Unidad 3 – CRIPTOGRAFIA

CONCEPTOS BASICOS

Criptografía
Conglomerado de técnicas que tratan sobre la protección u ocultamiento de información frente a observadores no autorizados. A su vez, también permite la autentificación de la información, es decir, identificar al autor. Existen dos trabajos fundamentes, el de Claude Shannon que sienta las bases de la teoría de la información y criptografía moderna, y el de Whitfield Diffie y Martin Hellman que introducen el concepto de criptografía asimétrica. Toda criptografía lleva un criptoanálisis.

Criptosistema
El criptosistema está compuesto por una quíntupla (m, D, E, k, C) donde:

$$D(k, E(k, m)) = m$$

- M: representa el conjunto de todos los mensajes sin cifrar (texto claro) que pueden ser enviados.
- C: representa el conjunto de todos los posibles mensajes cifrados, o criptogramas.
- K: representa el conjunto de claves que se pueden emplear en el criptosistema.
- E: conjunto de transformaciones de cifrado o familia de funciones que se aplica a cada elemento de M para obtener un elemento C. Existe una transformación diferente E para cada valor posible de la clave K.
- D es el conjunto de transformaciones de descifrado, análogo a E.

Si se obtiene un mensaje **M**, lo ciframos empleando la clave **K**, luego lo desciframos empleando la misma clave, obteniendo **M** nuevamente.

Esteganografía
Consiste en ocultar en el interior de una información otro tipo de información (cifrada o no).

Criptoanálisis
Consiste en comprometer la seguridad de un criptosistema. Se puede realizar descifrando un mensaje sin conocer la llave, o bien obteniendo a partir de uno o mas criptogramas la clave que ha sido empleada. Hemos de suponer que el algoritmo siempre es conocido. Suele llevarse a cabo estudiando grandes cantidades de pares de mensajes.

- **Texto Claro:** Por medio de pares de textos claros seleccionados, permite efectuar operaciones, pero no nos permite leer su clave.
- **Fuerza Bruta:** Técnicas que buscan exhaustivamente por el espacio de claves ***K.***
- **Ataque:** Cualquier técnica que permita recuperar un mensaje cifrado empleando menos esfuerzo computacional que el que se usaría por la fuerza bruta.

- **Análisis Diferencial:** Parte de pares de mensajes con diferencias mínimas (usualmente un bit) y estudia las variaciones que existen entre los mensajes cifrados correspondientes, tratando de identificar patrones comunes.
- **Análisis Lineal:** Emplea operaciones XOR entre algunos bits del texto claro y otros del texto cifrado, obteniendo así un bit único. Haciendo esto en reiterados casos, se obtiene una variable ***p*** que podrá ayudar a recuperar la clave.

CLASIFICACION

1) **Clásica**
    - i) **Transposición**
        - (1) Grupos
        - (2) Series
        - (3) Columnas
        - (4) Filas
    - ii) **Sustitución**
        - (1) Monoalfabética
        - (2) Polialfabética
2) **Moderna**
    - i) **Simétrica**
        - (1) Cifrado de Bloque
            - (a) Modos de Operación
        - (2) Cifrado de Flujo
            - (a) Generadores de Secuencia
    - ii) **Asimétrica**
        - (1) Establecer Claves
        - (2) Firma
        - (3) Cifrado
    - iii) **Funciones de HASH**

Criptografía Clásica

- **Cifradores de Transposición**: Utilizan una técnica de permutación de forma que los caracteres del texto se reordenan mediante un algoritmo especifico.
- **Cifradores por Sustitución**: Utilizan la técnica de modificación de cada carácter del texto en claro por otro correspondiente al alfabeto de cifrado.
  i) Monoalfabético: Donde el alfabeto cifrado es igual al del mensaje.
  ii) Polialfabético: El alfabeto cifrado es distinto al del mensaje. Utilizan diferentes caracteres para el reemplazo de un mismo carácter de origen. Cifrado de Vigenere.

Referencias

- Binario: 1 y 0.
- Hexadecimal: 123456789ABCDEF
- Base 64: todos los caracteres.

Criptografía Moderna

1. **Simétrica:** Son aquellos que emplean la misma clave ***K*** tanto para cifrar como para descifrar. El inconveniente es que, para transmitir el mensaje por medio de comunicaciones, la clave ***K*** tiene que estar tanto en el emisor como en el receptor.
   - **i) Ventajas:** Sencillez de implementación, robustez, velocidad de cifrado, longitud del mensaje.
   - **ii) Desventajas:** La clave debe ser compartida perdiendo seguridad. La comunicación entre múltiples factores requiere múltiples claves.

   Ejemplos:

   - **DES-LUCIFER**
   - **Serpent**
   - **MARS**
   - **IDEA**
   - **3DES**
   - **Twofish**
   - **GOST**
   - **Blowfish**
   - **AES-Rijndael**
   - **RC6**
   - **CAMELLIA**
   - **RC5**

a) Referencia Matemática – XOR

- Es conmutativa: A xor B = B xor A
- Es asociativa: (A xor B) xor C = A xor (B xor C)
- Es autoinversa: (A xor B) xor B = A

b) Modos de Cifrado en Bloques

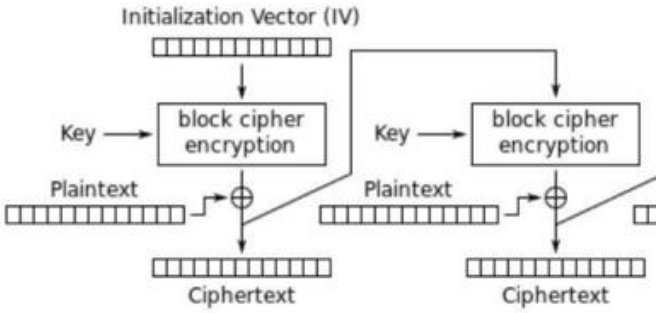
1. **ECB:** Electronic codebook, el mensaje se fracciona en partes y cada una es cifrada de manera independiente. Son bloques de una misma longitud. Emplean una técnica para el relleno de los mismos denominada Padding o Esquema de Relleno.

- **Padding:** ya que el método de cifrado en bloques emplea la separación de la información en bloques de misma longitud, el padding hace que el ultimo bloque de la cadena también posea esa misma longitud. Algunos algoritmos son los siguientes:

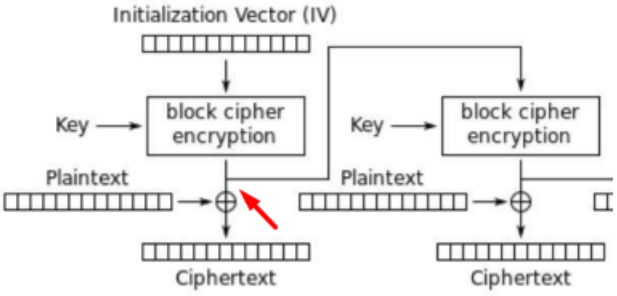
- **Bit Padding**
- **ISO/IEC 7816-4**
- **PKCS#7**
- **ISO 10126**
- **ANSI X.923**

2. **CBC:** Cipher Block Chaining, el mensaje se fracciona en partes y se realiza un XOR con el bloque previo antes de cifrar cada parte. Esto introduce un elemento de encadenamiento entre los bloques de datos, lo que ayuda a aumentar la seguridad y resistencia a ciertos tipos de ataques.

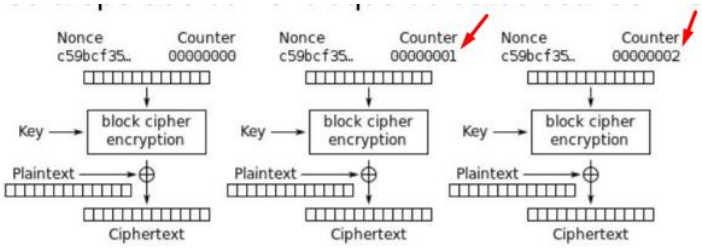
3. **CFB:** Cipher Feedback, el mensaje se fracciona en partes, se cifra un vector de inicialización y al resultado se le realiza un XOR con el bloque del mensaje. La salida de este bloque se utiliza como retroalimentación para los siguientes bloques.



4. **OFB:** Output Feedback, opera de manera similar al CFB con la diferencia que al bloque al ser utilizado como entrada del siguiente proceso, es tomado de la salida del algoritmo justo antes de realizar la XOR.



5. **CTR:** Modo de Counter o contador, en este modo de operación se utiliza un "nonce" (secuencia de valores) que se combina con una clave en un algoritmo de cifrado para generar una secuencia de valores de cifrado, que luego se combinan con los datos originales mediante una operación XOR para producir una salida cifrada.

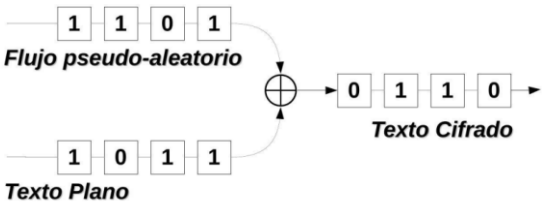


Modos de Cifrado de Bloques

- **PCBC**
- **EAX**
- **XCBC**
- **CCM**
- **GCM**
- **CWC**
- **PCFB**

c) Cifrado de Flujo

Se utiliza una función generadora de bits pseudo-aleatorios a fin de obtener un flujo de bits que pueda ser procesado con los bits del mensaje mediante una operación básica (XOR). Cada bit o byte es cifrado de manera independiente utilizando una secuencia de bits aleatorios llamada flujo clave.



Una función generadora de bits pseudo aleatoria es la que permite obtener secuencias criptográficamente aleatorias. Los posibles algoritmos de cifrado son:

- **RC4**
- **ChaCha20**
- **Chameleon**
- **Grain**
- **Phelix**
- **SOBER/SOBER-128**
- **Salsa20**
- **Trivium**
- **FISH**
- **MUGI**
- **Pike**
- **WAKE**
- **ISAAC**
- **A5/1, A5/2**
- **Helix**
- **Panama**
- **SEAL**
- **Rabbit**

2. Funciones de HASH

Es una función o método no reversible para generar un valor que represente de manera casi univoca a un dato. Sus principales usos son:

- Soporte para criptografía asimétrica
- Tablas de Hash
- Verificación de integridad
- Soporte para procesos de autenticación

Propiedades

- **r(m)** es de longitud fija, independientemente de la longitud de **m**.
- Dado **m**, es fácil calcular **r(m)**.
- Dado **r(m)**, es computacionalmente intratable recuperar **m**.
- Dado **m**, es computacionalmente intratable obtener un **m'** tal que **r(m) = r(m')**.

Funciones

- **MDC:** Message Digest Code, funciones que dan como resultado bloques de longitud fija **a** a partir de bloques de longitud fija **b**, con **a < b**.
- **MAC:** Message Authentication Code, adiciona criptografía al proceso de hash para aumentar la seguridad del mismo. Es como un hash pero requiere una contraseña que sea la misma que se utilizo para generar dicho hash. Esta misma se debe compartir por un canal seguro.
  Tipos de implementaciones:
  (a) Basados en cifrados por bloques: cifrar el mensaje en algoritmo de bloques en modo de operación CBC. El valor del MAC será el resultado de cifrar el ultimo bloque del mensaje.
  (b) HMAC: basados en el uso de función MDC, aplicada sobre una versión del mensaje a la que se ha añadido un conjunto de bits, calculados a partir de la clave que se quiere emplear.
  (c) Basados en generadores de secuencia: empleando un generador de secuencia pseudoaleatorio el mensaje se parte en dos subcadenas cada una de las cuales alimenta un registro de desplazamiento retroalimentado. El valor del MAC es el estado final de ambos registros.
- **MD4:** Message Digest, produce un valor hash de 128 bits. Manipulacion de bits para obtener el valor de hash. Estándar de Internet (RFC-1320)
- **MD5:** Extension a MD4. La obtención del valor hash es lento pero considerado mas seguro. Estándar de Internet (RFC-1321)
- **SHA-1:** Secure Hash Algorithm, produce un valor hash de 160 bits. Estándar (FIPS PUB 180-1)
- **SHA-2:** Conjunto de algoritmos comprendidos por **SHA-224, SHA-256, SHA-384, SHA-512.**
- **SHA-3:** Concurso abierto organizado por el NIST.
- **RIPEMD-160:** Race Integrity Primitives Evaluation. Genera una salida de 160 bits.

Funciones de Derivación de Claves (KDF)

Funciones no reversibles que tienen el objetivo de generar una o mas claves en base a un valor maestro o clave inicial secretos, mas un conjunto de parámetros que configurar el comportamiento de la función afectando el resultado.

- **PBKDF2**
- **HKDF**
- **Bcrypt**
- **Argon2**
- **Script**

3. Asimétricos

Utiliza dos claves diferentes pero relacionadas matemáticamente, una es publica y la otra privada. La primera se entrega a cualquier persona y la segunda solamente la conoce el propietario. El remitente usa la clave publica del destinatario para cifrar el mensaje y este solamente podrá ser descifrados por el destinatario que posee la clave privada.

$$K = z^{xy} \pmod{p}$$

Ejemplos

- **Diffie-Hellman**
- **DSA**
- **RSA**
- **ECC**
- **ElGamal**

Ventajas

- No requiere confidencialidad en la distribución de la clave
- La misma clave puede ser utilizada por múltiples actores en la comunicación
- Permite autentificar mensajes

Desventajas

- Velocidad de cifrado/descifrado
- Longitud de mensaje limitado
- Tamaño del mensaje cifrado es mayor
- Se requieren claves de gran extensión

a) Cifrado

Disponen de las siguientes operaciones:

Generación de claves, cifrado, descifrado, firma y verificación de firma.

b) Autentificación (Firma)

Algunos algoritmos asimétricos permiten autenticar un mensaje para garantizar su integridad. En este caso se utiliza la clave privada.

c) Ejemplos de cifrados asimétricos

- **RSA:** Se basa en la dificultad para factorizar grandes números.
  Para su cifrado es, la expresión es:

$$c = m^e \pmod{n}$$

  Para su descifrado:

$$m = c^d \pmod{n}$$

  Se utiliza también para la generación y verificación de firmas digitales. Se aplica una operación utilizando la clave privada para firmar un mensaje y la clave publica para verificar la autenticidad de la firma.

  Cifrado:

$$s = h^d \pmod{n}$$

  Descifrado:

$$h = s^e \pmod{n}$$

- **ElGamal:** Se basa en el problema de los logaritmos discretos. Se utiliza tanto para el cifrado y descifrado de datos como para verificación de firmas digitales. Tiende a ser mas lento que el RSA debido a los cálculos que conlleva su implementación.

  Cifrado:

$$a = p^k \pmod{n}$$
$$b = y^k m \pmod{n}$$

  Descifrado:

$$m = b \cdot a^{-x} \pmod{n}$$

  Cifrado de firmas:

$$a = p^k \pmod{n}$$
$$b = (m - xa)k^{-1} \pmod{(n-1)}$$

  Descifrado de firmas:

$$y^a a^b = p^m \pmod{n}$$

- **DSA:** Digital Signature Algorithm, es una variante del método asimétrico de ElGamal. Utilizado para garantizar la autenticidad, integridad y no repudio de un mensaje o documento digital. Se basa en la teoría de los números y utiliza operaciones matemáticas relacionadas con el problema de logaritmo discreto en un grupo cíclico finito. A diferencia de los anteriores, se utiliza específicamente

para la generación y verificación de firmas digitales y no para el cifrado y descifrado de datos.

Generación de Firma:

1. Seleccionar un número aleatorio **k** tal que **$0 < k < q$**.
2. Calcular **$r = (\alpha^k \bmod p) \bmod q$**.
3. Calcular **$k^{-1} \bmod q$**.
4. Calculate **$s = k^{-1} (h + ar) \bmod q$**.
5. La firma del mensaje **m** es el par **(r, s)**.

Verificación de la Firma:

1. Verificar que **$0 < r < q$** y **$0 < s < q$**. En caso contrario, rechazar la firma.
2. Calcular el valor de **h** a partir de **m**.
3. Calcular **$\omega = s^{-1} \bmod q$**.
4. Calcular **$u1 = \omega \cdot h \bmod q$** y **$u2 = \omega \cdot r \bmod q$**.
5. Calcular **$v = (\alpha^{u1} y^{u2} \bmod p) \bmod q$**.
6. Aceptar la firma si y solo si **$v = r$**.

Algoritmos públicos y privados

- **Públicos:** aquellos cuya definición y funcionamiento se ponen a disposición publica, permitiendo que cualquier persona o entidad acceda al mismo para su evaluación o investigación.
- **Privados:** aquellos cuyo funcionamiento interno es desconocido. Son considerados menos confiables en lo relacionado a la criptografia.

Protocolo HTTP, HTTPS

- **Http:** (Hyper Text Transfer Protocol), protocolo para transmisión de información en plano, sin cifrado. Puerto por defecto: 80.
- **Https:** (Hyper Text Transfer Protocol Secure), protocol para transmision de información cifrada mediante SSL o TLS. Puerto por defecto: 443.

SSL (Secure Sockets Layer)

Protocolo que proporciona privacidad e integridad entre dos aplicaciones. El sistema SSL es independiente del protocolo utilizado (HTTP, FTP, POP, IMAP). Capa adicional que permite garantizar la seguridad de los datos. Se ubica entre la capa de aplicación y transporte.

- **Funcionamiento:** Los datos que circulan en un sentido y otro entre cliente y servidor se cifran mediante un algoritmo simétrico (DES o RC4). Un algoritmo de clave publica (generalmente RCA), se utiliza para el intercambio de claves de cifrado y para firmas digitales. El algoritmo utiliza la clave publica en el certificado digital del servidor, con el cual el cliente también puede verificar, por ese certificado, la identidad del servidor.
- **Fases:** En primer lugar, un establecimiento de la conexión y negociación de los algoritmos criptográficos que van a ser utilizados en la comunicación.

En segundo lugar, el intercambio de claves empleando algún mecanismo de clave publica y autentificación de los interlocutores a partir de sus certificados digitales. Y, por último, el cifrado simétrico de tráfico.

TLS (Transport Layer Security)

Protocolo mediante el cual se establece una conexión segura por medio de un canal cifrado entre cliente y servidor.

- **Características:**
  1. Incompatible con SSL v3.0
  2. Uso de funciones MAC en lugar de funciones MDC
  3. Numeración secuencial de todos los campos que componen la comunicación.
  4. Protección frente a ataques que intenta forzar el empleo de versiones antiguas del protocolo o cifrados mas débiles.
  5. El mensaje que finaliza la fase de establecimiento de la conexión incorpora una signatura de todos los datos intercambiados por ambos interlocutores.
- **Algoritmos Utilizados:**
  1. Asimétrico

     - **Diffie-Hellman (DHE)**
     - **RSA**
     - **DSA**
     - **ECDHE**

  2. Simétrico:

     - **RC2**
     - **RC4**
     - **IDEA**
     - **DES**
     - **Triple DES o AES**

  3. Funciones de Hash:

     - **MD5**
     - **Familia SHA**

- **Versiones:**
  RFC 2246 (1999)
  RFC 4346 (2006)
  RFC 5246 (2008)
  RFC 8446 (2018)

Firma Electrónica

Resultado de aplicar a un documento digital un procedimiento matemático que requiere información de exclusivo conocimiento del firmante, encontrándose esta bajo absoluto control. La firma debe ser susceptible de verificación por terceras partes para así detectar al firmante y posibles alteraciones.

- **Propiedades:**
  1. Ligada al mensaje. Valida únicamente para un documento.
  2. Solo puede ser generada por su legitimo titular.
  3. Públicamente verificable.

Modelos de Infraestructura de Seguridad

PKI – Infraestructura de Clave Publica

Combinación de hardware, software, políticas y procedimientos de seguridad que define un entorno de confianza centralizado y provee garantías para operaciones criptográficas como el certificado.

- **Componentes:**
  1. **Autoridad de Certificacion**
  2. **Autoridad de Registro**
  3. **Autoridad de Validacion**
  4. **Autoridad de Sellado de Tiempo**
  5. **Los Repositorios**
  6. **Los Usuarios y Entidades finales**

Certificados Digitales

Documento digital firmado digitalmente por un certificador, que vincula los datos de verificación de firma a su titular.

Certificados Digitales de revocación

Cuando una clave publica pierde su validez, es necesario anularla. Para ello se emplean este tipo de certificados que no son mas que un mensaje que identifica a la clave publica que se desea anular, firmada por la clave privada correspondiente.

Online Certificate Status Protocol

OCSP: método para determinar el estado de vigencia de un certificado digital X.509 usando otros medios que no sean el uso de CRL.

Anillo o Circulo de Confianza

Modelo de confianza distribuido que provee garantías para operaciones criptográficas como el cifrado, la firma, etc de transacciones electrónicas basado en la cantidad de firmas de actores de confianza que posea una clave pública.

SEGURIDAD

Clasificación

- **Sistemas Aislados:** Son los que no están conectados a ningún tipo de red.
- **Sistemas Interconectados:** Conectados por medio de la red, permitiendo recolectar información externa casi constantemente.
- **Seguridad Física:** Salvaguarda de los soportes físicos de la información (medidas contra incendios, sobrecargas eléctricas, etc).
- **Seguridad de la Información:** Preservación de la información frente a observadores no autorizados. Se puede emplear tanto criptografía simétrica como asimétrica, solamente simétrica para los sistemas aislados.
- **Seguridad del Canal de Comunicación:** No son considerados seguros debido a que la mayoría de los casos pertenecen a terceros y resulta imposible asegurar totalmente de que no están siendo escuchados y/o intervenidos.
- **Problemas de Autentificación:** Se suele emplear criptografía asimétrica con funciones de resumen (hash) para asegurarse que la información proviene realmente de donde deseamos.
- **Problemas de Suplantación:** Se emplean mecanismos basados en contraseñas ya que los usuarios pueden ingresar al sistema desde afuera y se requiere saber si realmente es el usuario autorizado.
- **No Repudio:** Cuando se recibe un mensaje, no solo es necesario poder identificar a la fuente, sino que también, este asuma todas las responsabilidades derivadas de la información que haya enviado y no poder negar su autoría.
- **Anonimato:** En ciertos casos es primordial garantizar el anonimato del usuario para poder preservar su intimidad y libertad.

Tipos de Autentificación

- **Autentificación de mensaje:** Se conoce como firma digital y lo que permite es garantizar la procedencia de un mensaje conocido y así evitar su falsificación.
- **Autenticación de usuario mediante contraseña:** Se trata de garantizar la presencia de un usuario legal en el sistema. El usuario debe poseer una contraseña que le permite ingresar al sistema y/o identificarse.
- **Autenticación de dispositivo:** Puede tratarse de una llave electrónica que sustituye a la contraseña para identificar a un usuario.

CERTIFICADOS DIGITALES

OpenSSL

Proyecto de software libre basado en SSLeay. Consta de herramientas y bibliotecas criptográficas que asisten a implementaciones de sistemas de seguridad como SSL, TLS y SSH. También puede ser utilizado para generar certificados en servidores como Apache y Tomcat.

Generador Clave RSA

```
openssl genrsa -des3 -out server.key 1024
```

Generar CSR (Certificate Signing Request)

```
openssl req -new -key server.key -out server.csr
```

Generar un certificado "Self-Signed"

```
openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
```

Remover contraseña de una clave RSA

```
cp server.key server.key.orig
openssl rsa -in server.key.orig -out server.key
```

TLS en Apache Tomcat

Dos implementaciones diferentes para utilizar SSL:

- **Implementación JSSE** (Java 1.4)
- **Implementación APR** (OpenSSL)