working directly with the IDS. One popular open-source IDS is Snort, which can be downloaded for free from [www.snort.org](http://www.snort.org).

While it is not the only IDS available, it is free, and that makes it an attractive option for many people.

Snort is free and open source, but many people have a great deal of difficulty working with it at first. The slightest error in your configuration file or the command-line startup will cause it to not run correctly.

**2) Honeypot**

A honey pot is an interesting technology. Essentially, it assumes that an attacker is able to breach your network security, and it would be best to distract that attacker away from your valuable data.

Therefore, a honey pot involves creating a server that has fake data —perhaps an SQL server or Oracle server loaded with fake data, and just a little less secure than your real servers. Then, since none of your actual users ever access this server, monitoring software is installed to alert you when someone does access this server.

A honey pot achieves two goals. First, it takes the attacker’s attention away from the data you wish to protect. Second, it provides what appears to be interesting and valuable data, thus leading the attacker to stay connected to the fake server, giving you time to try to track the attacker.

Commercial solutions, such as Specter (www.specter.com), are available. These solutions are usually quite easy to set up and include monitoring/tracking software. You may also find it useful to check out www.honeypots.org for more information on honey pots in general and on specific implementations.

**3) Intrusion Deterrence**

Intrusion deterrence is about making a system appear unappealing or too risky for someone to try and hack into. The idea is to make the potential reward of breaking in seem like it's not worth the effort.

One way to do this is by hiding or disguising the most valuable parts of the system, like important data or resources. This makes it harder for someone to see what they could gain from hacking in.

Another tactic is to make it seem like there's a high chance of getting caught if someone tries to break in. This can be done by putting up warnings or signs that say the system is being monitored closely.

Even if the actual security of the system hasn't changed much, just making it look less valuable or riskier can stop many potential intruders from trying. It's about creating the perception of security to discourage attacks.

**4) Intrusion Deflection**

Intrusion deflection is becoming increasingly popular among security conscious administrators. The essence of it is quite simple: An attempt is made to attract the intruder to a subsystem set up for the purpose of observing intruders. This is done by tricking the intruder into believing that he has succeeded in accessing system resources when, in fact, he has been directed to a specially designed environment. Being able to observe the intruder while he practices his art will yield valuable clues and can lead to his arrest.

Intrusion deflection is often done by using a honey pot. Essentially, you set up a fake system, possibly a server that appears to be an entire subnet. You make that system look very attractive by perhaps making it appear to contain sensitive data, such as personnel files, or valuable data, such as account numbers or research. The actual data stored in this system is fake. The real purpose of the system is to carefully monitor the activities of any person who accesses the system. Since no legitimate user ever accesses this system, it is a given that anyone accessing it is an intruder.

**5) Explain concept of SSL.**

**6) What is firewall? Explain types of firewalls.**

Firewall Types and Components There are numerous types of firewalls and variations on those types. But most firewalls can be grouped into one of the following three families of firewalls:

* Packet inspection
* Stateful packet inspection
* Application

The following sections discuss each of these and assess the advantages and disadvantages of each.

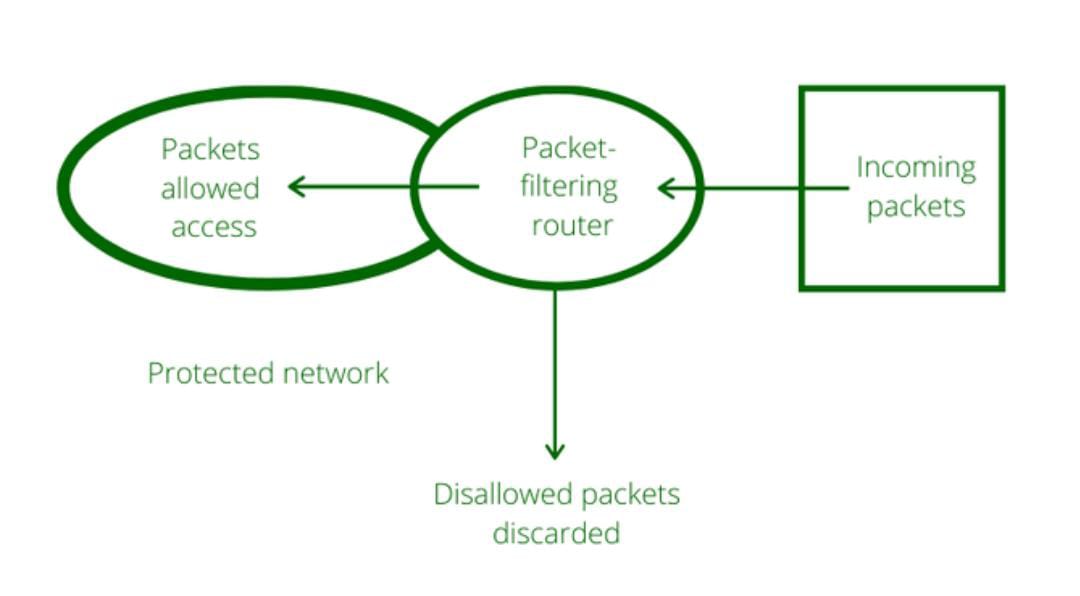
**Packet Filtering**

Basic packet filtering is the simplest form of firewall. It involves looking at packets and checking to see if each packet meets the firewall rules. For example, it is common for a packet filtering firewall to consider three questions:

1. Is this packet using a protocol that the firewall allows?
2. Is this packet destined for a port that the firewall allows?
3. Is the packet coming from an IP address that the firewall has not blocked?

These are three very basic rules. Some packet filter firewalls check additional rules. But what is not checked is the preceding packets from that same source.

Essentially, each packet is treated as a singular event, without reference to the preceding conversation. This makes packet filtering firewalls quite susceptible to some DoS attacks, such as SYN floods.



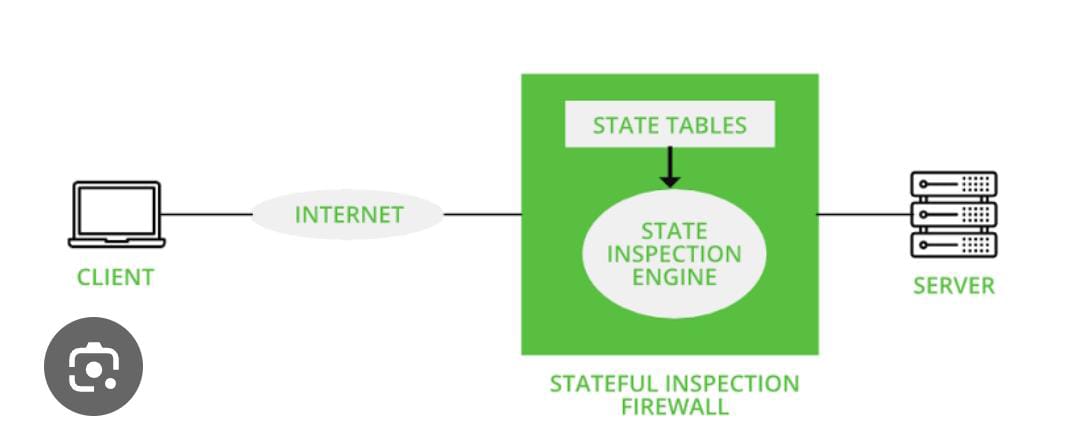
**Stateful Packet Inspection**

Any stateful packet inspection (SPI) firewall will examine each packet and deny or permit access based not only on the examination of the current packet but also on data derived from previous packets in the conversation.

The firewall is therefore aware of the context in which a specific packet was sent. This makes such a firewall far less susceptible to ping floods and SYN floods, as well as less susceptible to spoofing.

For example, if a firewall detects that the current packet is an ICMP packet and a stream of several thousand packets have been continuously coming from the same source IP, the firewall will see that this is clearly a DoS attack, and it will block the packets.

A stateful packet inspection firewall can also look at the actual contents of a packet, which allows for some very advanced filtering capabilities. Most high-end firewalls use the stateful packet inspection method; when possible, this is the recommended type of firewall.



**Application Gateways**

An application gateway (also known as application proxy or application-level proxy) is a program that runs on a firewall.

When a client program, such as a web browser, establishes a connection to a destination service, such as a web server, it connects to an application gateway, or proxy. The client then negotiates with the proxy server in order to gain access to the destination service.

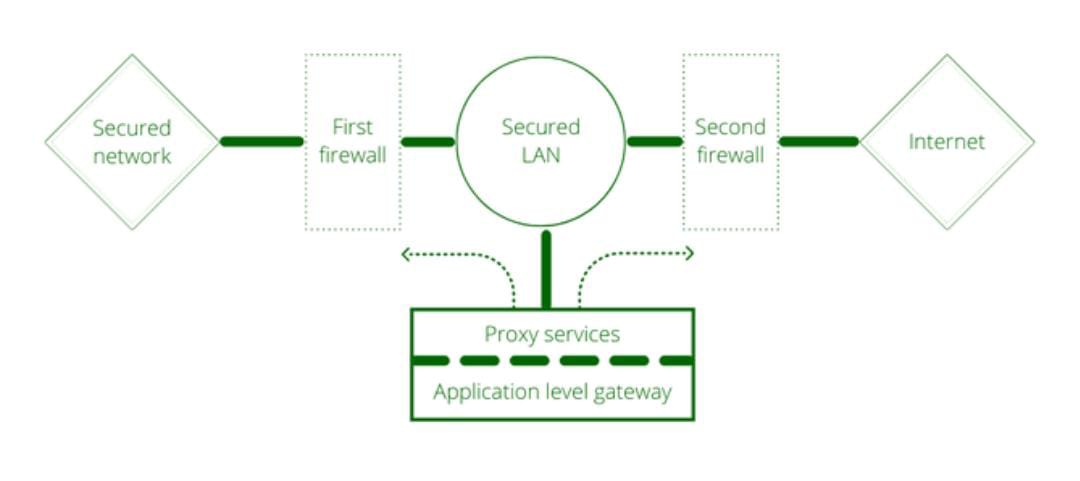
In effect, the proxy establishes the connection with the destination behind the firewall and acts on behalf of the client, hiding and protecting individual computers on the network behind the firewall.

This process actually creates two connections. There is one connection between the client and the proxy server, and there is another connection between the proxy server and the destination.

Once a connection is established, the application gateway makes all decisions about which packets to forward. Since all communication is conducted through the proxy server, computers behind the firewall are protected.

Essentially, an application firewall is used for specific types of applications, such as database or web server applications.

It is able to examine the protocol being used (such as HTTP) for any anomalous behavior and block traffic that might get past other types of firewalls. It is common to have an application firewall that also includes stateful packet inspection.



**7) Explain the concept of VPN in detail?**

Virtual Private Networks A VPN (or virtual private network) essentially provides a way to use the Internet to create a virtual connection between a remote user or site and a central location. The packets sent back and forth over this connection are encrypted, thus making it private. The VPN must emulate a direct network connection.

Three different protocols are used to create VPNs:

* Point-to-Point Tunneling Protocol (PPTP)
* Layer 2 Tunneling Protocol (L2TP)
* Internet Protocol Security (IPsec)

**Point-to-Point Tunneling Protocol**

Point-to-Point Tunneling Protocol (PPTP) is the oldest of the three protocols used in VPNs. It was originally designed as a secure extension to Point-to-Point Protocol (PPP). PPTP was originally proposed as a standard in 1996 by the PPTP Forum—a group of companies that included Ascend Communications, ECI Telematics, Microsoft, 3Com, and U.S. Robotics. It adds the features of encrypting packets and authenticating users to the older PPP protocol. PPTP works at the data link layer of the OSI model (discussed in Chapter 2, “Networks and the Internet”).

PPTP offers two different protocols for authenticating the user: Extensible Authentication Protocol (EAP) and Challenge Handshake Authentication Protocol (CHAP). EAP was actually designed specifically for PPTP and is not proprietary. CHAP is a three-way process whereby the client sends a code to the server, the server authenticates it, and then the server responds to the client. CHAP also periodically reauthenticates a remote client, even after the connection is established.

PPTP uses Microsoft Point-to-Point Encryption (MPPE) to encrypt packets. MPPE is actually a version of DES. DES is still useful for many situations; however, newer versions of DES, such as DES 3, are preferred.

**Layer 2 Tunneling Protocol**

Layer 2 Tunneling Protocol (L2TP) was explicitly designed as an enhancement to PPTP. Like PPTP, it works at the data link layer of the OSI model. It has several improvements over PPTP. First, it offers more and varied methods for authentication: PPTP offers two methods (CHAP and EAP), whereas L2TP offers five (CHAP, EAP, PAP, SPAP, and MS-CHAP).

Besides making more authentication protocols available for use, L2TP offers other enhancements. PPTP will only work over standard IP networks, whereas L2TP will work over X.25 networks (a common protocol in phone systems) and ATM (Asynchronous Transfer Mode a high-speed networking technology) system. L2TP also uses IPsec for encryption.

**IPsec**

Internet Protocol Security (IPsec) is the newest of the three VPN protocols. One of the differences between IPsec and the other two methods is that it encrypts not only the packet data (recall the discussion of packets in Chapter 2) but also the header information. It also has protection against unauthorized retransmission of packets. This is important because one trick that a hacker can use is to simply grab the first packet from a transmission and use it to get his own transmissions to go through. Essentially, the first packet (or packets) has to contain the login data. If you simply re-send that packet (even if you cannot crack its encryption), you will be sending a valid logon and password that can then be followed with additional packets. Preventing unauthorized retransmission of packets prevents this from happening.

* IPsec operates in one of two modes: Transport mode, in which only the payload is encrypted, and Tunnel mode, in which both data and IP headers are encrypted. Following are some basic IPsec terms:
* Authentication headers (AHs) provide connectionless integrity and data origin authentication for IP packets.
* Encapsulating Security Payload (ESP) provides origin authenticity, integrity, and confidentiality protection of packets. It offers encryption-only and authentication-only configurations.
* Security associations (SAs) provide the parameters necessary for AH or ESP operations. SAs are established using Internet Security Association and Key Management Protocol (ISAKMP).
* Internet Security Association and Key Management Protocol (ISAKMP) provides a framework for authentication and key exchange.
* Internet Key Exchange (IKE and IKEv2) is used to set up a SA by handling negotiation of protocols and algorithms and to generate the encryption and authentication keys to be used.

Essentially during the initial establishment of an IPsec tunnel, SAs are formed. These SAs have information such as what encryption algorithm and what hashing algorithms will be used in the IPsec tunnel. (Recall that we discussed encryption in some depth in Chapter 8.) IKE is primarily concerned with establishing these SAs. ISAKMP allows the two ends of the IPsec tunnel to authenticate to each other and to exchange keys.