SEGUNDA UNIDAD DE APRENDIZAJE
SEGURIDAD EN LA WEB

2.1 MODELO OSI

El modelo de referencia OSI es el modelo principal para las comunicaciones por red, todos los fabricantes de redes relacionan sus productos con el modelo de referencia OSI, sobre todo en la capacitación de sus usuarios. Este modelo permite entender las funciones de red que se producen en cada capa y permite entender cómo se mueve la información a través de una red, dividida en muchos paquetes de datos, generada en los programas de aplicación (por ej., hojas de cálculo, documentos, etc.), a través de un medio de red (por ej., cables, etc.), hasta otro programa de aplicación ubicado en otro computador de la red, aun cuando el transmisor y el receptor tengan distintos tipos de medios de red.

El modelo de referencia OSI, está dividido en siete capas numeradas, cada una soportando una función de red distinta, lo que se denomina división en capas. Esta división aporta las siguientes ventajas:

- Divide la comunicación de red en partes más pequeñas, independientes y especializadas.
- Normaliza los componentes de red para permitir el desarrollo y el soporte de los productos de diferentes fabricantes.
- Permite a los distintos tipos de hardware y software de red comunicarse entre sí (interoperatividad).
- Impide que los cambios en una capa puedan afectar las demás capas, para que se puedan desarrollar cada una de ellas con independencia y mayor rapidez.
- Divide la comunicación de red en partes más pequeñas claramente identificable.

El modelo de interconexión de sistemas abiertos (OSI) tiene siete capas, desde la más baja en la jerarquía (la física) y hacia la más alta (la aplicación). Las capas se apilan de esta forma:

Figura 2.1 Modelo OSI

El propósito de cada capa es proveer los servicios para la capa superior, haciendo invisible a esta capa superior los detalles de cómo los servicios son implementados (de manera vertical funciona bajo un modelo de tipo cliente/servidor). Las capas son abstraídas físicamente, de tal manera que cada capa cree (de manera lógica) que se está comunicando con la capa equivalente en la otra computadora, usando las reglas definidas por un protocolo específico, cuando realmente cada capa se comunica sólo con las capas adyacentes de la misma computadora.

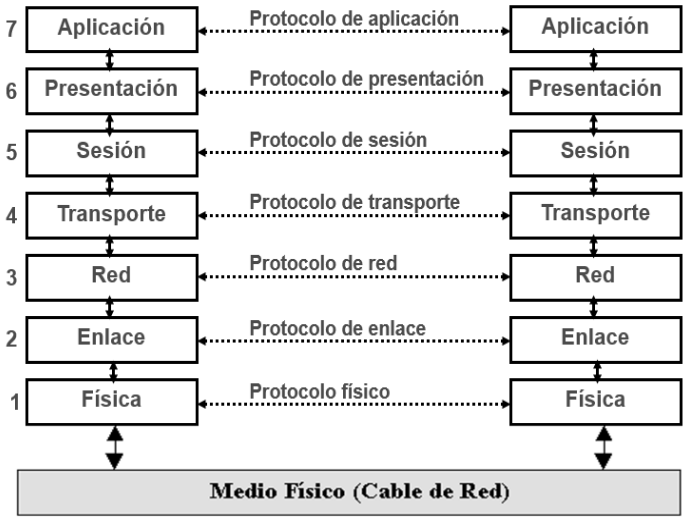


Figura 2.2 Modelo físico/lógico de comunicación entre dos hosts

A excepción de la capa más baja del modelo OSI, ninguna capa puede pasar información directamente a su equivalente en la otra computadora. La información que envía una computadora debe de pasar por todas las capas inferiores, hasta llegar al nivel físico, la información entonces se mueve a través del cable de red hacia la computadora que recibe y hacia arriba a través de las capas de esta hasta que llega al mismo nivel de la capa que envió la información. La interacción entre las capas adyacentes se llama **interface**, esta define qué servicios de la capa inferior son ofertados a la capa superior y como esos servicios son accesados. La serie de las reglas que se usan para la comunicación entre las capas se llama **protocolo**.

2.1.1 Funciones de los niveles del modelo OSI

Aunque se verán a detalle de manera individual, la función general de cada capa se indica a continuación

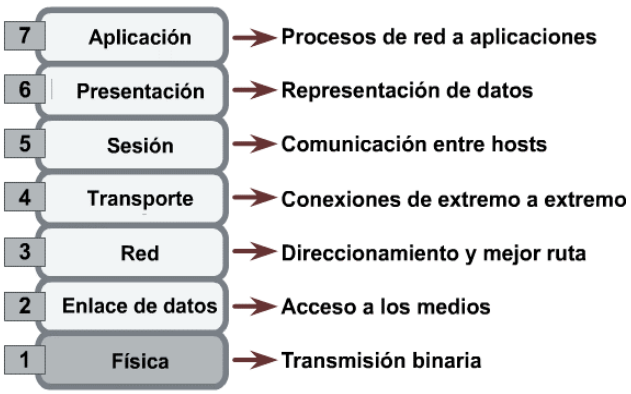


Figura 2.3 La función por cada capa

- *Capa de aplicación*: define como el usuario accede a la red, para una transferencia de archivos, login remoto, correo electrónico, consulta a bases de datos, etc.

- *Capa de presentación*: establece una sintaxis y semántica de la información transmitida, define la estructura de los datos a transmitir (campos de un registro: nombre, dirección, teléfono, etc). Define el código a usar para representar una cadena de caracteres (ASCII,

EBCDIC, etc), y las funciones asociadas a la compresión de datos y la seguridad (criptografía).

- *Capa de sesión*: permite a usuarios en diferentes máquinas establecer una sesión. Una sesión puede ser usada para efectuar un login a un sistema de tiempo compartido remoto, para transferir un archivo entre 2 máquinas, etc. Controla el diálogo (quién habla, cuándo, cuánto tiempo, half duplex o full duplex) y se encarga de la función de sincronización en la comunicación.

- *Capa de transporte*: establece conexiones punto a punto sin errores para el envío de mensajes. Permite multiplexar una conexión punto a punto entre diferentes procesos del usuario. Provee la función de difusión de mensajes (broadcast) a múltiples destinos y el control de flujo.

- *Capa de red*: divide los mensajes de la capa de transporte en paquetes y los ensambla. Utiliza el nivel de enlace para el envío o de paquetes, previo enrutamiento de paquetes. Para el envío de paquetes lo hace de nodo a nodo usando ya sea un circuito virtual o como datagramas. Finalmente se ocupa del control de la congestión.

- *Capa de enlace de datos*: estructura el flujo de bits bajo un formato predefinido llamado trama, para lo que agrega una secuencia especial de bits al principio y al final del flujo inicial de bits. Transfiere tramas de una forma confiable libre de errores (utiliza reconocimientos y retransmisión de tramas).

- *Capa física*: se encarga de la transmisión de flujo de bits a través del medio, por lo que se encarga de manejar las señales eléctricas, específica cables, conectores y componentes de interfaz con el medio de transmisión.

2.1.2 Funcionamiento del Modelo

Se denominan entidades a los elementos activos (hardware o software) que se hallan en cada una de las capas. Si las entidades residen en la misma capa de dos computadores, se les llama entidades pares. Si un computador (host A) debe enviar datos a otro computador (host B), los datos deben empaquetarse y ser preparados antes de transmitirse, por lo que se realiza un proceso denominado **encapsulamiento**, los datos se desplazan a través de las capas del modelo OSI, recibiendo encabezados, información de inicio y fin e información de control, en este proceso cada capa tiene una unidad de datos entrante y genera una unidad de datos saliente.

En este proceso llamamos:

- N-PDU (*Unidad de datos de protocolo*): es la información intercambiada entre las capas N pares (entidades pares) de dos hosts comunicados. Está compuesta por:
    - N-SDU (*Unidad de datos del servicio*) son los datos intercambiados por las unidades pares a través de la red.
    - N-PCI (*Información de control del protocolo*) Información intercambiada entre entidades pares conectadas para coordinar su operación conjunta.

- N-IDU (Unidad de datos de la interface): es el bloque de información transferido entre dos capas adyacentes del mismo host, a través de la interface entre ellas. Está compuesta por:
    - N-ICI (Información de control de la interface): es la información intercambiada entre una entidad y otra para coordinar la operación, controla la interface
    - Datos de Interface-(N) Información transferida entre entidades pares a través de la red, coincide con la (N+1) PDU

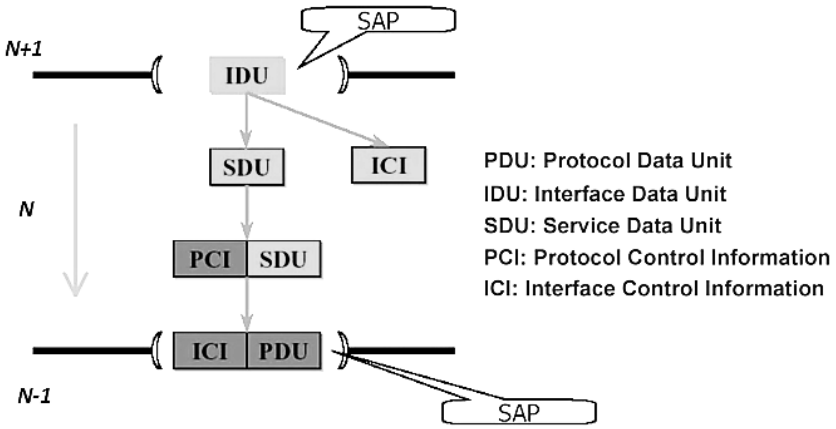


Figura 2.4 Trayectoria de los datos al atravesar las capas adyacentes en el host emisor

En el proceso de atravesar todas las capas se van formando las PDU (Protocol Data Unit) o unidad de datos de protocolos, este proceso recibe el nombre de *encapsulamiento* en el host emisor, mientras que en el host receptor al subir por la pila de capas se revierte el proceso, implementado el *desencapsulamiento.*

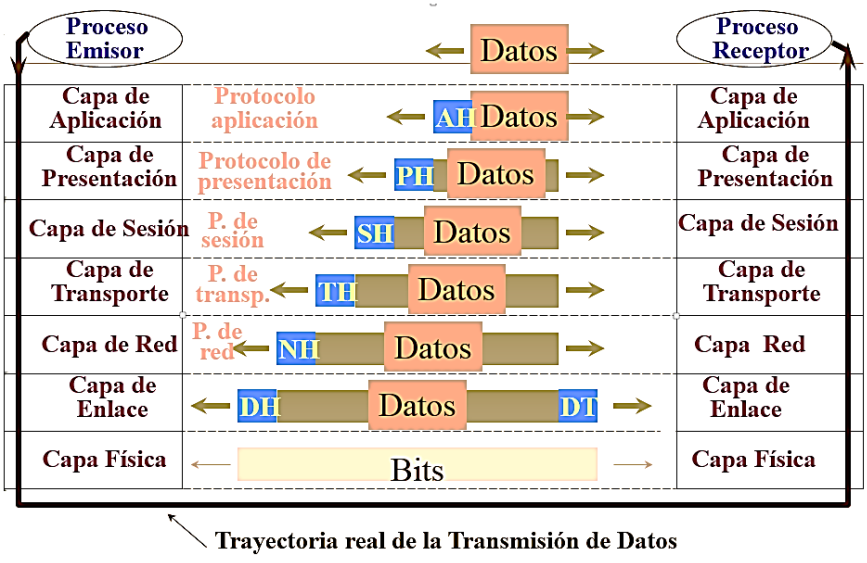


Figura 2.5 Encapsulamiento en el Modelo OSI

2.1.3 Servicios

Las entidades en un nivel *N* ofrecen servicios que son utilizados por las entidades del nivel *N+1*. El nivel *N* es el proveedor del servicio y el nivel *N+1* el usuario del servicio. Los servicios están disponibles en los *SAPs* (Puntos de Acceso al Servicio). Los SAPs del nivel

N son los puntos donde el nivel *N+1* puede acceder a los servicios ofrecidos por este. Un servicio es requerido por el usuario o es ofertado por el proveedor del servicio mediante el intercambio de un conjunto de primitivas de servicio a través de la interfaz entre los niveles *N* y *N+1*, estas primitivas son cuatro: *Request, Indication, Response, Confirm*. En general, los servicios pueden ser confirmados o no, se denomina servicio confirmado a aquel que utiliza las cuatro primitivas (se produce un diálogo de control), mientras que un servicio sin confirmar solo hace uso de las primitivas *Request* e *Indication*. El establecimiento de una conexión siempre es un servicio confirmado, mientras que la transferencia de datos puede ser sin confirmar o no. Los tipos de servicios comerciales (en la red) que encontramos son:

- **Servicios orientados a la conexión (P2P)**: requieren el establecimiento inicial de una conexión y la ruptura o liberación al final de la misma. En esta conexión se produce el intercambio de datos del usuario. Los bloques de datos (mensajes completos o bytes) se reciben en el mismo orden en que fueron emitidos y todos los paquetes siguen la ruta conseguida en la conexión, por ejemplo, el servicio telefónico.

- **Servicios sin conexión**: comunicación sin realizar una conexión con el destinatario. Se envían paquetes de datos (de tamaño fijo) con la dirección de destino, es la red la encargada de conducir los datos por una ruta apropiada. En algunos casos, el receptor debe enviar acuse de recibo al emisor, por ejemplo, el sistema postal.
    - Servicio de datagrama sin confirmación, no necesita confirmación del receptor, el datagrama no es confiable, por ejemplo, el protocolo IP, correo electrónico spam.
    - Servicio de datagrama con confirmación, requiere confirmación del emisor, es un datagrama con acuse de recibo (correo electrónico con acuse de recibo, correo registrado).
    - Servicio de petición y respuesta: a cada petición le sigue un mensaje de respuesta que contiene los datos solicitados (consulta de base de datos).

2.2 INTERNET Y TCP/IP

El objetivo principal de un modelo de referencia es ayudar a establecer funciones y procesos involucrados en el proceso de interconexión. Aunque el modelo de Interconexión de sistema abierto (OSI) es el modelo de referencia de internetwork más conocido y es usado para el diseño de redes de datos, especificaciones de funcionamiento y resolución de problemas, existe el modelo TCP/IP para las comunicaciones que puede trabajar en paralelo con el anterior, por lo que los diseñadores de servicios, dispositivos o protocolos de red pueden relacionar sus productos o servicios con el modelo OSI, el modelo TCP/IP o ambos.

El Protocolo de Internet (IP) y el Protocolo de Transmisión (TCP), se desarrollados en 1973 por el estadounidense Vinton Cerf como parte de un proyecto dirigido por el ingeniero norteamericano Robert Kahn y patrocinado por la Agencia de Programas Avanzados de Investigación (ARPA, siglas en inglés) del Departamento Estadounidense de Defensa. Internet comenzó siendo una red informática ARPA (ARPAnet) que conectaba redes de computadoras de varias universidades y laboratorios en investigación en Estados Unidos. World Wide Web se desarrolló en 1989 por el informático británico Timothy Berners-Lee para el Consejo Europeo de Investigación Nuclear (CERN, siglas en francés). Al inicio TCP/IP era conocido como el modelo de Internet, sus definiciones y protocolos se explican en un foro público y se definen en un conjunto de documentos disponibles, llamados Solicitudes de comentarios (RFC). Contienen las especificaciones formales de los protocolos de comunicación de datos y los recursos que describen el uso de los protocolos, también

contienen documentos técnicos y organizacionales sobre Internet, incluyendo las especificaciones técnicas y los documentos de las políticas producidos por el Grupo de Trabajo de Ingeniería de Internet (IETF).

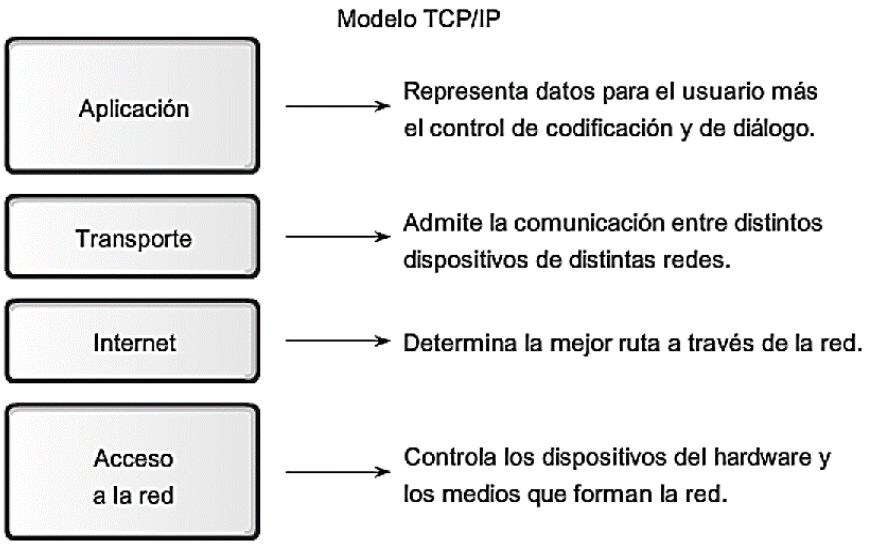


Figura 2.1 Modelo TCP/IP

La arquitectura TCP/IP consta de cuatro niveles o capas en las que se agrupan los protocolos:

- Aplicación: Se corresponde con los niveles OSI de aplicación, presentación y sesión. Aquí se incluyen protocolos destinados a proporcionar servicios, tales como correo electrónico (SMTP), transferencia de ficheros (FTP), conexión remota (TELNET) y otros como el protocolo HTTP (Hypertext Transfer Protocol).
- Transporte: coincide con el nivel de transporte del modelo OSI. Los protocolos de este nivel, tales como TCP y UDP, se encargan de manejar los datos y proporcionar la fiabilidad necesaria en el transporte de los mismos.
- Internet: es el nivel de red del modelo OSI, incluye el protocolo IP, que se encarga de enviar los paquetes de información a sus destinos correspondientes. Es utilizado con esta finalidad por los protocolos del nivel de transporte.
- Acceso a la red: análogo al nivel físico del OSI, es la interfaz de la red real, no especifica ningún protocolo concreto, así es que se rige por las interfaces conocidas, como 802.2, CSMA/CD, X.25, etc.

2.1.1 Proceso de comunicación

La suite de protocolos TCP/IP, se implementan en los hosts emisores y receptores e interactúan para establecer la comunicación entre las aplicaciones a través de la red. Este proceso de comunicación incluye los siguientes pasos:

1. Creación de datos en la capa de aplicación del dispositivo de origen
2. Segmentación y encapsulamiento de datos a medida que pasan por el stack o pila de protocolos en el dispositivo de origen
3. Generación de datos en los medios de la capa de acceso a la red
4. Transportación de los datos a través del medio físico/inalámbrico, el cual está compuesta por medios y cualquier dispositivo intermediario
5. Recepción de los datos en la capa de acceso a la red del dispositivo de destino

6. Desencapsulamiento y reensamblaje al pasar por el stack en el dispositivo destino
7. Transmisión de datos a la aplicación de destino en la capa de aplicación del dispositivo de destino

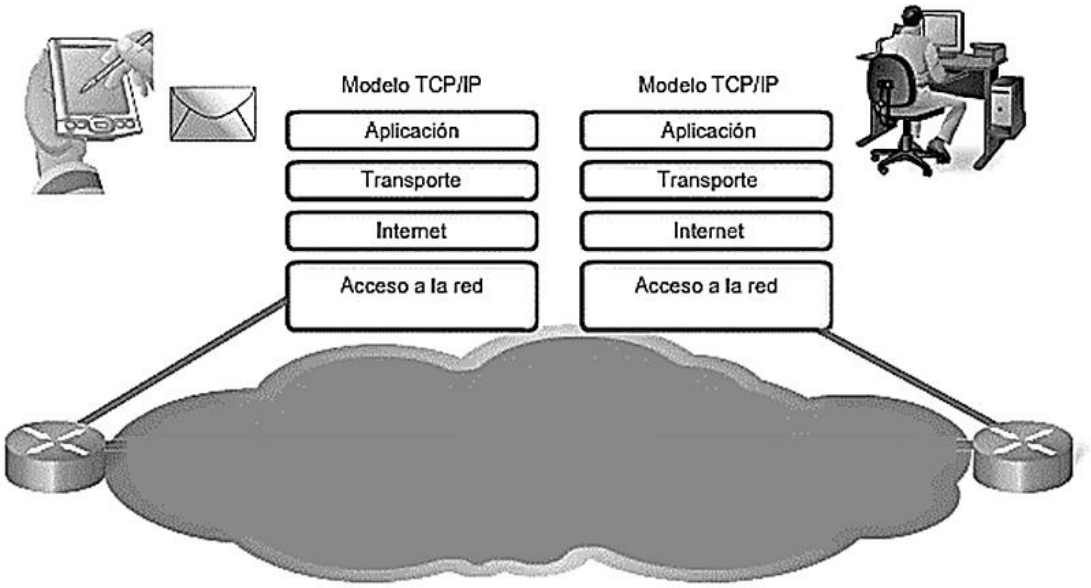


Figura 2.1 Ejemplo: envío de un mensaje

a) Encapsulamiento

Cuando los datos atraviesan la pila de capas, los protocolos de cada nivel le agregan información, proceso que se denomina encapsulación, las unidades de datos generadas en cada nivel con la información añadida reciben el nombre de Unidad de datos del protocolo (PDU), por lo que cada capa encapsula las PDU que recibe de la capa inferior de acuerdo al protocolo usado, y se le da un nombre distinto para identificar su nueva estructura, así tenemos:

- Datos: en la capa de aplicación
- Segmento: PDU de la capa de transporte
- Paquete: PDU de la capa de internet
- Trama: PDU de la capa de acceso de red
- Bits: PDU que se transmite físicamente por el medio

b) Envío y recepción

El stack de protocolos TCP/IP del host origen opera desde las capas superiores hacia las capas inferiores. Por ejemplo, si un servidor Web envía una página Web HTML a un cliente, se generará el siguiente proceso:

- El protocolo de la capa aplicación, HTTP, entrega los datos de la página Web (formato HTML) a la capa de transporte.
- La capa de transporte divide los datos en segmentos de TCP y a cada segmento le coloca un encabezado (información sobre qué procesos en el host destino deben recibir el mensaje, así como la información para ensamblar los datos de nuevo en su formato original) y los envía a la capa de Internet

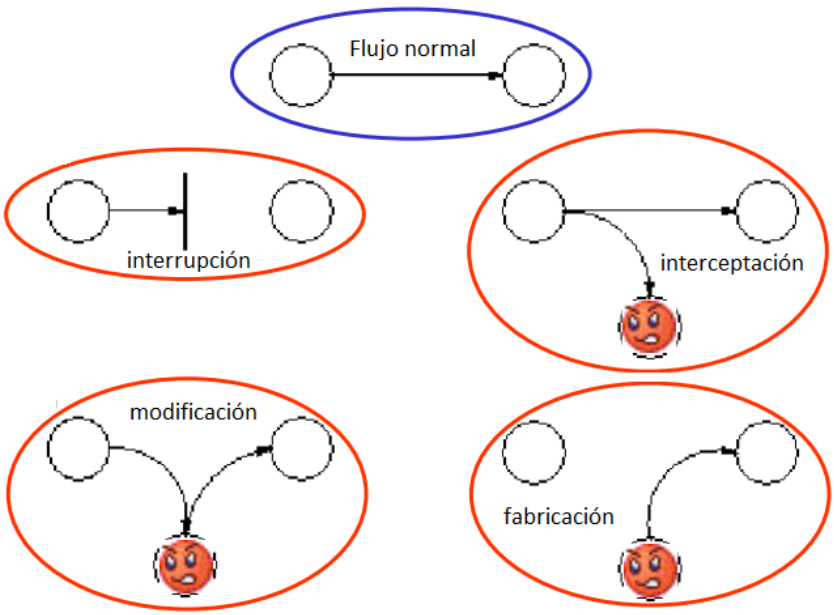
- En la capa de internet se implementa el protocolo IP, el segmento TCP se encapsula dentro de un paquete IP que agrega el encabezado IP (direcciones IP de host de origen y de destino y la información para entregar el paquete al proceso destino) y los envía a la capa de acceso a la red.
- En la capa de acceso a la red, el protocolo Ethernet encapsula el paquete IP en una trama y un tráiler. El encabezado de trama tiene la dirección física de origen y de destino, direcciones que identifican de forma única a los dispositivos de la red local, mientras que el tráiler contiene información para la verificación de errores. Finalmente, en la NIC (*network interface card)* del servidor los bits se codifican para ser enviados al medio Ethernet.

Este proceso se invierte en el host destino al momento de atravesar el host destino y las capas del modelo.

2.2 LA SEGURIDAD EN LA WEB

World Wide Web es una aplicación cliente-servidor corriendo sobre redes TCP/IP y presenta retos en relación a seguridad, debido a:

- Posibles ataques sobre los servidores Web
- La información sobre las corporaciones y productos puede ser perdida o adulterada
- El software de desarrollo Web tiene aún muchas vulnerabilidades
- Ataques locales específicos permiten el secuestro y la encriptación de la información
- Muchos de los usuarios finales, desconocen los riesgos y por ende las precauciones necesarias para un manejo seguro de su información



Dentro de los ataques más comunes tenemos:

- ❖ Rastreadores o sniffers
- ❖ Suplantaciones de IP o spoofing
- ❖ Ataques de contraseñas
- ❖ Control de salida ilegal de información sensible desde una fuente interna

- ❖ Ataques de hombre en el medio (man-the-middle attacks)
- ❖ Ataques de denegación de servicio, Denial of Service o ataques DoS.
- ❖ Ataques a nivel de aplicación para explotar vulnerabilidades conocidas
- ❖ Caballos de Troya (Trojan Horses), virus y otros códigos maliciosos

En base a estos ataques existen un conjunto de mecanismos de seguridad a ser considerados:

- De prevención:
  – mecanismos de autenticación e identificación
  – mecanismos de control de acceso
  – mecanismos de separación (física, temporal, lógica, criptográfica y fragmentación)
  – mecanismos de seguridad en las comunicaciones (cifrado de la información)

- De detección:
  – IDS (Intruder Detected System)

- De recuperación:
  – copias de seguridad (backup)
  – mecanismos de análisis forense: averiguar alcance, las actividades del intruso en el sistema

La tabla muestra un resumen de los problemas de seguridad que se producen en la red

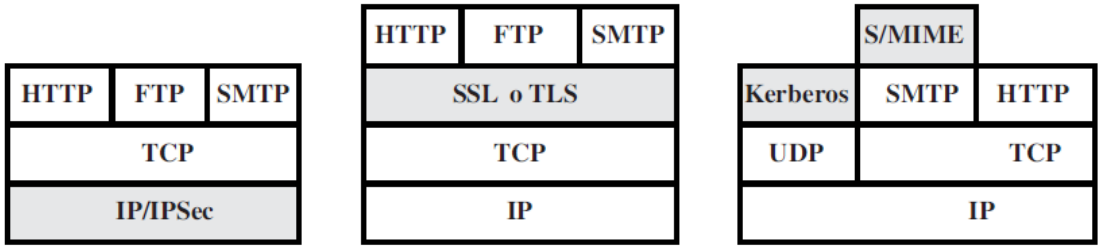
	ATAQUE	CONSECUENCIAS	PRECAUSIONES
INTEGRIDAD	• Modificación de datos de usuario • Caballo de Troya en buscador • Modificación de la memoria • Modificación del tráfico de mensajes en tránsito	• Pérdida de datos • Compromiso de la máquina • Vulnerabilidad a las demás amenazas	Checksum criptográficas
CONFIDENCIALIDAD	• Escucha de la red • Robo de información del servidor • Robo de datos del cliente • Información sobre la configuración de la red • Información sobre la que el cliente habla con el servidor	• Pérdida de información • Pérdida de privacidad	Encriptación, Proxys Web
DENEGACIÓN DE SERVICIO	• Matar hebras del usuario • Saturación de la máquina con peticiones falsas • Desborbamiento de disco o memoria • Aislamiento de la máquina por ataque al DNS	• Disruptivo • Molesto • Evita que el usuario reciba lo solicitado	Dificultad para prevenir
AUTENTICACIÓN	• Suplantación de identidad • Pérdida de datos	• Representación falsa del usuario • Tomar como cierta información falsa	Técnicas criptográficas

Los ataques también pueden clasificarse en:

- Ataques pasivos, que son aquellos que interceptan el tráfico entre navegadores y servidores para acceder a información restringida
- Ataques activos, en los que existe suplantación, alteración de los mensajes en tránsito entre servidor y cliente o la alteración de información en un sitio Web

Otra forma de clasificación de los ataques es en función a la localización de los mismos: servidor Web, navegador Web y tráfico de red entre navegador y servidor.

La provisión de seguridad Web corre a cargo de los protocolos corriendo en el stack TCP/IP, existiendo tres formas básicas de implementar la seguridad como se muestra en el diagrama siguiente



La forma más simple es usar IPsec, en el nivel más alto, proveyendo los servicios proxy, de encriptación y criptografía, los que son invisibles a los usuarios finales y son las aplicaciones son las que proveen soluciones generales, usando SSL (Secure Socket Layer) o TLS (Transport Layer Security), ambas provistas como parte de una suite de protocolos y por lo tanto son transparentes a las aplicaciones o provistos por aplicaciones particulares (por ejemplo: Netscape y Microsoft Explorer). Finalmente, en el diagrama c) se muestra como servicios de seguridad específicos están contenidos en aplicaciones particulares.

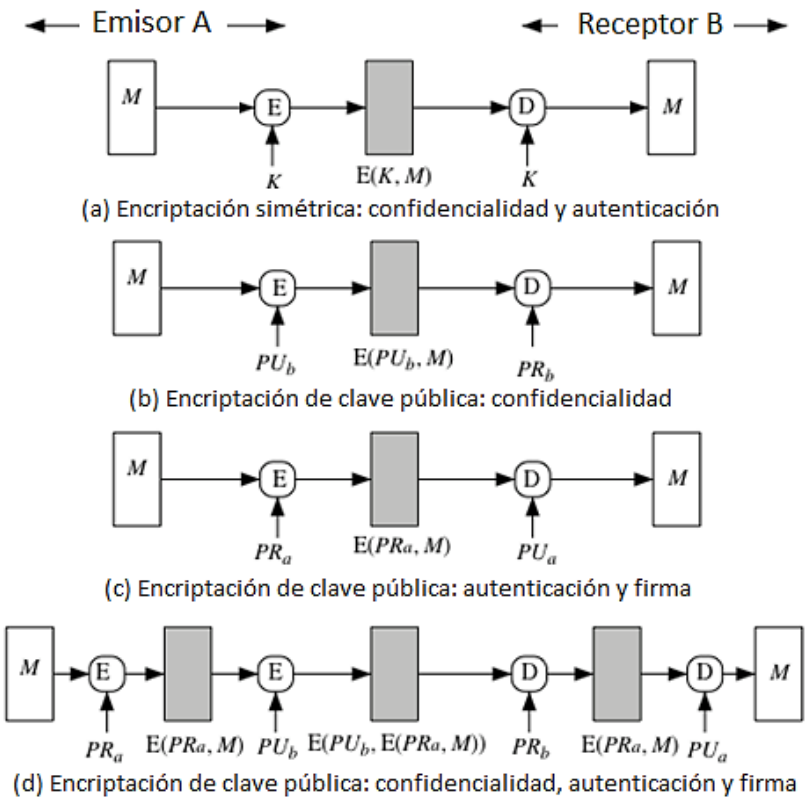
2.3 AUTENTICACION DE MENSAJES

En las comunicaciones a través de una red, se pueden dar los siguientes ataques:

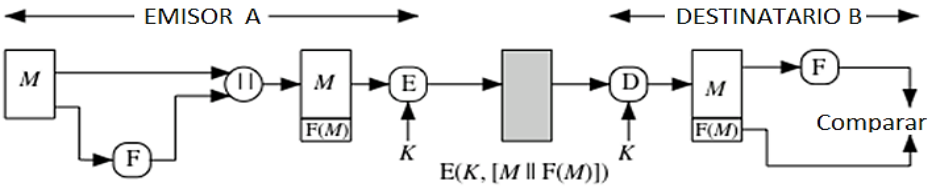
1. Divulgación: es liberación del contenido del mensaje a cualquier persona o proceso que no posea La clave criptográfica apropiada.
2. Análisis sobre el tráfico: descubrimiento del patrón de tráfico entre las partes. En una aplicación orientada a la conexión, la frecuencia y la duración de las conexiones podrían ser determinadas, en un entorno no orientado a la conexión se podría determinar el número y la duración de los mensajes entre las partes.
3. Mascarada: inserción de mensajes en la red desde una fuente fraudulenta, esto incluye la creación de mensajes por parte de un oponente que proceden de una entidad autorizada. También se incluyen reconocimientos fraudulentos de recibo de mensaje o de no recibo por alguien que no sea el destinatario del mensaje.
4. Modificación del contenido: Los cambios en el contenido de un mensaje, incluida la inserción, supresión, transposición y modificación.
5. Modificación de secuencias: cualquier modificación de una secuencia de mensajes, incluidas la inserción, la supresión y la reordenación.
6. Modificación de tiempo: retraso o repetición de mensajes. En una aplicación orientada a la conexión, toda una sesión o secuencia de mensajes podría ser una repetición de algunas sesiones anteriores válidas, o mensajes individuales en la secuencia podrían ser

retrasados o repetidos. En una aplicación no orientada a la conexión, un mensaje individual o datagrama, podría retrasarse o reproducirse.
7. Repudio de la fuente: denegación de la transmisión del mensaje por fuente.
8. Repudio de destino: denegación de recepción de mensaje por destino.

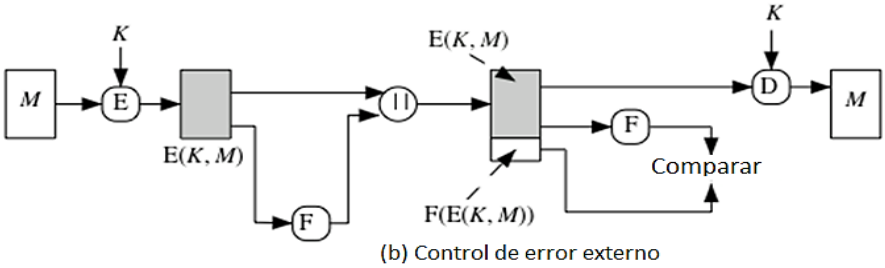
Las medidas para hacer frente a los dos primeros ataques están en el ámbito de la confidencialidad de los mensajes, las medidas para los puntos (3) a (6), son considerados como autenticación de mensajes, los mecanismos para tratar el punto (7) se hallan bajo las firmas digitales. En el punto (8) se requiere una combinación del uso de firmas digitales y un protocolo específico diseñado para contrarrestar este ataque.



Puede ser difícil determinar automáticamente si el texto cifrado entrante se descifra a un texto claro inteligible, por ejemplo, un archivo de objetos binarios o rayos X digitalizados. Por lo tanto, un oponente podría lograr un cierto nivel de interrupción simplemente mediante la emisión de mensajes con contenido aleatorio que pretenden venir de un usuario legítimo. Una solución a este problema es obligar al texto sin formato a tener cierta estructura que sea fácilmente reconocida pero que no pueda ser replicada sin recurrir a la función de cifrado. Por ejemplo, añadir un código de detección de errores, como una secuencia de comprobación de trama (FCS) o suma de comprobación, a cada mensaje antes del cifrado. Así el emisor A prepara un mensaje de texto plano M y luego proporciona esto como entrada a una función F que produce un FCS. El FCS se agrega a M y todo el bloque se cifra. En el destino, B descifra el bloque entrante y trata los resultados como un mensaje con un FCS añadido. B aplica la misma función F para intentar reproducir el FCS. Si el FCS calculado es igual al FCS entrante, entonces el mensaje se considera auténtico. Es improbable que cualquier secuencia aleatoria de bits exhiba la relación deseada.



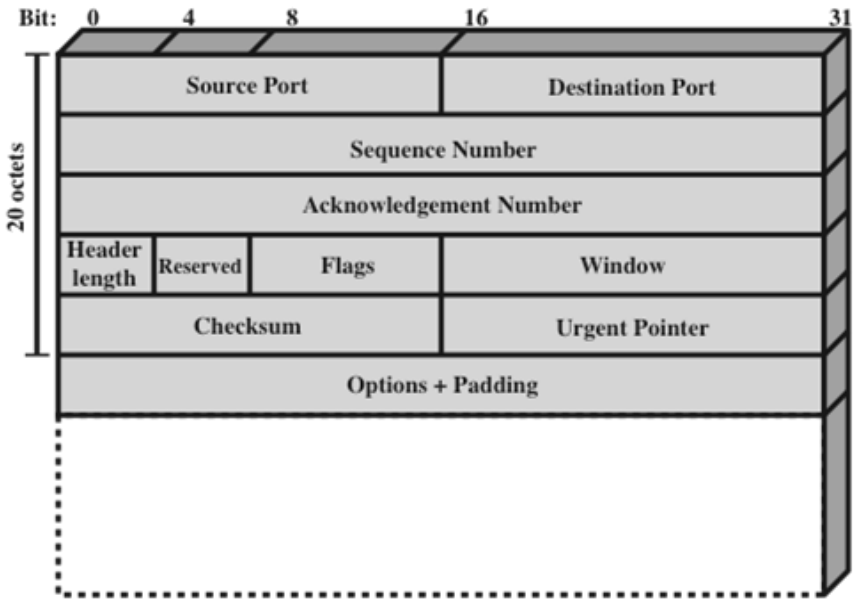
(a) Control de error interno



(b) Control de error externo

Hay que tener en cuenta que el orden en que se llevan a cabo las funciones FCS y cifrado es crítico. La secuencia ilustrada en la figura (a) se denomina control de error interno, si en cambio el FCS es en el código externo, un oponente puede construir mensajes con códigos válidos de control de errores. Aunque el oponente no puede saber qué es el texto cifrado podría crear confusión e interrumpir las operaciones.

Un código de control de errores es sólo una forma, cualquier tipo de estructura añadida al mensaje transmitido sirve para fortalecer la capacidad de autenticación, la estructura se añade en la arquitectura de comunicaciones que consta de protocolos en capas. En TCP / IP se añade en el encabezado del segmento TCP, si cada par de hosts debe compartir una clave secreta única usada en todos los mensajes independientemente de la aplicación, se podría encriptar todo el datagrama excepto el encabezado IP; si un oponente sustituye algún patrón de bits arbitrario para el segmento cifrado TCP, el texto claro descifrado no tendría un encabezado significativo, este encabezado incluye no sólo el checksum (suma de comprobación del encabezado) sino también otra información útil, tal como el número de secuencia, pero como esta numeración es secuencial, el cifrado asegura que un oponente no retrasa, desordena o elimina ningún segmento.



Cualquier mecanismo de autenticación de mensajes o de firma digital tiene dos niveles de funcionalidad: En el nivel inferior, debe haber algún tipo de función que produce un autenticador: un valor que se utilizará para autenticar un mensaje. Esta función de nivel inferior se utiliza entonces como una primitiva en un protocolo de autenticación de nivel superior que permite a un receptor verificar la autenticidad de un mensaje y puede ser de tres tipos:

- Función hash: función que asigna a un mensaje de cualquier longitud, un valor de hash de longitud fija, que sirve como autenticador
- Encriptación de mensajes: el texto cifrado de todo el mensaje sirve de autenticador
- Código de autenticación de mensajes (MAC): una función del mensaje y una clave secreta que produce un valor de longitud fija que sirve como autenticador

2.3.1 Código de Autenticación de Mensajes MAC

Una técnica de autenticación que usa una clave secreta para generar un pequeño bloque de datos de tamaño fijo, conocido como una suma de comprobación criptográfica o MAC, que se añade al mensaje. Esta técnica supone que dos partes comunicantes A y B, comparten una clave secreta común K. Cuando A tiene un mensaje que enviar a B, calcula el MAC como una función del mensaje y la clave:

MAC = C (K, M)

dónde

M = mensaje de entrada
C = función MAC
K = clave secreta compartida
MAC = código de autenticación de mensajes

El mensaje más MAC se transmiten al destinatario deseado. El destinatario realiza el mismo cálculo en el mensaje recibido, utilizando la misma clave secreta, para generar un nuevo MAC. El MAC recibido se compara con el MAC calculado, si sólo el receptor y el remitente conocen la identidad de la clave secreta, y si el MAC recibido coincide con el MAC calculado, entonces

3. El receptor está seguro de que el mensaje no ha sido alterado. Si un atacante altera el mensaje, pero no altera el MAC, el cálculo del MAC del receptor difiere del MAC recibido. Dado que se supone que el atacante no conoce la clave secreta, el atacante no puede alterar el MAC para que corresponda a las alteraciones del mensaje.

4. El receptor está seguro de que el mensaje es del supuesto remitente. Porque nadie más sabe la clave secreta, nadie más podría preparar un mensaje con un MAC adecuado.

5. Si el mensaje incluye un número de secuencia (tal como se utiliza con HDLC, X.25 y TCP), entonces el receptor puede estar seguro de la secuencia adecuada porque un atacante no puede alterar con éxito el número de secuencia.

Una función MAC es similar al cifrado. Una diferencia es que el algoritmo MAC no necesita ser reversible, como debe ser para el descifrado. En general, la función MAC es una función

de varios a uno. El dominio de la función consiste en mensajes de alguna longitud arbitraria, mientras que el rango consiste en todos los MAC posibles y todas las claves posibles. Si se utiliza un MAC de n bits, entonces hay 2^n MAC posibles.

El proceso proporciona autenticación, pero no confidencialidad, porque el mensaje como un todo se transmite en claro. La confidencialidad puede proporcionarse realizando el cifrado del mensaje después (b) o antes (c) del algoritmo MAC. En ambos casos, se necesitan dos claves independientes, cada una de las cuales es compartida por el remitente y el receptor. En el primer caso, el MAC se calcula con el mensaje como entrada y luego se concatena al mensaje. El bloque entero se cifra entonces. En el segundo caso, el mensaje se cifra primero. A continuación, el MAC se calcula utilizando el texto cifrado resultante y se concatena al texto cifrado para formar el bloque transmitido. Típicamente, es preferible vincular la autenticación directamente al texto sin formato, por lo que se utiliza el método de la figura (b)

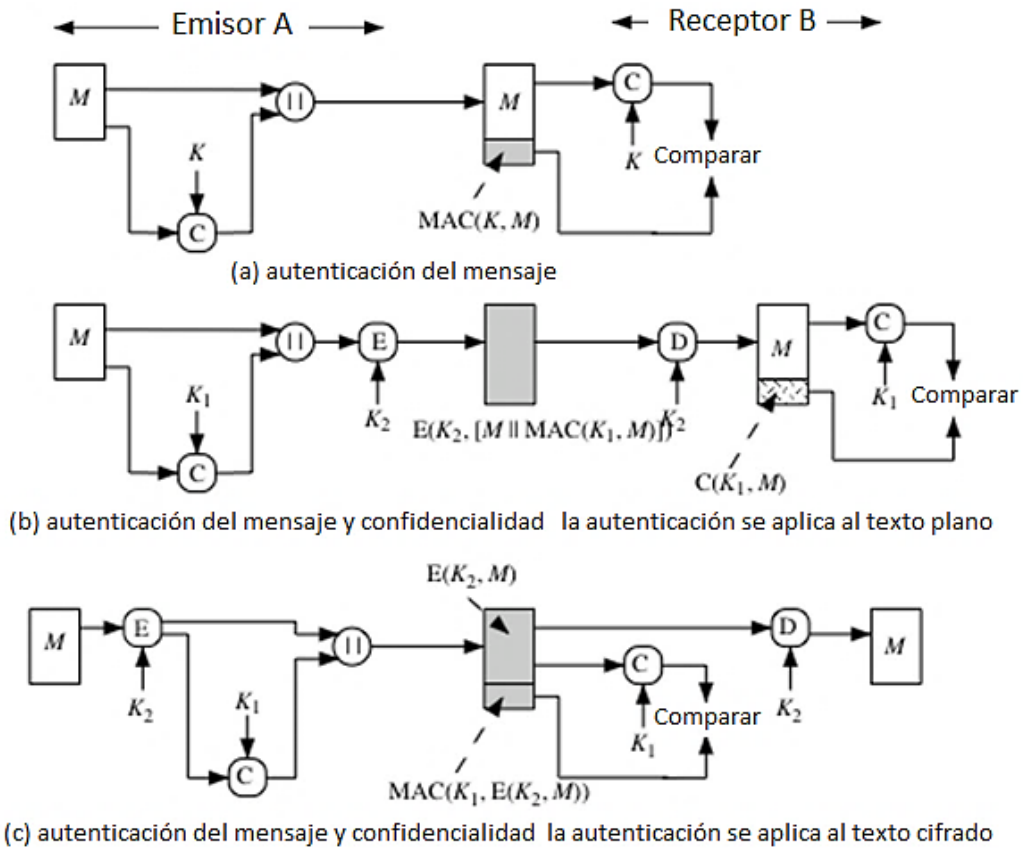
Al evaluar la seguridad de una función MAC, hay que considerar los tipos de ataques que pueden ser montados en su contra, debe satisfacer los siguientes requisitos.

1. El primer requisito se refiere a los ataques de reemplazo de mensajes, en los que un oponente es capaz de construir un nuevo mensaje para coincidir con un determinado MAC, aunque el oponente no sabe y no aprende la clave.

2. El segundo requisito se refiere a la necesidad de frustrar un ataque de fuerza bruta basado en el texto plano elegido.

3. El requisito final define que el algoritmo de autenticación no debe ser más débil con respecto a ciertas partes o bits del mensaje que otros.

Estas razones motivaron la generación de MAC a partir de funciones hash: HMAC, DAA y CMAC, AE, CCM y GCM, Key Wrap,



2.4 PROTOCOLOS Y MANEJO DE SEGURIDAD

Con la finalidad de que puedan crearse un conjunto de aplicaciones que corran de manera segura sobre Internet, hay un conjunto de protocolos para soportar ello.

2.4.1 SSL (Secure Socket Layer) / TLS (Transport Layer Security)

Protocolo desarrollado por Netscape, proporciona una comunicación segura end-to-end al incorporar cifrado mediante el empleo de criptografía, hoy cuenta con la versión 3, la versión 1.0 no fue publicada, la versión 2.0 se publicó en 1995 pero contenía puntos débiles en la seguridad. Las primeras implementaciones de SSL podían usar claves simétricas con un máximo de 40-bit por restricciones del gobierno de Estados Unidos sobre tecnología criptográfica, para asegurar que un ataque de fuerza bruta de las agencias de seguridad nacional pudiera leer el tráfico cifrado. Después de varios años de controversia, esta restricción ha desaparecido. Las implementaciones actuales usan claves de 128-bit (o más) para claves de cifrado simétricas

- TLS 1.0 (SSL 3.1)

En 1999 se introdujeron mejoras y se denominó TLS (Transport Layer Security), descrita en RFC 2246. y es un estándar IETF y es generalmente usado en los protocolos HTTPS, SSH, y otros tipos de comunicaciones cifradas. Es un protocolo robusto, usado por los navegadores actuales y en el peor de los casos, sino dispone del protocolo handshake TLS se usará SSL 3.0

- TLS 1.1 (SSL 3.2)