Seguridad e integridad de la información

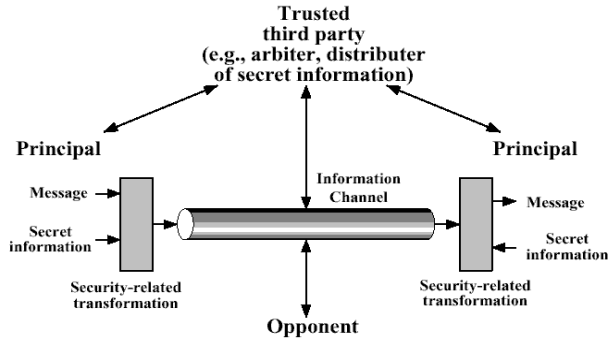
Resumen y notas de clases

Ultima modificación el miércoles 4 de octubre de 2006 a las 16:57:57

Administración de seguridad

- Política de seguridad
- Modelo de seguridad
- Arquitectura de seguridad
- Servicios de seguridad
- Mecanismos de seguridad
- Algoritmos de seguridad

Modelo de seguridad en redes



Servicios de seguridad

Confidencialidad

Protege los datos contra su divulgación no autorizada.

Integridad

Protege los datos contra su modificación no autorizada; sólo reporta, no repara.

Autenticación de entidad corresponsal

Verifica la identidad de quien declara estar en el otro extremo de una comunicación.

Autenticación de origen

Verifica la identidad de quien declara ser la fuente original de los datos.

No repudio de emisión (recepción)

Protege contra la falsa negativa del remitente (destinatario) de haber enviado (recibido) los datos.

Control de acceso

Protege los recursos contra su uso no autorizado.

Disponibilidad

Protege al sistema para asegurar que provea servicios según diseño, cuando sus usuarios los requieran.

Ataques

Un **ataque** es un asalto a la **seguridad** de un sistema, que se origina en una amenaza inteligente; un intento **deliberado** de evadir los servicios de seguridad y violar la política de seguridad del sistema.

Según su origen

Es **externo** si el atacante no tiene autorización alguna.

Es **interno** si el atacante emplea recursos en una forma no aprobada por quienes lo autorizaron.

Según su grado de intervención

Es **pasivo** si no altera los recursos, limitándose al intento de tomar conocimiento de información del sistema, ataca la **confidencialidad**.

Es **activo** si intenta alterar los recursos del sistema o afectar su operación.

Modificación: ataca la **integridad**, alterando el contenido, el orden o la temporización (retardo o repetición).

Fabricación: ataca la **autenticación**, insertando mensajes fraudulentos.

Interrupción: ataca la **disponibilidad**.

Sobre recursos **técnicos** (hardware)

Sobre recursos **humanos**.

Terminología

Criptografía

Arte y ciencia de proveer seguridad a los mensajes.

Criptoanálisis

Arte y ciencia de vulnerar la seguridad provista por la criptografía.

Criptología

Rama de las matemáticas que abarca a ambas.

Clave

La clave determina un mapeo entre cada entrada posible y la salida correspondiente.

Algoritmos

Se supone que los algoritmos son conocidos (Kerckhoffs), es decir que la seguridad reside únicamente en el secreto de la clave.

Criptoanálisis

Un algoritmo criptográfico es **incondicionalmente seguro** si una cantidad ilimitada de texto cifrado no proporciona suficiente información para vulnerarlo. **Sólo se conoce uno: el one-time pad, que es poco práctico (clave aleatoria no reutilizable).**

Los demás algoritmos se pueden vulnerar por **fuerza bruta**, descifrando con cada clave posible hasta obtener mensajes con significado.

Una clave de n bits requiere,en promedio,2^{n-1} pruebas. Para algunos algoritmos existen mejores ataques.

Un algoritmo es **computacionalmente seguro** si no se puede vulnerar con los recursos disponibles. Se tiene en cuenta: cantidad de datos necesarios, tiempo de procesamiento, necesidades de almacenamiento.

Ataques comunes

Los ataques más comunes son:

Sólo texto cifrado (ciphertext-only): intenta recuperar texto llano y/o la clave, conociendo sólo mensajes cifrados.

Texto llano conocido (known-plaintext): intenta recuperar la clave, conociendo (o suponiendo) pares de mensajes llanos y cifrados.

Texto llano elegido (chosen-plaintext): como el anterior, pero proponiendo mensajes a cifrar. Puede ser adaptativo.

Texto cifrado elegido (chosen-ciphertext): como el anterior, pero proponiendo mensajes a descifrar.

Clasificación de cifradores

Por la cantidad de claves

Una clave (algoritmo simétrico o de clave secreta)

Por el modo de procesamiento:

Por bloques

De flujo continuo (stream)

Dos claves (algoritmo asimétrico o de clave pública y privada)

Seguridad de las claves

En los algoritmos **simétricos**, la clave debe ser distribuida por un canal seguro, en forma **confidencial**.

En los algoritmos **asimétricos**, la clave publica debe distribuirse en forma **autenticada**.

Digest o hash criptográfico

Sin clave.

Con **M** de cualquier tamaño, proporciona **H(M)** de tamaño fijo y pequeño.

Unidireccional: computacionalmente imposible obtener **M** dado **H(M)**.

Resistente a **colisiones**: computacionalmente imposible hallar mensajes $\mathbf{M_1 != M_2}$ tales que $\mathbf{H(M_1) = H(M_2)}$, fijado o no uno de ellos.

Código de autenticación de mensaje (MAC)

Con clave simétrica, M (mensaje) de cualquier tamaño, proporciona CK(M) de tamaño fijo y pequeño. Es **unidireccional** (del resultado no puedo obtener el mensaje original) y resistente a colisiones.

Basado en función hash (HMAC) o en cifrador con encadenamiento (CMAC).

Un código de autenticación de mensajes, MAC (Message Authentication Code) es una construcción que preserva la integridad de los mensajes enviados a través de un canal inseguro. Mientras el cifrado previene que un atacante pueda leer el mensaje, aprendiendo la información que viaja en él, un MAC previene la manipulación de los mensajes.

Los MAC también necesitan de una clave privada o simétrica (como hemos indicado anteriormente), que sólo conozcan el origen y el destino del mensaje, pero no los atacantes. Matemáticamente se define como una función que toma dos argumentos, una clave K de tamaño fijo y un mensaje m de longitud arbitraria. El resultado es un código MAC de longitud fija.

HMAC soporta claves de cualquier tamaño y se permite truncar la salida tomando una cantidad de MSB no inferior a la mitad (se puede truncar a 32 bits si por otros medios se limitan ataques por prueba y error).

CMAC esta basado en cifrado en modo CBC, con IV = 0. No usa el algoritmo de descifrado. La salida es el último bloque cifrado. Se permite truncar como en HMAC.

Algoritmos comunes empleados en crear el hash MAC

- MD5 : salida de 128 bits (ya no se considera computacionalmente seguro), bloque interno: 512 bits, con 64 rounds.
- SHA : existen varios, con bloque de 512 bits existe SHA-1 (160 bits de salida), SHA-224 (224 b/s), SHA-256 (256 b/s), y con bloque de 1024 bits esta SHA-384 (384 b/s), SHA-512 (512 b/s).

Limitaciones de la criptografía

Un sistema criptográfico se debilita si los mensajes tienen cierta **estructura,** no evita que el atacante **elimine** datos. No siempre evita el **análisis de tráfico**.

Se inutiliza si el atacante accede al texto **llano, a**ccede a la **clave**, o puede adivinarla, altera el **software** de seguridad o conoce una forma relativamente fácil de **romper** el algoritmo, o una **puerta trasera**.

Criptografía de clave simétrica

Una sola clave **secreta** compartida. Los algoritmos son más **rápidos** que los de clave **pública**.

La necesidad de **compartir** la clave en forma **segura** es una desventaja; requiere *mecanismos de distribución de claves.*

Algoritmos de ejemplo:

- AES
- DES (no recomendado) y Triple DES (TDEA)
- RC2 (bloque de 64 bits, clave de tamaño variable)
- RC4 (de flujo, clave de tamaño variable)
- RC5 (bloque de 64 o 128 bits, clave de tamaño variable)
- IDEA (bloque de 64 bits, clave de 128 bits)

Cifrador de flujo con XOR (O-exclusiva)

Es un sistema perfecto si la clave es verdaderamente aleatoria y nunca se repite.

La puerta lógica O-exclusiva, más conocida por su nombre en inglés XOR, realiza la función booleana A'B+AB'. Su símbolo es el más (+) inscrito en un círculo : F = A (+) B

Su tabla de verdad es la siguiente:

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

Propiedades

Conmutativa: A (+) B = B (+) A

Asociativa: A (+) (B (+) C) = (A (+) B) (+) C

0 (+) A = A 1 (+) A = ~A

A (+) A = 0 A (+) ~A = 1

A (+) B (+) A = B ~(A (+) B) = ~A (+) B = A (+) ~B

Data Encryption Standard (DES)

Bloque de 64 bits.

Clave de 56 bits.

16 rounds.

Triple DES (TDES)

El DES **simple** ya no se recomienda porque es vulnerable por fuerza bruta.

Se ha demostrado que el **doble** DES no incrementa la seguridad.

El esquema **E-D-E** permitiría que trabaje como DES **simple** con $\mathbf{K_3 = K_2 = K_1}$.

3TDEA: 3 claves distintas (**168** bits en total).

2TDEA: 2 claves con $\mathbf{K_3 = K_1}$ (**112** bits en total).

Advanced Encryption Standard (AES)

El sucesor de DES.

AES trabaja con un bloque de 128 bits, y un tamaño de clave de 128, 192 o 256 bits.

Cifrador de Feistel

Para **cifrar**, en cada round se substituye la mitad izquierda, sometiéndola a la o-exclusiva con la función F de la mitad derecha, parametrizada por la subclave; luego se permutan las mitades.

Para **descifrar**, Se emplea el mismo algoritmo, pero con las subclaves aplicadas en orden inverso.

Modos de operación en algoritmos de clave simétrica

Para confidencialidad

ECB (electronic codebook)

CBC (cipher block chaining)

CFB (cipher feedback)

OFB (output feedback)

CTR (counter)

Para autenticación

CMAC (cipher-based MAC)

Para autenticación y confidencialidad

CCM (counter with CBC-MAC)

Padding: (relleno) hasta obtener un múltiplo del tamaño de bloque: p. ej. siempre un 1 seguido de suficientes 0.

Criptografía de clave asimétrica

Utilizan dos claves matemáticamente relacionadas, una publica y una privada (que se mantiene en **secreto**). La clave publica debe distribuirse en forma **autenticada**.

Es computacionalmente imposible obtener una clave a partir de la otra.

Ejemplos: Diffie-Hellman -sólo intercambio de clave- (1976), RSA (1978), ElGamal (1985), DSA -sólo firma digital- (1991).

Estos algoritmos son mucho mas lentos que los de clave simétrica.

Esta criptografía proporciona autenticación, integridad, y no repudio de emisión (firma digital).

Debe asegurarse la autenticidad de las claves públicas mediante certificado; de lo contrario un impostor puede violar la **confidencialidad** del mensaje, **personificando** al **destinatario**, y/o violar la **autenticidad** del mensaje, **personificando** al **remitente**.

Algoritmos de clave asimétrica

Diffie-Hellman (solo intercambio de claves)

El protocolo Diffie-Hellman (debido a Whitfield Diffie y Martin Hellman) permite el intercambio secreto de claves entre dos partes que no han tenido contacto previo, utilizando un canal inseguro.

Se emplea generalmente como medio para acordar claves simétricas que serán empleadas para el cifrado de una sesión.

Su seguridad radica en la extrema dificultad (conjeturada, no demostrada) de calcular logaritmos discretos en un campo finito.

Sólo para acuerdo de clave secreta, con clave pública y privada.

El par de claves puede reutilizarse o ser efímero.

Ambos participantes obtienen el mismo número, intercambiando solamente valores públicos.

RSA

Es un algoritmo asimétrico cifrador de bloques, con una clave pública y otra privada.

Emplea expresiones exponenciales en aritmética modular.

Su seguridad radica en la extrema dificultad (conjeturada, no demostrada) de descomponer un número muy grande en sus factores primos.

Actualmente se recomiendan claves (módulos) de unos 1500 bits (aprox. 450 dígitos decimales).

ElGamal

Tiene dos usos: encriptación y firma digital.

El algoritmo de encriptación es similar en su naturaleza a Diffie-Hellman.

El protocolo ElGamal de negociación de claves provee negociación en un solo paso y autenticación unilateral (del receptor hacia el emisor) si la clave pública del receptor es conocida de antemano por el emisor.

La seguridad del algoritmo depende en la dificultad de calcular logaritmos discretos.

DSA (solo firma digital)

Este algoritmo como su nombre lo indica, sólo sirve para firmar y no para cifrar información. Una desventaja de este algoritmo es que requiere mucho más tiempo de cómputo que RSA.

Certificados de clave publica

Un certificado contiene la clave pública de un sujeto, quien demostró conocer la clave privada.

Asocia esta clave pública con los datos identificatorios del sujeto, ademas establece el período de validez de esta clave y establece formas de uso de esta clave.

Está firmado por alguna Autoridad Certificante (issuer o CA). El usuario debe confiar en los procedimientos de este ente y conocer en forma segura, mediante otro certificado válido, la clave pública de la CA, a fin de verificar la autenticidad del certificado.

Autenticación

Kerberos

Kerberos es un protocolo de autenticación de red que permite que individuos comunicándose sobre una red insegura prueben su identidad a otro en una manera segura.

Evita ataques de escucha (eavesdropping) o repetición (replay), y asegura la integridad de los datos, ya que puede proporcionar **integridad** y **confidencialidad**.

Esta basado en **criptografía de clave simétrica**, y necesita de un **tercero confiable** (KDC o Key Distribution Center) el cual consiste de dos partes lógicas : un servidor de autenticación (AS) y un Servidor de Otorgamiento de Tickets (TGS).

Kerberos trabaja sobre la base de "**tickets**" los cuales sirven para probar la identidad de los usuarios.

El sistema mantiene una base de datos de claves secretas, las cuales corresponden cada una a una entidad en la red. Esta clave es conocida solo por la entidad y Kerberos ; el conocimiento de esta clave permite probar la identidad de una entidad de la red. Para comunicación entre dos entidades, Kerberos genera una clave de sesion (el ticket) el cual utilizan para validar sus interacciones.

Como funciona

Utiliza un Key Distribution Center (KDC), un tercero confiable que conoce las claves.

El KDC dividido en dos componentes lógicos: Authentication Server (AS) y Ticket-Granting Server (TGS).

Una vez por logon del usuario:

> El cliente solicita a AS credenciales para presentar ante TGS.
>
> AS le entrega un TGT (ticket-granting ticket) reusable.
>
> El usuario introduce su contraseña y descifra la respuesta.

Para cada tipo de servicio:

> El cliente solicita a TGS credenciales para el servidor de aplicación, presentando el TGT y un autenticador no reusable.
>
> TGS verifica la autenticidad y le entrega un SGT (service-granting ticket) reusable.

Para cada sesión con un servidor de aplicación:

> El cliente presenta el SGT y un autenticador no reusable; además, puede proponer una subclave de sesión no reusable.
>
> El servidor de aplicación verifica la autenticidad; opcionalmente, se autentica ante el cliente (autenticación mutua).

Protocolos de comunicación segura

Secure Sockets Layer (SSL) y su sucesor Transport Layer Security (TLS)

Sirven para proteger una sesión entre cliente y servidor.

SSL proporciona autenticación, integridad y privacidad de la información entre extremos sobre Internet mediante el uso de criptografía.

La **autenticación**, del servidor frente al cliente, opcionalmente, del cliente frente al servidor, se hace mediante **certificados de clave pública.**

La **integridad** se garantiza mediante **MAC** y números de secuencia.

La **confidencialidad**, opcional, mediante **cifrado con algoritmo simétrico.**

SSL implica una serie de fases básicas:

- Negociar entre las partes el algoritmo que se usará en la comunicación.
- Intercambio de claves públicas y autenticación basada en certificados digitales.
- Encriptación del tráfico basado en cifrado simétrico.

Handshake

Negociación de algoritmos y parámetros.

Autenticación (del servidor o mutua).

Canal seguro para compartir un secreto inicial.

Derivación de claves en cada extremo.

Integridad de todo el intercambio.

Transferencia de datos

Usa las claves derivadas en el handshake.

Provee integridad.

Opcionalmente, provee confidencialidad.

Autentica el cierre de cada conexión.

Sesión

Asociación entre cliente y servidor, creada por el protocolo de handshake.

Puede incluir múltiples conexiones, sucesivas o simultáneas.

Mantiene un estado actual y un estado pendiente; al concluir con éxito el handshake, el estado pendiente pasa a ser actual.

Su estado se caracteriza por:

identificador de sesión,

certificado del corresponsal,

especificaciones de cifrado, y

master secret (48 bytes).

Conexión

Bidireccional, p.ej. TCP; pertenece a una sesión.

Cada sentido (client write o server write) se caracteriza por:

número aleatorio (32 bytes), clave de cifrado, secreto para el MAC, y número de secuencia.

Medidas de seguridad de SSL y TLS

- Numerando todos los registros y usando el número de secuencia en el MAC.
- Usando un resumen de mensaje mejorado con una clave (de forma que solo con dicha clave se pueda comprobar el MAC). Esto se especifica en el RFC 2104).
- Protección contra varios ataques conocidos (incluidos ataques man in the middle attack), como los que implican un degradado del protocolo a versiones previas (por tanto, menos seguras), o conjuntos de cifrados más débiles.
- El mensaje que finaliza el protocolo handshake (Finished) envía un hash de todos los datos intercambiados y vistos por ambas partes.
- La función pseudo aleatoria divide los datos de entrada en 2 mitades y las procesa con algoritmos hash diferentes (MD5 y SHA), después realiza sobre ellos una operación XOR. De esta forma se protege a sí mismo de la eventualidad de que alguno de estos algoritmos se revelen vulnerables en el futuro.

Como funciona

[completar , igual no hay apuro , esto va en el 2ndo parcial, o sea, SSL no entra en el primer parcial]

OpenSSL

Es un toolkit criptográfico que implementa SSL, TLS y otros algoritmos criptográficos y auxiliares que se requieren. Se basa en la biblioteca SSLeay de Young y Hudson. Es open source y su licencia permite usarlo tanto para propósitos comerciales como no-comerciales.

Stunnel

Es un wrapper (envoltorio) que permite proteger conexiones TCP arbitrarias con SSL; requiere OpenSSL o SSLeay. Es open source y su licencia GNU-GPL permite usarlo tanto para propósitos comerciales como no-comerciales.

Puede tanto funcionar como cliente SSL, o como servidor SSL y verificar certificados tanto de servidor como de cliente.