APPLICATION AND WEB SECURITY

Syllabus --- 08 Marks 04 Hours

Objectives:

➢ To understand application hardening and patches.
➢ To understand web security.

Contents:

6.1 Application hardening, application patches, web servers, active directory.
6.2 Web security threats, web traffic security approaches,
6.3 Secure socket layer and transport layer security,
6.4 secure Electronic transaction.

Hardening of Operating System

Hardening

The Process Of securing an OS is called as hardening... Main Intension is to make the system more resistant to attack. Each OS has its own approach to security. The Process of hardening is generally seen, but different steps to be taken to secure each OS...

Operating System Hardening

• The operating system(os) is the basic software which handles input, output, display, memory management, and all the other detailed tasks to support the user environment and associated applications.

• Many of users are friendly with the Microsoft’s operating systems; Windows 95, Windows 98, Windows NT, Windows 2000, Windows ME, and Windows XP. Majority of home and business PCs run some version of a Microsoft operating system. Other users may be familiar with Mac OS, Solaris, or with UNIX operating system.

• A network operating system (NOS) is an operating system that includes additional functions and capabilities to help in connecting computers and devices like printers, to a local area network (LAN). Vary common network operating system is Novell’s Networks.

• Many modern operating system, including Windows 2000, Solaris, and Linux uses operating system and network operating system as they perform all the basic function and provide enhanced capabilities for connecting to LANs.

• Developers and manufactures of operating system have a same common problem. There is no possible way to anticipate may different configurations and variations that the group of users will require from their products.

• So, rather wasting time and money to meet every need, manufactures provide a default installation for their products which usually contains the base operating
The manufactures will not provide security, while the operating systems can be used for variety of purposes, and can be placed in any number of logical locations. But the manufacturer may provide some recommendations or simplified tools and settings to facilitate security of the system, but in general the end users are responsible for securing their own systems.

Generally this will have removing unnecessary applications and utilities, disabling unnecessary services, setting of appropriate permissions on files, and updating the operating system and application code to the latest version.

This process of securing an operating system is known as operating system more secure.

Each operating system has its own approach for security, and while the process of hardening is generally the same, but different steps must be taken to secure each operating system.

General Steps for Securing Windows Operating System

As windows 2000 or XP operating systems are most popular systems for business & desktops, so we will focus on these two operating systems only. The following are general guide for securing Windows operating systems such as Windows 2000 or XP Professional.

1. **Disable all unnecessary service** :- in general, Windows systems will serve one main purpose (web server, mail serve, DNS serve, domain/ login server, and so on). One you have strong with that the main purpose of the system will be, then disable a service which is not necessary for that purpose.

2. **Restrict permissions on files and access to the Registry** - This step may take some to restrict who can read, write, and execute certain files and can provide some more needed security. Additionally, the Windows Registry must be protected to ensure that entries are not modified or deleted.

3. **Remove unnecessary programs** - any applications or utility which is not required should be removed. This reduces the chances of an attacker exploiting a weakness or enabling unneeded services.

4. **Apply the latest patches and fix** - Make sure that the operating system and all applications have the latest vendor-supplied patches applied.

5. **Remove unnecessary user accounts and ensure password guidelines are in place** - Default accounts like guest should be disabled or removed. Password guidelines should be enabled and enforce to check that a user chooses appropriate password.
Linux Hardening

- By above general rules, any operating system can be made relatively secure. The UNIX operating systems can be more secure than default installations of Windows Operating systems.

- But the UNIX systems are not completely secure by default and it also needs some additional security configuration.

- Similar to Windows operating systems, UNIX also need to be carefully examined, modified, and baseline to provide secure operations.

- Making UNIX systems secure can be more challenging than securing Windows systems because the operating system is very powerful and flexible. But this is all depending on the skill and knowledge of the system administrator because so much control is placed in the administrator’s hand.

- In Windows systems there is a single manufacturer who provides specific guidelines and step-by-step checklists, but in UNIX systems there are many general and version-specific guidelines that must be adapted and applied to your specific version of UNIX to complete the basic process.

- This general UNIX base-lining is similar to Windows operating systems that means disable unnecessary services, restrict, permissions on files and directions, apply password guidelines, remove unnecessary software, apply patches, and remove unnecessary users.

- Some UNIX versions provide GUI-based tools for such tasks, whereas others need the help of administrator’s to manually edit configuration files.

- UNIX systems are easier to secure and baseline when they are providing a single service or performing a single function, like acting as SMTP or web server.

- Before performing any installing or base-lining, the purpose of the systems should be defined and all required capabilities and function should be identified.

- Advantage of UNIX systems is that the control over what does or does not get installed on the system. During the installation process, it is easy to select which services and applications are placed on the systems. This means that the administrator knows and understands the system purpose.

- Apart from of the installation decision, the administrator may remove the applications or components that are no longer needed.

- In UNIX systems package managers will enable administrators to automatically remove unnecessary components and applications.
UNIX systems services can be controlled by number of different mechanisms. For example, as the root user, an administrator can start and stop services manually from command line. The operating system can also stop and start services automatically through configuration files usually it contained in the “/etc” directory.

UNIX systems have different run levels, where the systems can be configured to bring up different services depending on the level selected.

On UNIX system by using the process status or by ps command, you can see which processes, applications, and services are running.

An administrator can identify the service by its unique process identifier or PID. To stop a running service the process is identified by PID and then kill command is use to stop the service. When the system is rebooted, if you want to avoid any service from starting then you need to modify the appropriate run levels to remove the service or modify the configuration files which control the services.

With the help of GUI and command line interface UNIX systems account can be controlled. The user information can be found in the password file located in the /etc directory. By manually editing this file, you can add, delete or modify user accounts on the systems. By examining this file an administrator can see which user accounts exist on the system and then determine which accounts to remove or disable.

In some cases, a patch will consist of a series of manual steps requiring the administrator to replace files, change permissions, and alter directories. In cases, the patches are executable scripts or utilities that perform the patch actions automatically.

In some UNIX versions like Red Hat and Solaries, have built-in utilities which handles the patching process. In that case, the administrator downloads a specific formatted file i.e. the patching utility then processes it to performed any modifications or updates that need to be made.

**Weakness Of Unix/Linux And Windows**

Following are weaknesses of windows operating system:

- Windows OS is commonly used OS. Many users don't understand the security risk related to system while configuring it, so this will cause for different attacks.

- Windows OS installation is insecure because it includes hidden shares, blank passwords & it will not provide protection for known vulnerabilities.

- For the administrator it is difficult to understand how to properly use & configure the software on various hardware setups.
Windows OS provides poor auditing like Event Viewer with cryptic message & missing information.

Windows OS is very tightly integrated. It slows down after running 24 hours.

Less actual control over files.

Microsoft windows is not open source and the majority of windows programs are not open source.

Microsoft windows always provides additional features to their product but it requires backward compatibility with older, less secure version.

**Application Hardening**

Application hardening is securing an application against local and internet based attacks. It is a secure feature designed to prevent exploitation of various types of vulnerabilities in software application. Application hardening is the process to address application security weakness. To mitigate the weakness of application, a number of latest software patches, hot fixes, updates and secured version of protocols. This chapter examines the various security protocols associated with the internet and web security. Since the internet and web technology is vast and various security mechanisms exist for specialized internet and web services, such as e-mail, Electronics commerce and payments, etc.

The number of individuals and companies with internet access is expanding rapidly and all of these have graphical web browsers. As a result, businesses are enthusiastic about setting up facilities on the web for electronic commerce. But the reality is that the internet and the web are extremely vulnerable to compromises of various sorts. As businesses wake up to this reality, the demand for secure web services grows because virtually all businesses, most government agencies, and many individuals now have web sites.

The topic of web security is a very broad one. In this chapter, the general requirements for web security are discussed and then focus on two standardized schemes that are becoming increasingly important as a part of web commerce.

**What is Application Hardening?**

- Hardening is a step by step process of securely configuring a system to protect it against unauthorized access, while also taking steps to make the system more reliable.
- Generally, anything that is done in the name of system hardening ensures that the system is secure and reliable.
- Hardening refers to providing various means of protection in a computer system.
- Protecting in layers means to protect at the host level, the application level, the operating system level, the user level, the physical level and all the sublevels in the network.
- Each level requires a unique method of security. A hardened computer system is a more secure computer system.
Chapter-6 : Application and Web Security

- The purpose of hardening is to make your systems and networks a little harder to break into than your neighbors.
- Application hardening is securing an application against local and internet based attacks. Application hardening is a secure feature designed to prevent exploitation of various types of vulnerabilities in software application.
- It is simply similar to hardening operating system. It is possible to remove the functions or components that user don’t need. It can be restricted access and make sure the application is kept up-to-date with patches.
- Maintaining application security is very important because we need to make applications accessible to users.
- Application hardening can be done by changing the default application configuration.
- Application hardening can be maintained by implementing the latest software patches, hotfixes, and updates.

Need of Hardening

- All computer systems are insecure to some degree.
- It is easy to be cracked.
- The reaction of vulnerabilities discovered.
- The reverse of hacking.
- It is a part of security lifecycle.
- Most crackers are lazy. They won’t keep bothering user unless they have an easy way in. If there is no hardening then, hackers move quickly.
- User system might be hijacked without their knowledge, and then used to attack another system, or spread viruses, or distribute illegal content such as pornography or software.
- User Company’s proprietary information could be stolen; money could be wasted paying employees to sit around, unable to do their work while the system is down.

Goal of Hardening

The goal of Application Hardening is to eliminate as many risks and threats to a computer system. Application Hardening activities for a computer system can include:

- Keeping security patches and hot fixes updated.
- Monitoring security bulletins that are applicable to a system’s operating system and applications.
- Installing a firewall.
- Closing certain ports such as server ports.
- Not allowing file sharing among programs.
- Installing virus and spyware protection, including an anti-adware tool so that malicious software cannot gain access to the computer on which it is installed.
- Keeping a backup, such as a hard drive, of the computer system.
- Disabling cookies.
- Creating strong passwords.
- Never opening e-mails or attachments from unknown senders.
- Removing unnecessary programs and user accounts from the computer.
- Using encryption where possible.
Chapter-6 : Application and Web Security

- Hardening security policies, such as local policies relating to how often a password should be changed and how long and in what format a password must be in.

Benefits of Hardening

- It establishes the trust relationship and enhances the business with other company and hence does more business.
- User can have more confidence in the integrity of their data.
- Performance improvements can be experienced since unnecessary services are removed and inefficiencies in system configuration are detected.
- The company’s reputation is protected.
- Clients are happier as a result of fewer system failures or delays.

Application patches

- A patch is a piece of software which is designed to fix problems, or update a computer program or its supporting data.
- This includes fixing security vulnerabilities and other bugs and improving the usability or performance. They are supplied from the vendor to sell the application.
- By not applying a patch user might be leaving the door open for malware to come in. Malware exploits flaws in a system in order to do its work, while the time frame between an exploit and when a patch is released is continuously getting shorter.
- Defects in clients like web browsers, e-mail programs, image viewers, instant messaging software, and media players may allow malicious websites, e-mail messages, images, and sound files to infect or compromise user computer with no action on your part other than viewing or listening to the website, message, or media.
- Application patches can be downloaded from the vendor’s web site or FTP site, or they can be received on a CD.
- A patch is small binary application that, when run, automatically replaces defective application binaries with updated ones. The patch may also change settings or modifies' configuration files.
- Application patches are likely to come in three varieties
  - (1) Hotfixes.  
  - (2) Service Pack.  
  - (3) Patches.  
  - (4) Updates.

Hotfixes

- It is a small software update designed to address a specific software problem such as a buffer flow in an application that exposes the system to attacks.
- There are typically reactions to the developed problems and are produced and delivered quickly.
- Hotfixes address critical security related issues and should be applied to the affected application or OS as soon as possible.
- A hotfix or Quick Fix Engineering update (QFE update) is a single, cumulative package that includes information (often in the form of one or more files) that is used to address a problem in a software product (i.e. a software bug).
- Typically, hotfixes are made to address a specific customer situation.
Chapter-6 : Application and Web Security

- The term "hotfix" was originally applied to software patches that were applied to "hot" systems; that is, systems which are live, currently running and in production status rather than development status.
- For the developer, a hotfix implies that the change may have been made quickly and outside normal development and testing processes. This could increase the cost of the fix by requiring rapid development, overtime or other urgent measures.
- For the user, the hotfix could be considered more risky or more likely to fail to resolve the problem. This could cause an immediate loss of services and/or revenue, so depending on the severity of the bug, it may be desirable to delay a hotfix.
- The risk of applying the hotfix must be considered against the risk of not applying it, because the problem to be fixed might be so critical that it could be considered more important than a potential loss of service (e.g., a major security breach).
- Similar use of the terms can be seen in hot swappable disk drives. The more recent usage of the term is likely due to software vendors making a distinction between a hotfix and a patch.

**Note:** Hotfixes are also sometimes referred to as LDRs, or QFE's (Quick Fix Engineering). The term QFE is an old term that is mostly no longer used in reference to current versions of Windows.

**Example : Microsoft Hotfix**
- When an individual customer reports a bug to Microsoft for a specific scenario, the WinSE team releases Hotfixes to address these problems.
- Hotfixes are not meant to be widely distributed and go through a limited amount of testing due to the customer's need for an urgent fix.
- Hotfixes are developed in a separate environment than the regular Updates. This allows Microsoft to release Updates that do not include the Hotfix files, thereby minimizing risk for the customer.
- Once the Hotfix is ready and packaged by WinSE, a KB article is written describing the problem, with instructions on how to obtain the Hotfix. Microsoft recommends that only customers experiencing the particular problem install the Hotfix for that problem.

**Patches**
- This form is used to a more formal. Larger software update that may address several or many other software problems.
- Patches often contain enhancements or additional capabilities as well as fixes for known bugs.
- Patches are usually developed over a long period of time.
- A **patch** is a piece of software designed to update a computer program or its supporting data, to fix or improve it. This includes fixing security vulnerabilities and other bugs, and improving the usability or performance.
- Though meant to fix problems, poorly designed patches can sometimes introduce new problems. In some special cases updates may knowingly break the functionality, for instance, by removing components for which the update provider is no longer licensed or disabling a device.
- Patch management is the process of using a strategy and plan of what patches should be applied to which systems at a specified time.

*Notes Prepared by – Prof. Manoj Kavedia (9860174297, 9324258878)*
Chapter-6 : Application and Web Security

- Patches for proprietary software are typically distributed as executable files instead of source code. This type of patch modifies the program executable—the program the user actually runs—either by modifying the binary file to include the fixes or by completely replacing it.
- Patches can also circulate in the form of source code modifications. In these cases, the patches (also called "diffs") consist of textual differences between two source code files. These types of patches commonly come out of open source projects. In these cases, developers expect users to compile the new or changed files themselves.
- Because the word "patch" carries the connotation of a small fix, large fixes may use different nomenclature. Bulky patches or patches that significantly change a program may circulate as "service packs" or as "software updates". Microsoft Windows NT and its successors (including Windows 2000, Windows XP, and later versions) use the "service pack" terminology.

Service Pack:
- In computing, a service pack or SP (in short SP) or a feature pack (FP) comprises a collection of updates, fixes, or enhancements to a program delivered in the form of a single installable package.
- Companies often release a service pack when the number of individual patches to a given program reaches a certain (arbitrary) limit, or the software release has shown to be stabilized with a limited number of remaining issues based on users' feedback and bug tracking such as bugzilla.
- In large software applications such as office suites, operating systems, database software, or network management, it is not uncommon to have a service pack issued within the first year or two of a product's release.
- Installing a service pack is easier and less error-prone than installing many individual patches, even more so when updating multiple computers over a network, where service packs are common.
- Service packs are usually numbered, and thus shortly referred to as SP1, SP2, SP3 etc. They may also bring, besides bug fixes, entirely new features, as is the case of SP2 of Windows XP, or SP3 and SP4 of the heavily database dependent
- This form is given to a larger collection of patches and hotfixes rolled into a single one larger package.
- These are designed to bring system up to the latest known performance level.
- The administrator should download dozens of updates separately.

Updates:
- A new version of a software or hardware product designed to replace an older version of the same product is called as updation.
- A computer program can be upgraded to improve the usability or the performance

Benefits of software update

1. New features not available or found in previous versions.
2. Often, the new version of a program has better stability and increased performance.
3. After so long an older software program will be discontinued and often no longer supported.

**Examples Microsoft Updates**
- These Updates are typically available on Windows Update.
- They frequently contain security fixes, and from time to time also contain reliability rollup packages.
- These updates are thoroughly tested and Microsoft highly recommends that you update your computer with these releases.
- In fact, most are automatically downloaded to your machine if you have Windows Update turned on.
- In most cases, Update releases are also available as standalone downloads from the download center.

**WebServer**


A web server is a specialized type of file server. Its job is to retrieve files from the server’s hard drive, format the files for the web browser, and send them out via the network. Web servers are designed to do a great job of sending static content out to a large number of users. The pages delivered by the server are expected to be the same for everyone who visits the server.

Fig. shows that a web server is a program that, using the client/server model and the World Wide Webs Hypertext Transfer Protocol (HTTP), serves the files that form web pages to Web users (whose computers contain HTTP clients that forward their requests). Every computer on the internet that contains a web site must have a web server program. Two leading web servers are Apache widely-installed web server, and Microsoft’s Internet Information Server (IIS)

**Function of web servers:**
The function of a typical web server fig. is shown in
The primary function of a web server is to deliver web pages to clients. The communication between client and server takes place using the Hypertext Transfer Protocol (HTTP).

The user requests a web page. The web server finds the web page file in a local directory and sends it back out to the user. Pages delivered are most frequently HTML documents, which may include images, style sheets and scripts in addition to text content.

When graphic files are requested, the same thing happens. The web server finds the requested graphic files and sends them back to the user.

The web server standards were originally designed to publish static documents on the Internet. There was a limited capability for accessing dynamic content, but this was never intended to support high volume, highly interactive web applications.

Then, as the internet and web browsers became popular in the mid-1990’s, organizations saw an opportunity to provide web pages that contained dynamic content like stock prices, weather information, inventory levels, and shipping status for a package.

Web browsers became such a pervasive way to access information that users and organizations desired to access a wide variety of information and applications through their Web browser. So, Web servers had to be extended to allow software application development and access to databases.

A web server can (and usually does) contain more than one website. In fact, many hosting companies host hundreds, or even thousands of websites on a single web server.

Each website is usually assigned a unique IP address which distinguishes it from other IP address is websites on the same machine. This used to resolve the domain name.
Chapter-6: Application and Web Security

- It is also possible to configure multiple websites without using different IP addresses using host headers and/or different ports. This can be useful in a development environment and is quite easy to do.

Features of web servers

- Most web servers have features that allow user to do the following:
- Create one or more websites that is set up the website in the web server, so that the website can be viewed via HTTP.
- Configure log file settings, including where the log files are saved, what data to include on the log files etc. (Log files can be used to analyze traffic etc).
- Configure website/directory security. For example, which user accounts are/aren’t allowed viewing the website, which IP addresses are/aren’t, allowed to view the website, etc.
- Create an FTP site. An FTP site allows users to transfer files to and from the site.
- Create virtual directories, and map them to physical directories.
- Configure/nominate custom error pages. This allows you to build and display user friendly error messages on your website. For example, you can specify which page is displayed when a user tries to access a page that doesn’t exist (i.e. a “404 error”).
- Specify default documents. Default documents are those that are displayed when no file name is specified.

Advantages:

- Web server is really a category of web hosting services that delivers network services for every single individuals and many group to obtain a websites to stay connected with the entire world through World-Wide-Web.
- Web server is normally used hosting type of all several primary types because it found to be more effective for the users.
- Shared web servers can comply with the unlimited free space.
- The databases provided by the web server’s service provider will vary greatly on the type of shared hosting plan that user have opted.
- The best way to try to choose the shared web-hosting providers is to have money-back guarantee on the services provided.
- Data backups shall be an integral part of the shared web server’s services as it saves you from reworking on any website. The web host shall ideally provide free data backup for every shared account.
- The reliability of web hosting services is majorly based on the technical support they provide. It is easy to host website on any web host but the real problem may start after accomplishing the start up procedures.

Disadvantages:

- It can be more costly as opposed to electronic website hosting.
- It is well suited for most online enterprise, but it’s not suggested regarding company and that is superior and needs customized design & design and style using some other web hosting server, called clustered web hosting machine.
- If a user is actually initiator inside web organization, digital hosting is cheap as well as sufficient, thus web server is simply smart choice, till the consumer tends

Notes Prepared by – Prof. Manoj Kavedia (9860174297, 9324258878)
Chapter-6 : Application and Web Security
to make development in the enterprise degree along with knobs his or her organization of utilizing many web sites.
• For long business expression affiliate business plans, improve involving dedicated web hosting is required in the future for compared to cost.

Web Application Server
• Web application servers are designed specifically to extend web servers to support dynamic content. The application server software “hooks in” to the web server software and automatically intercepts any user requests for dynamic content. The web server still sends out static web pages and graphic files–Just like before.
• But now, the application server can create dynamic content by mixing data with templates, running programs, or by accessing databases.

How a Web Application server works
• There are a variety of Web Application Server product available. Some of the more popular products include:
  ✓ Macromedia: Cold Fusion
• Cold Fusion is inexpensive and easy to learn resulting in low cost for both development and implementation. Its may be less applicable for complex or high-volume projects.
  ✓ Microsoft: Active Server Pages and .Net
Active Directory

Source - http://windows.uwaterloo.ca/server/disaster/ms_docs/disaster.htm

- Active Directory is a directory service. The term directory services refers to two things such as a directory where information about users and resources is stored and a service or services that let user access and manipulate those resources.
- Active Directory is a way to manage all elements of network, including computers, groups, users, domains, security policies, and any type of user-defined objects.
- Active Directory is a special-purpose database, it is not a registry replacement. The directory is designed to handle a large number of read and search operations and a significantly smaller number of changes and updates.
- Active Directory data is hierarchical, replicated, and extensible. Because it is replicated, users do not want to store dynamic data, such as corporate stock prices or CPU performance. If user data is machine-specific, store the data in the registry.
- Typical examples of data stored in the directory include printer queue data, user contact data, and network/computer configuration data.
- The Active Directory database consists of objects and attributes. Objects and attribute definitions are stored in the Active Directory schema.
- Active Directory provides a central service for administrators to organize network resources, manage users, computers, and applications. Many different objects can be stored in the Active Directory, including, Users and Groups. The illustration in Figure below depicts many different objects stored centrally in the active Directory.
- A directory, in the most generic sense, is a comprehensive listing of objects. A phone book is a type of directory that stores information about people, businesses, and government organizations. Phone books typically record names, addresses, and phone numbers.
- Active Directory is similar to a phone book in several ways, and it is far more flexible. Active Directory will store information about organizations, sites, systems, users, shares, and just about any other network object that you can imagine.
Chapter-6 : Application and Web Security
Not all objects are as similar to each other as those stored in the phone book, so Active Directory includes the ability to record different types of information about different objects. Active Directory stores information about network components. It allows clients to find objects within its namespace. The term namespace (also known as console tree) refers to the area in which a network component can be located.

- Example: The table of contents of this book forms a namespace in which chapters can be resolved to page numbers. DNS is a namespace that resolves host names to IP addresses.
- Telephone books provide a namespace for resolving names to telephone numbers.
- Active Directory provides a namespace for resolving the names of network objects to the objects themselves. Active Directory can resolve a wide range of objects, including users, systems, and services on a network.
- Everything that Active Directory tracks is considered an object. An object is any user, system, resource, or service tracked within Active Directory. The generic term object is used because Active Directory is capable of tracking a variety of items, and many objects can share common attributes.
- Attributes describe objects in Active Directory.
Example:

- All User objects share attributes to store a user name, full name, and description. Systems are also objects, but they have a separate set of attributes that include a host name, an IP address, and a location.
- The set of attributes available for any particular object type is called a schema. The schema makes object classes different from each other. Schema information is actually stored within Active Directory, which allows administrators to add attributes to object classes and have them distributed across the network to all corners of the domain, without restarting any domain controllers.
- A container is a special type of object used to organize Active Directory. It does not represent anything physical, like a user or a system. Instead, it is used to group other objects. Container objects can be nested within other containers.
- The term tree is used to describe a set of objects within Active Directory. When containers and objects are combined hierarchically, they tend to form branches, hence the term. A related term is contiguous subtree, which refers to an unbroken branch of the tree.
- Continuing the tree metaphor, the term forest describes trees that are not part of the same namespace but that share a common schema, configuration, and global catalog. Trees in a forest all trust each other, so objects in these trees are available to all users if the security allows it. Organizations that are divided into multiple domains should group the trees into a single forest.
A site is a geographical location, as defined within Active Directory. Sites correspond to logical IP subnets, and as such, they can be used by applications to locate the closest server on a network. Using site information from Active Directory can profoundly reduce the traffic on wide area networks.

**Features Of Active Directory**

**Scalability**
AD DS includes one or more domains, each with one or more domain controllers, enabling user to scale the directory to meet any network requirements.

**Flexible and global searching:**
Users and administrators can use desktop tools to search AD DS. By default, searches are directed to the global catalog, which provides forest-wide search capabilities.

**Storage for application data:**
AD DS provides a central location to store data that is shared between applications and with applications that need to distribute their data across entire windows networks.

**Systematic synchronization of directory updates:**
Updates are distributed throughout the network through secure and cost-efficient replication between domain controllers.

**Remote administration:**
User can connect to any domain controller remotely from any windows-based computer that has administrative tools installed.

**Single, modifiable, and extensible schema:**
The schema is a set of objects and rules that provide the structure requirements for AD DS objects. User can modify the schema to implement new types of objects or object properties.
Light weight Directory Access Protocol (LDAP):
LDAP is the industry standard directory access protocol, making AD DS widely accessible to management and query applications. ADDS supports LDAPv3 and LDAPv2.

Advantages:
- A central location for network administration and delegation of administrative authority.
- Users have access to objects representing all network users, devices, and resources and the ability to group objects for ease of management and application of security and Group Policy.
- Information security and single sign-on for user access to network resources. Tight integration with security eliminates costly tracking of accounts for authentication and authorization between systems.
- A single user name and password combination can identify each network user, and this identity follows the user throughout the network.
- Integration of object names with Domain Name System (DNS), the internet-standard computer location system. AD uses DNS to implement an IP-based naming system so that AD services and domain controllers are locatable over standard IP both on intranets and the internet.

Disadvantages
- In AD a client cannot move from one group to another, has to be done from the active directory only.
- If copied the client will be available as per the organizational unit structure in the existing group but will be copied to the new group where policy from new group will be applied.
- During the copy procedure, it’s possible that a single client may use more than I seat, causing over deployment.
- Not easy to apply policies for various components.
- Ideal only for large environments.
- Imported Organizational Units are read only. Data in the Organizational Unit cannot be changed manually.

Web Security Threats
- Security threats to web sites and web applications (webapps) come in many forms. Data centers and other assets used for hosting web sites and their associated systems need to be protected from all types of threat.
- The World Wide Web is fundamentally a client/server application running over the internet and TCP/IP intranets.
- Threats should be identified using application threat modeling and then evaluated with a vulnerability assessment.
- Vulnerabilities can be removed or reduced and countermeasures put in place to mitigate the effects of an incident should the threat be realized.

Different Web Security Threats
Physical threats include loss or damage to equipment through fire, smoke, water and other fire suppressants, dust, theft and physical impact.

Physical impact may be due to collision or the result of malicious or accidental damage by people.

Power loss will affect the ability for servers and network equipment to operate depending upon the type of back-up power available and how robust it is.

Human error:

- Errors caused by people include operator/user error such as accidental deletion of data or destruction of software programs, configurations or hardware.
- The other major error caused by people is leaving weaknesses (vulnerabilities) in software. This can include escalation of privileges, authentication which can be bypassed, incorrect implementation of encryption, failure to validate input and output data, weak session management, failure to handle errors correctly, etc.
- Good programming practices can reduce the vulnerabilities which human error can exploit.

Malfunction

- Both equipment and software malfunction threats can impact upon the operations of a website or web application.
- All assets required for the operation of the web system must be identified to be able to evaluate the threats.
- Malfunction of software is usually due to poor development practices where security has not been built into the software development life cycle.

Malware:

- Malware or malicious software, comes in many guises.
- Web servers are popular targets to aid distribution of such code and sites which have vulnerabilities that allow this are popular targets.

Virus Threats:

A computer virus is a program written to alter the way a computer operates, without the permission or knowledge of the user. A virus replicates and executes itself, usually doing damage to your computer in the process.

Spyware Threats:

A serious computer security threat, spyware is any program that monitors your online activities or installs programs without your consent for profit or to capture personal information. We've amassed a wealth of knowledge that will help you combat spyware threats and stay safe online.

Worms

A worm can be injected into a network by any types of means, like an USB stick or an email attachment. Email worm tends to send itself to all email addresses it finds on the infected PC. The email then appears to originate from the infected user, who may be on your trusted senders’ list, and catch you off guard.

Spoofing:

- Spoofing where a computer assumes the identity of another and masquerading where a user pretends to be another.
Chapter-6 : Application and Web Security

- Usually with higher privileges, can be used to attack web systems to poison data deny service or damage systems.

Hackers Threat

People, not computers, create computer security threats and malware. Hackers are programmers who victimize others for their own gain by breaking into computer systems to steal, change or destroy information as a form of cyber-terrorism. What scams are they using lately? Learn how to combat dangerous malware and stay safe online.

Phishing Threats

Masquerading as a trustworthy person or business, phishers attempt to steal sensitive financial or personal information through fraudulent email or instant messages. Internet Based Attacks While your computer is connected to the Internet it can be subject to attack through your network communications. Some of the most common attacks include:

Bonk – An attack on the Microsoft TCP/IP stack that can crash the attacked computer.

RDS_Shell – A method of exploiting the Remote Data Services component of the Microsoft Data Access Components that lets a remote attacker run commands with system privileges.

Win Nuke – An exploit that can use NetBIOS to crash older Windows computers.

Viral Web Sites Threat

- Users can be enticed, often by email messages, to visit web sites that contain viruses or Trojans. These sites are known as viral web sites and are often made to look like well known web sites and can have similar web addresses to the sites they are imitating.
- Users who visit these sites often inadvertently download and run a virus or Trojan and can then become infected or the subject of hacker attacks.

Spyware, Adware and Advertising Trojans

- Spyware, Adware and Advertising Trojans are often installed with other programs, usually without your knowledge. They record your behaviors on the Internet, display targeted ads to you and can even download other malicious software on to your computer. They are often included within programs that you can download free from the Internet or that are on CDs given away free by magazines.
- Spyware doesn’t usually carry viruses but it can use your system resources and slow down your Internet connection with the display of ads. If the Spyware contains bugs (faults) it can make your computer unstable but the main concern is your privacy.
- These programs record every step that you take on the Internet and forward it to an Ad Management Centre which reviews your searches and downloads to determine your shopping preferences. The Ad Management Centre will build up a detailed profile of you, without your knowledge, and can pass this on to third parties, again without your knowledge.
Some Spyware can download more serious threats on to your computer, such as Trojan Horses.

Unsecured Wireless Access Points
If a wireless access point, e.g. an ADSL (Broadband) Router, hasn't been secured then anyone with a wireless device (laptop, PDA, etc) will be able to connect to it and thereby access the Internet and all the other computers on the wireless network.

Blue snarfing
The act of stealing personal data, specifically calendar and contact information, from a Bluetooth enabled device.

Social Engineering
Tricking computer users into revealing computer security or private information, e.g. passwords, email addresses, etc, by exploiting the natural tendency of a person to trust and/or by exploiting a person’s emotional response.

Spyware
Spyware is often secretly installed without users consent when a file is downloaded or a commercial pop-up is clicked. Spyware can reset your auto signature, monitor your keystrokes, scan, read and delete your files, access your applications and even reformat your hard drive. It constantly streams information back to the person that controls spyware.

Adware
This malware launches advertisements, mostly in the form of pop-ups. These are customized to you as a user, based on your behavior on the Internet, which may be monitored by spyware.

Spam
Spam may be defined as unwanted emails. Most users are exposed to scam, which is more than 50% of all Internet emails. Though spam is not a direct threat, it can be used to send different kinds of malware.

Scanning:
- Scanning of web systems are usually part of network or application fingerprinting prior to an attack.
- It also includes brute force and dictionary attacks on username, passwords and encryption keys.

Eavesdropping:
- Monitoring of data (on the network, or on user’s screens) may be used to uncover passwords or other sensitive data.
Out of band:
- Network attack techniques such as tunneling to access low level system functions can mean the target such as a router or server can be taken over.
- Once an attacker has control, this can be used to attack other assets required for the continued operation of a web site.
- The table shows the type of security threats faced in using the web. One way to group these threats is in terms of passive and active attacks.
- Passive attacks include eavesdropping on network traffic between browser and server and gaining access to information on a web site that is supposed to be restricted.
- Active attacks include impersonating another user, altering messages in transit between client and server, and altering information on a web site.

Table: A Comparison of Threats on the Web

<table>
<thead>
<tr>
<th>Threats</th>
<th>Consequences</th>
<th>Countermeasures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integrity</td>
<td>· Modification of user data</td>
<td>· Loss of information</td>
</tr>
<tr>
<td></td>
<td>· Trojan horse browser</td>
<td>· Compromise of machine</td>
</tr>
<tr>
<td></td>
<td>· Modification of memory</td>
<td>· Vulnerability to all other</td>
</tr>
<tr>
<td></td>
<td>· Modification of message traffic in transit</td>
<td>threats</td>
</tr>
<tr>
<td>Confidentiality</td>
<td>· Eavesdropping on the Net</td>
<td>· Loss of information</td>
</tr>
<tr>
<td></td>
<td>· Theft of info from server</td>
<td>· Loss of privacy</td>
</tr>
<tr>
<td></td>
<td>· Theft of data from client</td>
<td>Encryption, Web proxies</td>
</tr>
<tr>
<td></td>
<td>· Info about network configuration</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Info about which client talks to server</td>
<td></td>
</tr>
<tr>
<td>Denial of Service</td>
<td>· Killing of user threads</td>
<td>· Disruptive</td>
</tr>
<tr>
<td></td>
<td>· Flooding machine with bogus threats</td>
<td>· Annoying</td>
</tr>
<tr>
<td></td>
<td>· Filling up disk or memory</td>
<td>· Prevent user from getting work</td>
</tr>
<tr>
<td></td>
<td>· Isolating machine by DNS attacks</td>
<td>done</td>
</tr>
<tr>
<td>Authentication</td>
<td>· Impersonation of legitimate users</td>
<td>Cryptographic techniques</td>
</tr>
<tr>
<td></td>
<td>· Data forgery</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Misrepresentation of user</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Belief that false information is valid</td>
<td></td>
</tr>
</tbody>
</table>

- Table provides a summary of the types of security threats faced in using the Web. One way to group these threats is in terms of passive and active attacks. Passive attacks include eavesdropping on network traffic between browser and server and...
Chapter-6 : Application and Web Security

gaining access to information on a Web site that is supposed to be restricted. Active attacks include impersonating another user, altering messages in transit between client and server, and altering information on a Web site.

- Another way to classify Web security threats is in terms of the location of the threat: Web server, Web browser, and network traffic between browser and server. Issues of server and browser security fall into the category of computer system security; Part Four of this book addresses the issue of system security in general but is also applicable to Web system security.

How to Secure a System
How to Secure the System There are basic three methods to secure the system from online security attack.

Prevention:
If you were to secure your house, prevention would be similar to placing dead bolt locks on your doors, locking your window, and perhaps installing a chain link fence around your yard. You are doing everything possible to keep the threat out.

Detection:
You want to be sure you detect when such failures happen. Once again using the house analogy, this would be similar to putting a burglar alarm and motion sensors in the house. These alarms go off when someone breaks in. If prevention fails, you want to be alerted to that as soon as possible.

Reaction:
Detecting the failure has little value if you do not have the ability to respond. What good does it to be alerted to a burglar if nothing is done? If someone breaks into your house and triggers the burglar alarm, one hopes that the local police force can quickly respond. The same holds true for information security. Once you have detected a failure, you must execute an effective response to the incidence.

Preventing from Attack and Threats
- Recovering from Viruses, Worms, and Trojan Horses
- Avoiding Social Engineering and Networking Attacks
- Avoiding the Pitfalls of Online Trading
- Using Caution with USB Drives
- Securing Wireless Networks

Preventing from Email and communication
- Using Caution with Email Attachments
- Reducing Spam
- Using Caution With Digital Signatures
- Using Instant Messaging and Chat Rooms Safely
- Staying safe on social Network Sites.

Use Safe Browsing
- Evaluating Your Web Browser’s Security Settings
- Shopping Safely Online
Tips for securing the system attack

- Install and Use Anti-Virus Programs
- Use Care When Reading Email with Attachments
- Install and Use a Firewall Program
- Make Backups of Important Files and Folders
- Use Strong Passwords
- Use Care When Downloading and Installing Programs
- Install and Use a Hardware Firewall
- Install and Use a File Encryption Program and Access Controls
- Safeguard your Data
- Real-World Warnings keep you safe online.
- Keeping Children Safe Online

Web traffic security approaches

- The various approaches provide web security. They are similar in the services they provide and, to some extent, in the mechanism that they use.
- But they differ with respect to their scope of applicability and their relative location within the TCP/IP protocol stack.
- Figure illustrates this difference. One way to provide web security is to use IP Security.
- The advantage of using IPSec is that it is transparent to end users and applications and provides a general-purpose solution.
- Further, IPSec includes a filtering capability so that only selected traffic need incur the overhead of IPSec processing.

TCP/IP protocol stack

- Another relatively general-purpose solution is to implement security just above TCP Fig. The foremost example of this approach is the Secure Sockets Layer (SSL) and the follow-on internet standard known as Transport Layer Security (TLS).
- At this level, there are two implementation choices. For full generality, SSL (or TLS) could be provided as part of the underlying protocol suite and therefore be transparent to applications.
Alternatively, SSL can be embedded in specific packages.

**For Example** Netscape and Microsoft Explorer browsers come equipped with SSL, and most web servers have implemented the protocol.

Application-specific security services are embedded within the particular application. Fig. shows examples of this architecture.

The advantage of this approach is that the service can be tailored to the specific needs of a given application.

In the context of web security, an important example of this approach is Secure Electronic Transaction (SET).

---

**Secured Socket Layer**

Source: [https://www.cs.bham.ac.uk/~mdr/teaching/modules06/netsec/lectures/tls/tls.html](https://www.cs.bham.ac.uk/~mdr/teaching/modules06/netsec/lectures/tls/tls.html)

- SSL Stands for Secure Sockets Layer. SSL is a secure protocol developed for sending information securely over the Internet.
- Many websites use SSL for secure areas of their sites, such as user account pages and online checkout. Usually, when user is asked to “log in” on a website, the resulting page is secured by SSL.
- The Secure Sockets Layer (SSL) is a method for providing security for web based applications.
- SSL is standard security technologies for establishing an encrypted links between a server and a client—typically a web server (website) and a browser; or a mail server and a mail client.
- SSL allows sensitive information such as credit card numbers, social security numbers, and login credentials to be transmitted securely.
- Normally, data sent between browsers and web servers is sent in plain text leaving users vulnerable to eavesdropping. If an attacker is able to intercept all data being sent between a browser and a web server they can see and use that information.
- More specifically, SSL is a security protocol. Protocols describe how algorithms should be used; in this case, the SSL protocol determines variables of the encryption for both the link and the data being transmitted.
- SSL encrypts the data being transmitted so that a third party cannot eavesdrop on the transmission and view the data being transmitted. Only the user’s computer and the secure server are able to recognize the data.
- SSL keeps users name, address, and credit card information between user and merchant to which user are providing it. Without this kind of encryption, online shopping would be far too insecure to be practical.
- When user visit a Web addresses starting with “https,” the “s” after the “http” indicates the website is secure. These websites often use SSL certificates to verify their authenticity.
- While SSL is most commonly seen on the web (HTTP), it is also used to secure other internet protocols, such as Simple Mail Transfer Protocol (SMTP) for sending e-mail and Network News Transfer Protocol (NNTP) for newsgroups.
- Old implementations of SSL were limited to 40-bit encryption, but now most SSL secured protocols use 128-bit encryption or higher.
Chapter-6 : Application and Web Security

- SSL encrypts messages/attachments, but only in transport between SSL/TLS enabled mail servers. So, users SSL e-mail will be secure between any computer, laptop or smartphone and e-mail servers, but if the message travels outside the environment to unsecured (non-SSL) e-mail systems, user message is no longer secure and is not protected by SSL as shown in Fig.

**Objectives of SSL**

1) **Authenticating the client and server to each other**
   a. The SSL protocol supports the use of standard key cryptographic techniques (public key encryption) to authenticate the communicating parties to each other.
   b. Though the most frequent application consists in authenticating the service client on the basis of a certificate, SSL may also use the same methods to authenticate the client.

2) **Ensuring data integrity:**
   a. During a session, data cannot be either intentionally or unintentionally tampered with.

3) **Securing data privacy:**
   a. Data in transport between the client and the server must be protected from interception and be readable only by the intended recipient.

**Working of Secure Sockets Layer**

- Secure Sockets Layer (SSL) is a special security mechanism to encode communications between client (browser) and server (IIS) in order to prevent tampering and eavesdropping of the transmitted delicate data.
- Generally, the data that travels between browsers and a web server is sent in an un-encrypted format, which leads to data snooping vulnerability. If a hacker somehow manages to intercept traffic being sent between a web server and browser, they can easily reveal that information and exploit our private account.
- So, SSL ensures the privacy of sensitive data that travels across the wire by keeping it confidential or intact.
Chapter-6 : Application and Web Security

✓ It has always been a point of discussion and confliction on which OSI layer the SSL does indeed operate. Some references place the SSL protocol in the session layer, while others say that it works at the transport layer.

✓ SSL implementation requires a web server accompanied with SSL features and browser. SSL only imparts security for the connection (communication medium between server and client) but does not indeed offer security for the data once received.

✓ This means data is encrypted while being transmitted, but doesn’t secure data once they received. SSL uses public key encryption and provide data encoding, message integrity, server and client authentication.

✓ When a client accesses a website which typically may contain both secured and public segments, the secure portion mandates the user to be authenticated in some fashion. When client goes from a public page to the secure portion, the web server starts SSL and protects this type of communication.

✓ The server then sends a message back to the client, indicating a secure session is established and the client in response sends its security parameters. The server compares those values to its own until it finds a match.

✓ The server authenticates to the client by sending it a digital certificate, and if the client decides to trust the server, the process continues, otherwise terminated, and finally a secure communication medium is established.

✓ In some rare conditions, the server can require the client to send over a digital certificate for mutual authentication.

✓ The client generates a session key and encrypts it with the server’s public key and sends it across to the server, and now a genuinely secure channel is established, where both ends use that symmetric key to encrypt data back and forth.

✓ The following diagram fig. displays the comprehensive process of mutual authentication of both the server and client to each other, via secure socket layer.

✓ The asymmetric encryption is the basic building block of SSL where the public key is freely distributed to encrypt the message, which can only be decrypted by a corresponding private key.

✓ The Secure Socket Layer employs a third party organization, a Certificate Authority (CA), to identify one end or both ends of the transactions.

![SSL Communication Diagram]
Chapter-6 : Application and Web Security

Following are the number of steps that works on SSL:

- A browser sends a request to connect to the server and ask for a secure page (usually any document).
- The web server sends its public key with its signing certificate back to client.
- The browser checks whether the certificate was issued by a CA it trusts. The client compares the information in the certificate with the information received from the website and verifies all the details. If so, the browser shows the purity of server certificate by showing a green padlock, and client proceeds.
- The browser generates a random symmetric encryption key and then encrypts it to the public key of server. Finally, it sends it to the server, along with the encrypted URL and other encrypted FITTP data.
- The web server decrypts the incoming packet using its private key and uses the symmetric key to decrypt the URL and HTFP data that was generated randomly at client side.
- Then the requested document from the client along with other data is encrypted with the symmetric key and sent back to browser.
- Finally, the browser decrypts the packet using the symmetric key and secure handshaking is established.

Advantages

- Secure online banking transactions, hides credit card details.
- Secure system and website logins credentials along with other online exchanges of sensitive information.
- Secure transfer of files over HTTPS and FTP(s) services.
- To secure hosting control panel logins and activity like cPanel and Parallels.
- Secure private network (intranet) based data traffic such as file sharing and database connections.
- Secure network logins and other network traffic with SSL VPNs.
- Secure webmail and applications like Exchange server, Office Communications Server, and Outlook.
- Secure virtualization applications like cloud-based computing platforms.

Disadvantages

- It requires both parties to the communication to do extra work in exchanging handshakes and encrypting and decrypting the messages, making this form of communication slower than communication without SSL.
- To make SSL work, the server computer requires a file called a certificate. This file contains unique mathematical codes that identify the web host.
- A secure SSL session is slower than a standard, non-secure connection because the SSL software mathematically encodes all the data passing through it, so an eavesdropper monitoring the connection on the Internet sees only a jumble of random data.
- It increases network traffic and can reduce the speed of response of the runtime server.
Application of SSL
The applications of SSL are as follows:
- To secure online credit card transactions.
- To secure system logins and any sensitive information exchanged online.
- To secure webmail and applications like Outlook Web Access, Exchange and Office Communications Server.
- To secure workflow and virtualization applications like cloud-based computing platforms.
- To secure the connection between an e-mail client such as Microsoft Outlook and an e-mail server such as Microsoft Exchange.
- To secure the transfer of files over https and FTP(s) services such as website owners updating new pages to their websites or transferring large files.
- To secure intranet based traffic such as internal networks, file sharing, extranets, and database connections.
- To secure network logins and other network traffic with SSL, VPNS such as VPN Access Servers or Access Gateway.

SSL Architecture
1) SSL uses a cryptographic system that uses two keys to encrypt data - a public key known to everyone and a private or secret key known only to the recipient of the message.
2) Both Netscape Navigator and Internet Explorer support SSL, and many web sites use the protocol to obtain confidential user information, such as credit card numbers. By convention, URLs that require an SSL Connection start with https.
3) Technically, SSL is a transparent protocol which requires little interaction from the end user when establishing a secure session.
4) SSL is designed to make use of TCP to provide a reliable end-to-end secure service. SSL is not a single protocol but rather two layers of protocols, as illustrated in fig.
5) It can be seen that one layer makes use of TCP directly. This layer is known as the SSL Record Protocol and it provides basic security services to various higher layer protocols.
6) An independent protocol that makes use of the record protocol is the Hypertext Markup Language (HTTP) protocol. Another three higher level protocols that also make use of this layer are part of the SSL stack.
7) The SSL Record Protocol provides basic security services to various higher-layer protocols. In particular, the Hypertext Transfer Protocol (HTTP), which provides the transfer service for Web client/server interaction, can operate on top of SSL.

8) Three higher-layer protocols are defined as part of SSL:
   (i) The Handshake Protocol.
   (iii) The Alert Protocol.

   These SSL-specific protocols are used in the management of SSL exchanges and two important SSL concepts are the SSL session and the SSL connection, which are defined in the specification as follows:

Connection
A connection is a transport (in the OSI layering model definition) that provides a suitable type of service. For SSL, such connections are peer-to-peer relationships. The connections are transient. Every connection is associated with one session.

Session
- An SSL session is an association between a client and a server. Sessions are created by the Handshake Protocol.
- Sessions define a set of cryptographic security parameters, which can be shared among multiple connections. Sessions are used to avoid the expensive negotiation of new security parameters for each connection.
- Between any pair of parties (applications such as HTTP on client and server), there may be multiple secure connections.
- There are actually a number of states associated with each session. Once a session is established, there is a current operating state for both read and write (i.e., receive and send).
- In addition, during the Handshake Protocol, pending read and write states are created. Upon successful conclusion of the Handshake Protocol, the pending states become the current states.

SSL record Protocol Layer
The SSL Record Protocol provides two services for SSL connections:

Confidentiality:
The Handshake Protocol defines a shared secret key that is used for conventional encryption of SSL payloads.

Message Integrity
The Handshake Protocol also defines a shared secret key that is used to form a message authentication code (MAC).

Fig. indicates the overall operation of the SSL Record Protocol. The Record Protocol takes an application message to be transmitted, fragments the data into manageable blocks, optionally compresses the data, applies a MAC, encrypts, adds a header, and transmits the resulting unit in a TCP segment. Received data are decrypted, verified, decompressed, and reassembled and then delivered to higher-level users.
In order to operate on data the protocol performs the following actions:

**Fragmentation:**

The first step is fragmentation. It takes an application message (upper-layer message) to be transmitted and fragments it into manageable blocks. These blocks are $2^{14}$ bytes = 16, 384 bytes or less.

**Compression:**

These blocks are then optionally compressed which must be lossless and may not increase the content length by more than 1024 bytes. In SSLv3 (as well as the current version of TLS), no compression algorithm is specified, so the default compression algorithm is null.

**Message Authentication Code:**

A message authentication code is then computed over the compressed data using a shared secret key. This is then appended to the compressed (or plaintext) block. For this purpose, a shared secret key is used.

**Encryption:**

The compressed message plus MAC are then encrypted using symmetric encryption. Encryption may not increase the content length by more than 1024 bytes, so that the total length may not exceed $2^{12}$ bytes + 2048. A number of different encryption algorithms are permitted.

**Append Header:**

The final step is to append a header, consisting of the following fields. The overall format is shown in fig.:
Content type (8 bits):
The higher layer protocol used to process the enclosed fragment.

Major Version (8 bits): Indicates major version of SSL in use. For SSLv3, the value is 3.

Minor Version (8 bits)
Indicates minor version in use. For SSLv3, the value is 0.

Compressed Length (16 bits)
- The length in bytes of the compressed (or plaintext) fragment.
- The “content type” above is one of four types; the three higher level protocols given above that make use of the SSL record, and a fourth known as “application data”.

SSL handshake Protocol
The SSL Handshake protocol layer defines three higher-layer protocols are defined as part of SSL:
- The Handshake Protocol.
- The Alert Protocol.

Fig. shows the SSL Record Protocol Payload which is used by three higher layer protocols.

The handshake Protocol
Handshake Protocol is the most complex part of SSL. It allows the server and client to authenticate each other and to negotiate an encryption and MAC algorithm and cryptographic keys to be used to protect data sent in an SSL record.
This protocol is used before any application data is sent. It consists series of messages exchanged by the client and server, all of which have the format shown in Fig. Each message has three fields:

Type (1 byte)
Indicates one of 10 messages. Fig lists the defined message types.

Length (3 bytes)
The length of the message in bytes.
Chapter-6 : Application and Web Security

The initial exchange needed to establish a logical connection between client and server is shown in Fig. The exchange can be viewed as having four phases.

**Phase 1 - Establish Security Capabilities:**
- This phase is used to initiate a logical connection and to establish the security capabilities that will be associated with it.
- The exchange is initiated by the client, who sends a client_hello message with the parameters such as Version, Random, Session ID, Cipher Suite, and Compression Method.

**Phase 2 - Server Authentication and Key Exchange:**
- The server begins this phase by sending its certificate, if it needs to be authenticated; the message contains one or a chain of X.509 certificates.
- The certificate message is required for any agreed-on key exchange method except anonymous Diffie-Hellman.

**Phase 3 - Client Authentication and Key Exchange:**
- Upon receipt of the server_done message, the client should verify that the server provided a valid certificate if required and check that the server_hello parameters are acceptable.
- If all is satisfactory, the client sends one or more messages back to the server. If the server has requested a certificate, the client begins this phase by sending a certificate message. If no suitable certificate is available, the client sends a no certificate alert instead.

**Phase 4 - Finish**
- This phase completes the setting up of a secure connection. The client sends a change_cipher_spec message and copies the pending CipherSpec into the current CipherSpec.
- The client then immediately sends the finished message under the new algorithms, keys, and secrets.
Chapter-6 : Application and Web Security

- The finished message verifies that the key exchange and authentication processes were successful.

The Change Cipher Spec Protocol:
- The Change Cipher Spec Protocol is one of the three SSL-specific protocols that use the SSL Record Protocol, and it is the simplest.
- This protocol consists of a single message which consists of a single byte with the value 1 is shown in Fig.
- The sole purpose of this message is to cause the pending state to be copied into the current state, which updates the cipher suite to be used on this connection.

The Alert Protocol:
- This protocol is used to convey SSL-related alerts to the peer entity.
- As with other applications that use SSL, alert messages are compressed and encrypted, as specified by the current state.
- It consists of two bytes the first of which takes the values 1 (warning) or 2 (fatal).
- If the level is fatal SSL immediately terminates the connection. The second byte contains a code that indicates the specific alert.
- Following are the alerts that are always fatal (definitions from the SSL specification)
  unexpected_message
  An inappropriate message was received.
  bad_record_mac:
  An incorrect MAC was received.
  decompression_failure
  The decompression function received improper input (e.g. unable to decompress or decompress to greater than maximum allowable length).
  handshake_failure:
  Sender was unable to negotiate an acceptable set of security parameters given the option available.
  illegal_parameter
  A field in a handshake message was out of range or inconsistent with other fields.
  The remainder of the alerts are the following:
  close_notify:
  Notifies the recipient that the sender will not send any more messages on this connection. Each party is required to send a close_notii alert before closing the right side of a connection.
  no_certificate:
  May be sent in response to a certificate request if no appropriate certificate is available.
  bad_certificate
unsupported_certificate
The type of the received certificate is not supported.

certificate_revoked
A certificate has been revoked by its signer.

certificate_expired
A certificate has expired.

certificate_unknown
Some other unspecified issue arose in processing the certificate, rendering it unacceptable.

Transport Layer Security
- Transport Layer Security (TLS) protocol has been developed by the Internet Engineering Task Force (IETF) as the standard protocol for providing security services in the context of E-commerce over the Internet.
- TLS is an IETF standardization initiative whose goal is to produce an Internet standard version of SSL.
- TLS is defined as a Proposed Internet Standard in RFC 2246. RFC 2246 is very similar to SSLv3.
- Transport Layer Security (TLS) is a protocol that ensures privacy between communicating applications and their users on the internet.
- When a server and client communicate, TLS ensures that no third party may eavesdrop or tamper with any message. TLS is the successor to the Secure Sockets Layer (SSL).
- Transport Layer Security, a protocol that guarantees privacy and data integrity between client/server applications communicating over the Internet.
- TLS is application protocol-independent. Higher-level protocols can layer on top of the TLS protocol transparently shown in Fig.

![TLS Protocol Diagram]

- The TLS Record Layer is used for encapsulation of various higher level protocols such as the handshake protocol, the alert protocol, the change cipher spec protocol, and the application data protocol.

Notes Prepared by – Prof. Manoj Kavedia (9860174297, 9324258878)
The TLS Handshake Layer consists of the handshake protocol, the alert protocol and the change cipher spec protocol.

The HTTP1 layer and TLS Record layer combined constitutes Hypertext Transfer Protocol Secure (HTTPS) as shown in Fig. If HTTP1 layer is replaced with FTP, will be the File Transfer Protocol Secure (FTPS). Of course, HTTP2 is the plain HTTP without any security support.

**TLS is composed of two layers, they are:**
(i) The TLS Record Protocol and
(ii) The TLS Handshake Protocol

**The TLS Record Protocol**
- The TLS Record Protocol provides connection security with some encryption method such as the Data Encryption Standard (DES).
- The TLS Record Protocol can also be used without encryption.
- It is layered on top of a reliable transport protocol, such as TCP, it ensures that the connection is private by using symmetric data encryption and it ensures that the connection is reliable.
- It is also used for encapsulation of higher-level protocols, such as the TLS.
- The record protocol is responsible for shifting data between the two ends of the link using the parameters agreed via the handshake protocol.

**The TLS Handshake Protocol:**
The TLS Handshake Protocol allows the server and client to authenticate each other and to negotiate an encryption algorithm and cryptographic keys before data is exchanged.

As shown in fig. that how the TLS handshake protocol also uses the record protocol to send its messages during the handshake phase.

This seems counter intuitive because the handshake protocol is used to negotiate the parameters of the record protocol layer over which it is communicating.

TLS is design to handle this bootstrap process; in its initial state, the record protocol just forwards data without any encryption or compression.

The record protocol layer operates according to a group of settings or parameters called a connection state. The connection state should be thought of as the configuration settings for the layer. It includes things like which encryption algorithm is in use and what are the encryption keys.
Objectives of TLS:

- To provide confidentiality, (sometimes referred to as privacy), data integrity, identification, and authentication using digital certificates.
- To securely encrypt data being exchanged.
- To guarantee the identity of one or both parties.
- To prevent data tampering.
- To prevent replay attacks.
- Enabling interoperability between applications.
- Providing an extensible framework that can readily incorporate new public key and bulk encryption methods.
- Ensuring relative computational efficiency.

Working of TLS

- Transport Layer Security (TLS) is an application layer cryptographic protocol. The main purpose of TLS and SSL (Secure Sockets Layer; predecessor of TLS) is to ensure privacy between the applications and the users communicating over internet.
- When the server and client communicate, TLS provides communication security and ensures that no malicious party can tamper with the message-packets and also protects against serious threats like eavesdropping and message-forgery.
- The protocol has several versions that are widely internet.
- TLS is widely recognized and is also a standard protocol issued by IETF (Internet Engineering Task Force) for communication of emails securely over the Internet and also creates a secured used in applications such as e-mail, internet faxing, voice-over-IP (VOIP) for protecting sensitive data that are transmitted over the environment for applications like web-browsing, c-mails and other client-server applications.

- Transport Layer Security or TLS is a protocol used in application layer that ensures message security over the internet; it uses a combination of asymmetric cryptography, symmetric encryption and message authentication codes for key exchange, privacy and message integrity respectively to encrypt the network connections above Transport layer.
- TLS is especially designed to provide protection against eavesdropping, message-forging, message tampering and protect confidentiality and data integrity by encrypting the data transmission of applications like e-mail between client and server.
Chapter-6 : Application and Web Security

To indicate the server that the client is using a TLS connection can be achieved by two ways:

(i) By using different port numbers for TLS connections.
(ii) By using general port numbers where the client requests the server to switch the connection to TLS using any protocol-specific mechanism.

- After both the client and server have agreed on a TLS connection they use a handshaking mechanism in which they agree on several parameters to establish a secure connection and negotiate a stateful connection.
- Finally when TLS is established on both the ends information is exchanged by encrypting the plain text to ensure data confidentiality.
- The client acknowledges the receipt of the certificate. The client then encrypts a random number which can only be decrypted by the server’s private key and then the session keys are generated.
- The connection is established only after all the steps are successfully done; failing in which results in connection failure.
- TLS is based on specifications developed by Netscape Communications’ SSL protocol, which is the predecessor of TLS. TLS and SSL are not interoperable, i.e. TLS cannot be implemented as SSL.
- Since transferring unencrypted data increases the risk of threats of message tampering and alteration, in some organizations that deals with storing of confidential data and sensitive messages, implementing TLS is not only a good idea but instead a mandated option.

Working of TLS protocol

- The TLS protocol allows a client-server application to communicate securely across an untrusted network (such as the Internet), in a way designed to prevent eavesdropping (stealing information by reading the communication between the sender and receiver) and tampering (changing the transmitted information).
- It uses asymmetric cryptography for securely exchanging keys between the client and the server, and then uses symmetric cryptography for the actual encryption of secret data being transmitted. Message Authentication Codes (MACs) are used for message integrity.
- Once the client has requested to use TLS for communication, a series of “records” are exchanged between the client and the server.
- This is called the TLS handshake and is used to establish various parameters which are used for the security of the connection as shown in fig

A simple example of a TLS handshake which is presented below:

✓ The TLS handshake starts with a negotiation phase. The client sends a “ClientHello” message to the server. This message contains certain parameters, such as the highest TLS version supported by the client, a list of supported cipher suites, and a list of compression methods supported by the client. Newer clients also include the expected server name, to enable virtual hosting of multiple services on the same IP address.
✓ The server then sends a “ServerHello” message, containing the chosen cipher suite, compression method and TLS version, along with a randomly generated number. The server also sends its certificate to the client. The
Chapter-6 : Application and Web Security

The server then sends a Server Hello Done message indicating that it is done with the initial handshake.

- The client generates a PreMasterSecret, which is encrypted using the public key of the server certificate. It sends this along with a Client Key Exchange record to server.

- Finally, the client and the server use the previously exchanged random number and the Pre Master Secret to generate a common secret called the "Master Secret". All other key data is generated by passing this "Master Secret" through Pseudo-Random Number Generators (PRNGs). At this stage the Negotiation phase is complete.

- The Client now sends a Change Cipher Spec record to the server, telling it that from now onwards all communication will be encrypted. Finally, the client sends a finished message to the server, which is encrypted and authenticated.

- The Server decrypts this message using the previously exchanged secrets. (The hash and the MAC.) If it fails to decrypt/jverify the message, the connection is turned down.

- The Server sends a Change Cipher Spec record to the client and everything described above happens in the reverse direction. At this stage the TLS handshake is complete.

The TLS handshake described above established session keys. These keys are used to encrypt/decrypt data sent between the client and the server using symmetric encryption.

Advantages

- It is relatively simple, well-understood, standard technology.
- It applies to both a message body and its attachments.

| Cipher suites (previously agreed upon during the handshake phase) are used to encrypt the data records.
| Once the data transmission is complete, this secure channel is turned down and all the cryptographic key material is discarded.

Notes Prepared by – Prof. Manoj Kavedia (9860174297, 9324258878)
Chapter-6 : Application and Web Security

✔ TLS is easy to use. Probably the most used security on the internet.
✔ TLS do not need any operating system support.
✔ When messages are exchanged over the internet, they are checked while transmitting from one computer to another. This feature offers reliability of the web based communication.
✔ TLS protocol stops unauthorized user access from interfering as a third party in the middle of a communication on the internet. The third party will only take part in the communication when it has been noticed by two authorized users.
✔ TLS is in use by most web browsers.
✔ It is widely recognized as the secure HTTP (HTTPS) Protocol.

Disadvantages

✔ It is tightly coupled with the transport-layer protocol.
✔ It represents an all-or-nothing approach to security. This implies that the security mechanism is unaware of message contents, so that you cannot selectively apply security to portions of the message as you can with message-layer security.
✔ Protection is transient. The message is protected only while in transit. Protection is removed automatically by the endpoint when it receives the message.
✔ It is not an end-to-end solution, simply point-to-point.
✔ TLS often mistake firewalls as man in the middle attack.
✔ It is exposed to clogging over TCP.

Application

✔ The Transport Layer Security (TLS) protocol enables two parties to communicate with privacy and data integrity. The TLS protocol evolved from the SSL 3.0 protocol but TLS and SSL do not interoperate.
✔ The TLS protocol provides communications security over the internet.
✔ It allows client/server applications to communicate in a way that is private and reliable.

Problems

• In SSL and TLS, why is there a separate Change Cipher Spec Protocol, rather than including a change_cipher_spec message in the Handshake Protocol?
• Consider the following threats to Web security and describe how each is countered by a particular feature of SSL.
  1. **Brute-Force Cryptanalytic Attack:** An exhaustive search of the key space for a conventional encryption algorithm.
  2. **Known-Plaintext Dictionary Attack:** Many messages will contain predictable plaintext, such as the HTTP GET command. An attacker constructs a dictionary containing every possible encryption of the known-plaintext message. When an encrypted message is intercepted, the attacker takes the portion containing the encrypted known plaintext and looks up the ciphertext in the dictionary. The ciphertext should match against an entry that was encrypted with the same secret key. If there are several matches, each of these can be tried against the full
Chapter-6 : Application and Web Security

ciphertext to determine the right one. This attack is especially effective against small key sizes (e.g., 40-bit keys).
3. **Replay Attack**: Earlier SSL handshake messages are replayed.
4. **Man-in-the-Middle Attack**: An attacker interposes during key exchange, acting as the client to the server and as the server to the client.
5. **Password Sniffing**: Passwords in HTTP or other application traffic are eavesdropped.
6. **IP Spoofing**: Uses forged IP addresses to fool a host into accepting bogus data.
7. **IP Hijacking**: An active, authenticated connection between two hosts is disrupted and the attacker takes the place of one of the hosts.
8. **SYN Flooding**: An attacker sends TCP SYN messages to request a connection but does not respond to the final message to establish the connection fully. The attacked TCP module typically leaves the "half-open connection" around for a few minutes. Repeated SYN messages can clog the TCP module.

---

**Set electronics Transaction (SET)**

- SET is an open encryption and security specification designed to protect credit card transactions on the internet. The current version, SETv1, emerged from a call for security standards by MasterCard and Visa in February 1996.
- SET (Secure Electronic Transaction) is a very comprehensive security protocol, which utilizes cryptography to provide confidentiality of information, ensure payment integrity, and enable identity authentication.
- For authentication purposes, cardholders, merchants, and acquirers will be issued digital certificates by their sponsoring organizations.
- Secure Electronic Transactions (SET) relies on the science of cryptography - the encoding and decoding messages. There are two primary encryption methods in use today such as secret-key cryptography and public-key cryptography.
- A wide range of companies were involved in developing the initial specification, including IBM, Microsoft, Netscape, RSA, Terisa, and Verisign. Beginning in 1996.
- SET is not itself a payment system. Rather it is a set of security protocols and formats that enables users to employ the existing credit card payment infrastructure on an open network, such as the internet, in a secure fashion.
- Fig. depicts that SET provides security functions at the highest (application) layer of the protocol stack. SET is an application and security its functions are not available to other applications.
- The integrity of SET relies on the ability to resolve identities to a particular individual, merchant or payment gateway.

---

![Diagram of S/MIME, PGP, SET, HTTP, FTP, SMTP, TCP, IP]
In essence, SET provides three services:
- Provides a secure communications channel among all parties involved in a transaction.
- Provides trust by the use of X.509v3 digital certificates.
- Ensures privacy because the information is only available to parties in a transaction when and where necessary.

Requirement of SET
The SET specification lists the following business requirements for secure payment processing with credit cards over the Internet and other networks:
- **Provide confidentiality of payment and ordering information:**
  - It is necessary to assure cardholders that this information is safe and accessible only to the intended recipient.
  - Confidentiality also reduces the risk of fraud by either party to the transaction or by malicious third parties.
  - SET uses encryption to provide confidentiality.

- **Ensure the integrity of all transmitted data:**
  - That is, ensure that no changes in content occur during transmission of SET messages.
  - Digital signatures are used to provide integrity.

- **Provide authentication that a cardholder is a legitimate user of a credit card account:**
  - A mechanism that links a cardholder to a specific account number reduces the incidence of fraud and the overall cost of payment processing.
  - Digital signatures and certificates are used to verify that a cardholder is a legitimate user of a valid account.

- **Provide authentication that a merchant can accept credit card transactions through its relationship with a financial institution:**
  - This is the complement to the preceding requirement.
  - Cardholders need to be able to identify merchants with whom they can conduct secure transactions.
  - Again, digital signatures and certificates are used.

- **Ensure the use of the best security practices and system design techniques to protect all legitimate parties in an electronic commerce transaction:**
  - SET is a well-tested specification based on highly secure cryptographic algorithms and protocols.

- **Create a protocol that neither depends on transport security mechanisms nor prevents their use:**
  - SET can securely operate over a raw TCP/IP stack.
    - However, SET does not interfere with the use of other security mechanisms, such as IPSec and SSL/TLS.

- **Facilitate and encourage interoperability among software and network providers:**
  - The SET protocols and formats are independent of hardware platform, operating system, and web software.
Key Feature of Set

To meet the requirements, SET incorporates the following features

Confidentiality of information:
- Cardholder account and payment information is secured as it travels across the network.
- An interesting and important feature of SET is that it prevents the merchant from learning the cardholder’s credit card number; this is only provided to the issuing bank.
- Conventional encryption by DES is used to provide confidentiality.

Integrity of data:
- Payment information sent from cardholders to merchants includes order information, personal data, and payment instructions.
- SET guarantees that these message contents are not altered in transit. RSA digital signatures, using SHA-1 hash codes, provide message integrity.
- Certain messages are also protected by HMAC using SHA-1.

Cardholder account authentication:
- SET enables merchants to verify that a cardholder is a legitimate user of a valid card account number.
- SET uses X.509v3 digital certificates with RSA signatures for this purpose.

Merchant authentication:
- SET enables cardholders to verify that a merchant has a relationship with a financial institution allowing it to accept payment cards.
- SET uses X.509v3 digital certificates with RSA signatures for this purpose.

Set Participants
Figure indicates the participants in the SET system, which include the following:
Cardholder:
- In the electronic environment, consumers and corporate purchasers interact with merchants from personal computers over the internet.
- A cardholder is an authorized holder of a payment card (e.g., MasterCard, Visa) that has been issued by an issuer.

Merchant:
- A merchant is a person or organization that has goods or services to sell to the cardholder.
- Typically, these goods and services are offered via a website or by electronic mail.
- A merchant that accepts payment cards must have a relationship with an acquirer.

Issuer
- This is a financial institution, such as a bank, that provides the cardholder with the payment card.
- Typically, accounts are applied for and opened by mail or in person.
- Ultimately, it is the issuer that is responsible for the payment of the debt of the cardholder.

Acquirer
- This is a financial institution that establishes an account with a merchant and processes payment card authorizations and payments.
- Merchants will usually accept more than one credit card brand but do not want to deal with multiple bankcard associations or with multiple individual issuers.
- The acquirer provides authorization to the merchant that a given card accounts is active and that the proposed purchase does not exceed the credit limit.
- The acquirer also provides electronic transfer of payments to the merchant’s account. Subsequently, the acquirer is reimbursed by the issuer over some sort of payment network for electronic funds transfer.

Payment gateway:
- This is a function operated by the acquirer or a designated third party that processes merchant payment messages.
- The payment gateway interfaces between SET and the existing bankcard payment networks for authorization and payment functions.
The merchant exchanges SET messages with the payment gateway over the internet, while the payment gateway has some direct or network connection to the acquirers financial processing system.

Certification authority (CA):
- This is an entity that is trusted to issue X.509v3 public-key certificates for cardholders, merchants, and payment gateways.
- The success of SET will depend on the existence of a CA infrastructure available for this purpose.

Working of SET
Some sequences of events are required for a transaction. Fig depicts the working of SET.

The customer opens an account:
- The customer obtains a credit card account, such as MasterCard or Visa, with a bank that supports electronic payment and SET.

The customer receives a Certificate
- After suitable verification of identity, the customer receives an X.509v3 digital certificate, which is signed by the bank.
- The certificate verifies the customer’s RSA public key and its expiration date.
- It also establishes a relationship, guaranteed by the bank, between the customer’s key pair and his or her credit card.

Merchants have their own certificates:
- A merchant who accepts a certain brand of card must be in possession of two certificates for two public keys owned by the merchant: one for signing messages, and one for key exchange.
- The merchant also needs a copy of the payment gateway’s public-key certificate.

The customer places an order:
- This is a process that may involve the customer first browsing through the merchant’s website to select items and determine the price.
- The customer then sends a list of the items to be purchased to the merchant, who returns an order form containing the list of items, their price, a total price, and an order number.

The merchant is verified:
In addition to the order form, the merchant sends a copy of its certificate, so that the customer can verify that he or she is dealing with a valid store.

The order and payment are sent:
- The customer sends both order and payment information to the merchant, along with the customer’s certificate.
- The order confirms the purchase of the items in the order form. The payment contains credit card details.
- The payment information is encrypted in such a way that it cannot be read by the merchant. The customer’s certificate enables the merchant to verify the customer.

The merchant requests payment authorization:
- The merchant sends the payment information to the payment gateway, requesting authorization that the customer’s available credit is sufficient for this purchase.
The merchant confirms the order:
- The merchant sends confirmation of the order to the customer.

The merchant provides the goods or service:
- The merchant ships the goods or provides the service to the customer.

The merchant requests payment:
This request is sent to the payment gateway, which handles all of the payment processing.

Practice Questions
Q.1. State the need of application hardening.
Q.2. What are the goals and benefits of application hardening?
Q.3. State the advantages and disadvantages of web server.
Q.4. Enlist the features of active directory.
Q.5. What are the advantages and disadvantages of active directory?
Q.6. List and explain types of threats to web system.
Q.7. Explain SSL/TLS with neat diagram.
Q.8. Explain the concept of Secure Socket Layer. List the objectives of SSL.
Q.9. Describe the working of SSL.
Q.10. State the advantages, disadvantages and application of SSL.
Q.13. Explain the working concept of handshake protocol.
Q.15. Explain working of TLS.
Q.17. State the advantages, disadvantages and application of TLS.
Q.18. What is Secure Electronic Transaction?
Q.19. Explain the key feature of SET.
Q.20. Describe the working principle of SET.
### MSBTE Important Questions

**Summer-2008 , Winter 2008**

| Q1 | Describe SSL with its objectives at which layer of OSI it works. Draw and explain in brief SSL protocol stack |

**Summer 2009**

| Q2 | Name the three sub protocols of SSL. Describe each in brief. |
| Q3 | What are the key participants in Secure Electronic Transaction? |
| Q4 | Explain following terms with example: Hotfix, patch, service pack |

**Winter 2009**

| Q5 | Explain hotfix, services pack and why they are important. |
| Q6 | Explain the architecture of SSL (Secure Socket Layer). |
| Q7 | How does SET (Secure Electronic Transaction) protect the payment information from the merchant? |
| Q8 | Discuss different threats of web security. |

**Summer 2010**

| Q9 | What is application hardening? |
| Q10 | Explain SSL in detail. |
| Q11 | List and describe different ways of web security. |

**Winter 2010**

| Q12 | Name three Sub protocol of the secure socket Layer? Describe them in Brief? |
| Q13 | What are the different components of secure electronic transaction? |

**Summer-11**

| Q14 | Explain web security threats and web traffic security approaches. |
| Q15 | Draw SSL architecture and describe how to secure socket layer. |

**Winter-11**

| Q16 | List and describe different ways of web security. |
| Q17 | Explain secure socket layer with its two objectives. At which layer of OSI it works? |
| Q18 | Draw and explain SSL protocol stack. |
| Q19 | What are the key participants in SET? Explain each in brief. |
| Q20 | Explain the terms: hotfix, patch, service pack. |

**Summer-12**

| Q21 | Describe SSL protocol in web security. |
| Q22 | Describe how secure electronic transaction works. |
| Q23 | Explain hotfix and patch. |

**Winter-12**

| Q24 | |
Chapter-6 : Application and Web Security

Q24) Discuss different possible threats to web security and their effects.

Summer-13
Q26) Describe the architecture of SSL and list its protocols.  
Q27) Explain Secure Electronic Transactions (SET) with its requirements and participants.

Winter-13
Q28) Explain web server and active directory.  
Q29) Describe the major transaction supported by SET.  
Q30) Describe the working of SSL.  
Q31) What are the key participants in SET?  
Q32) Explain the term hotfix, patch and service pack.

Summer-14
Q33) State secure electronic transaction and its purpose. Summarize various participants in SET and their roles.  
Q34) Describe SSL protocol in web security.  
Q35) Enlist threats to web-security. Describe any three of them in detail.  
Q36) State meaning of following terms.  
(I) Hot fix.  
(II) Update.  
(III) Patch.  
(iv) Service pack.

End-Chapter-6