



**INSTITUTO POLITÉCNICO NACIONAL**  
**Escuela Superior de Cómputo**

**ESCOM**

*Trabajo Terminal*

**“Protocolo criptográfico para el almacenamiento  
sin duplicados en la nube, resistente a ataques  
por fuerza bruta.”**

*2016-B045*

*Presentan*

**Eder Jonathan Aguirre Cruz**  
**Diana Leslie González Olivier**  
**Jhonatan Saulés Cortés**

*Directora*

**Dra. Sandra Díaz Santiago**

INSTITUTO POLITÉCNICO NACIONAL



**ESCOM**

**Mayo 2017**

# Índice

<b>1. Introducción</b>	<b>1</b>
1.1. Justificación . . . . .	1
1.2. Objetivos . . . . .	1
1.2.1. Objetivos Generales . . . . .	1
1.2.2. Objetivos Específicos . . . . .	1
<b>2. Preliminares</b>	<b>2</b>
2.1. Criptografía . . . . .	2
2.1.1. Criptoanálisis . . . . .	2
2.1.2. Esteganografía . . . . .	2
2.1.3. Servicios de Seguridad . . . . .	2
2.1.4. CONFIDENCIALIDAD . . . . .	2
2.1.5. AUTENTICACIÓN . . . . .	3
2.1.6. INTEGRIDAD . . . . .	3
2.1.7. NO REPUDIO . . . . .	3
2.1.8. CONTROL DE ACCESO . . . . .	4
2.1.9. DISPONIBILIDAD . . . . .	4
2.1.10. Ataques . . . . .	4
2.1.11. Ataques a los Métodos de Cifrado . . . . .	5
2.1.12. ATAQUE SÓLO CON TEXTO CIFRADO . . . . .	5
2.1.13. ATAQUE CON TEXTO ORIGINAL CONOCIDO . . . . .	5
2.1.14. ATAQUE CON TEXTO CIFRADO ESCOGIDO . . . . .	5
2.1.15. ATAQUE CON TEXTO ESCOGIDO . . . . .	5
2.1.16. Ataques a los Protocolos Criptográficos . . . . .	5
2.1.17. ATAQUE CON CLAVE CONOCIDA . . . . .	5
2.1.18. SUPLANTACIÓN DE PERSONALIDAD . . . . .	5
2.1.19. COMPILACIÓN DE UN DICCIONARIO . . . . .	6
2.1.20. BÚSQUEDA EXHAUSTIVA . . . . .	6
2.1.21. ATAQUE DE HOMBRE EN MEDIO . . . . .	6
2.1.22. Ataques en criptoanálisis . . . . .	6
2.1.23. Criptografía Simétrica . . . . .	7
2.1.24. Criptografía asimétrica . . . . .	8
2.1.25. Cifrado por bloques . . . . .	9
2.1.26. Modos de operación . . . . .	10
2.1.27. Funciones Hash . . . . .	14
<b>3. Adversarios Clasificadores</b>	<b>15</b>
<b>4. Tecnologías usadas</b>	<b>16</b>
<b>5. Análisis y Diseño</b>	<b>17</b>
5.1. Diagramas de caso de uso . . . . .	17
<b>6. Desarrollo de prototipos</b>	<b>18</b>
6.1. Prototipo 1 . . . . .	18

<b>7. Conclusiones y Trabajo a Futuro</b>	<b>19</b>
7.1. Conclusiones . . . . .	19
<b>A. Código fuente del prototipo 2</b>	<b>20</b>
<b>Bibliografía</b>	<b>21</b>

# Índice de Figuras

2.1. Diagrama Criptografía Simétrica . . . . .	7
2.2. Diagrama Criptografía Asimétrica . . . . .	8

# Índice de Tablas

# Capítulo 1

## Introducción

### 1.1. Justificación

### 1.2. Objetivos

#### 1.2.1. Objetivos Generales

#### 1.2.2. Objetivos Específicos

# Capítulo 2

## Preliminares

### 2.1. Criptografía

Es la ciencia que trata las escrituras ocultas, está comprendida por la Criptografía, el Criptoanálisis y la Esteganografía. La Criptografía es la ciencia que se encarga del estudio de técnicas para transformar la información a una forma que no pueda entenderse a simple vista; sin embargo, el objetivo de la Criptografía no es sólo mantener los datos secretos, sino también protegerlos contra modificación y comprobar la fuente de los mismos.

#### 2.1.1. Criptoanálisis

Es la ciencia que se ocupa del análisis de un texto cifrado para obtener la información original sin conocimiento de la clave secreta, esto es, de forma ilícita rompiendo así los procedimientos de cifrado establecidos por la Criptografía, por lo que se dice que Criptoanálisis y Criptografía son ciencias complementarias pero contrarias. El criptoanálisis es el arte de descifrar comunicaciones encriptadas sin conocer las llaves correctas.

#### 2.1.2. Esteganografía

Por su parte, estudia la forma de ocultar la existencia de un mensaje. Esta ciencia consiste en esconder en el interior de un mensaje, otro mensaje secreto, el cual sólo podrá ser entendido por el emisor y el receptor y pasará inadvertido para todos los demás.

#### 2.1.3. Servicios de Seguridad

Los servicios de seguridad, son aquellos que garantizan en un sistema de información la adquisición, almacenamiento, procesamiento y transmisión de la información y para lograrlo se valen de uno o más mecanismos de seguridad.

#### 2.1.4. CONFIDENCIALIDAD

Este servicio asegura que sólo las personas o procesos autorizados tengan acceso a la información. Con ello se busca que un agente no autorizado no pueda leer, copiar o modificar la información. El servicio de confidencialidad se puede diferenciar en dos tipos:

- Servicio de confidencialidad de contenido: se busca proteger el contenido de un recurso del sistema, para ello se cifra la información para que en caso ser interceptada por alguien no autorizado, no pueda ser descubierta. Este servicio puede proporcionar protección a todos los datos transmitidos por un usuario durante una conexión o puede proteger sólo parte de ellos por ejemplo sólo a los mensajes con información importante o incluso se pueden proteger sólo algunos campos de un determinado mensaje.
- Servicio de confidencialidad del mensaje: busca ocultar el flujo de un mensaje durante una conexión, para ello se cifra y se utiliza una técnica de envoltura con el objetivo de que si un atacante está realizando un análisis de tráfico, no pueda descubrir por ejemplo quien está enviando la información ni quien la recibe ni la frecuencia con la que se envían los mensajes.

### **2.1.5. AUTENTICACIÓN**

Este servicio verifica la identidad de un agente que pretende acceder a la información. En una conexión entre dos entidades, el servicio verifica que las entidades sean quienes dicen ser, además de asegurar que un tercer individuo no pueda hacerse pasar por alguna de las entidades autorizadas y realizar una transmisión o recepción de datos.

### **2.1.6. INTEGRIDAD**

Este servicio asegura que el contenido de los datos no ha sido modificado y que la secuencia de los mismos se ha mantenido a lo largo de toda la transmisión, con ello se evita una réplica o un reordenamiento del mensaje por parte de un atacante. Al igual que el servicio de confidencialidad, la integridad puede aplicarse a todos los datos transmitidos por un usuario durante una conexión, sólo a parte de ellos o sólo a algunos campos dentro del mensaje. Cuando se tiene un ataque a la integridad de los datos, el sistema puede o no reportar dicha violación, por lo que se puede distinguir entre servicio de integridad con recuperación y servicio de integridad sin recuperación. El servicio de integridad también se puede diferenciar entre servicio de integridad del contenido y servicio de integridad de la secuencia del mensaje

- Servicio de integridad del contenido: proporciona pruebas de que el contenido no ha sido alterado o modificado.
- Servicio de integridad de la secuencia del mensaje: proporciona pruebas de que el orden de una secuencia de mensajes ha sido mantenida durante la transmisión.

### **2.1.7. NO REPUDIO**

Este servicio evita que las entidades en una conexión nieguen haber transmitido o recibido un mensaje. Existen varios tipos de este servicio y cada uno de ellos proporciona pruebas de haberse llevado a cabo:

- No repudio de origen: con este servicio, el emisor de un mensaje no puede negar haber sido él quien transmitió dicho mensaje.
- No repudio de envío: comprueba que los datos fueron enviados.



- No repudio de presentación: protege contra cualquier intento falso de negar que los datos fueron presentados para el envío.
- No repudio de transporte: protege contra cualquier intento de negar que los datos fueron transportados.
- No repudio de recepción: con este servicio, el receptor de un mensaje no puede negar haber recibido un mensaje.

### 2.1.8. CONTROL DE ACCESO

El servicio de control de acceso es utilizado con el fin de restringir el acceso a los medios de almacenamiento de la información. Este servicio está muy relacionado con el de autenticación ya que cualquier agente que quiera tener acceso a algún recurso del sistema primero deberá identificarse para que le sea permitido el acceso a dicha información y de acuerdo a los permisos o privilegios que tenga podrá manipularla.

### 2.1.9. DISPONIBILIDAD

El servicio de disponibilidad asegura que los agentes autorizados tengan acceso a la información en el momento en que ellos lo requieran y tantas veces como lo soliciten sin importar si ésta es correcta o no.

### 2.1.10. Ataques

Un ataque es una violación a la seguridad de la información realizada por intrusos que tienen acceso físico al sistema sin ningún tipo de restricción, su objetivo es robar la información o hacer que ésta pierda valor relativo, o que disminuyan las posibilidades de su supervivencia a largo plazo. Un intruso puede obtener información como:

- Bloques de direcciones IP
- Localización de sistemas críticos (DNSs, WINS, DHCPs, Servidores de correo, etc.)
- Puntos de acceso para números telefónicos y VPNs
- Información personal de los trabajadores de la organización
- Organizaciones asociadas, subsidiarias, etc.

Existen dos tipos de ataques que amenazan las comunicaciones secretas:

- Pasivo: es aquel en el cual el intruso sólo busca obtener la información y al hacerlo no la modifica, por lo que es difícil percatarse de que se está siendo atacado.
- Activo: el intruso además de obtener la información la modifica de tal modo que sirva a sus intereses y al ser modificada es más fácil percatarse de que se está siendo atacado. Los ataques activos se dividen en dos tipos: Ataques a los métodos de cifrado y Ataques a los protocolos criptográficos.

### **2.1.11. Ataques a los Métodos de Cifrado**

Este tipo de ataques se realizan con la intención de obtener la clave secreta para poder descifrar libremente cualquier criptograma, para ello se aprovechan las vulnerabilidades que pudiera tener el método de cifrado.

### **2.1.12. ATAQUE SÓLO CON TEXTO CIFRADO**

Este caso es cuando el criptoanalista sólo conoce el criptograma y el algoritmo con que fue generado; con esta información pretende obtener el texto en claro.

### **2.1.13. ATAQUE CON TEXTO ORIGINAL CONOCIDO**

En esta situación el criptoanalista conoce mensajes en claro seleccionados por él mismo y sus correspondientes criptogramas, así como el algoritmo con que éstos fueron generados; aquí el objetivo es conocer la clave secreta y poder descriptar libremente cualquier texto.

### **2.1.14. ATAQUE CON TEXTO CIFRADO ESCOGIDO**

El criptoanalista conoce el algoritmo de cifrado, así como un criptograma seleccionado por él mismo y su correspondiente texto en claro, su objetivo es obtener el mensaje en claro de todo criptograma que intercepte.

### **2.1.15. ATAQUE CON TEXTO ESCOGIDO**

En este caso el criptoanalista además de conocer el algoritmo de cifrado y el criptograma que quiere descriptar, también conoce el criptograma de un texto en claro que él elija y el mensaje en claro de un criptograma también elegido por él.

### **2.1.16. Ataques a los Protocolos Criptográficos**

Este tipo de ataques no pretenden encontrar la clave secreta para poder conocer el mensaje en claro, sino que buscan obtener la información vulnerando los protocolos criptográficos, es decir, pretenden burlar la serie de pasos establecidos para alcanzar los objetivos de seguridad y que tienen que ser realizados por las entidades involucradas en cierta comunicación. Ejemplos de este tipo de ataques son los siguientes:

### **2.1.17. ATAQUE CON CLAVE CONOCIDA**

El atacante conoce claves utilizadas en cifrados anteriores y con base en ellas intenta determinar nuevas claves.

### **2.1.18. SUPLANTACIÓN DE PERSONALIDAD**

El atacante asume la identidad de uno de los agentes autorizados en la red, y de esta manera obtiene libremente y sin tropiezos todos los mensajes en claro.

### **2.1.19. COMPILACIÓN DE UN DICCIONARIO**

Un diccionario es un archivo guardado en la memoria de la computadora que contiene contraseñas cifradas de los usuarios autorizados en el sistema. Si el método de cifrado con que se cifran las claves es público, el atacante puede generar claves aleatorias y después cifrarlas con el objeto de encontrar alguna contenida en el diccionario (previamente obtenido). Cuando una clave generada por el atacante coincide con una contenida en el diccionario, se ha encontrado una clave de acceso al sistema, mediante el usuario correspondiente a la clave encontrada.

### **2.1.20. BÚSQUEDA EXHAUSTIVA**

Este ataque se lleva a cabo generando aleatoriamente todos los valores posibles de las claves de acceso y probándolas hasta que una de ellas sea una clave válida en el sistema.

### **2.1.21. ATAQUE DE HOMBRE EN MEDIO**

El intruso se filtra en la línea de comunicación entre dos agentes autorizados en la red; obtiene la información de uno de ellos y se la envía al otro usuario una vez que la ha utilizado.

### **2.1.22. Ataques en criptoanálisis**

Aunque para validar la robustez de un criptosistema normalmente se suponen todas las condiciones del peor caso, existen ataques más específicos, en los que no se cumplen todas estas condiciones. Cuando el método de ataque consiste simplemente en probar todas y cada una de las posibles claves del espacio de claves hasta encontrar la correcta, nos encontramos ante un ataque de fuerza bruta o ataque exhaustivo. Si el atacante conoce el algoritmo de cifrado y sólo tiene acceso al criptograma, se plantea un ataque sólo al criptograma; un caso más favorable para el criptoanalista se produce cuando el ataque cumple todas las condiciones del peor caso; en este caso, el criptoanálisis se denomina de texto en claro conocido. Si además el atacante puede cifrar una cantidad indeterminada de texto en claro al ataque se le denomina de texto en claro escogido; este es el caso habitual de los ataques contra el sistema de verificación de usuarios utilizado por Unix, donde un intruso consigue la tabla de contraseñas (generalmente /etc/passwd) y se limita a realizar cifrados de textos en claro de su elección y a comparar los resultados con las claves cifradas (a este ataque también se le llama de diccionario, debido a que el atacante suele utilizar un fichero ‘diccionario’ con los textos en claro que va a utilizar). El caso más favorable para un analista se produce cuando puede obtener el texto en claro correspondiente a criptogramas de su elección; en este caso el ataque se denomina de texto cifrado escogido.

Cualquier algoritmo de cifrado, para ser considerado seguro, ha de soportar todos estos ataques y otros no citados; sin embargo, en la criptografía, como en cualquier aspecto de la seguridad, informática o no, no debemos olvidar un factor muy importante: las personas. El sistema más robusto caerá fácilmente si se tortura al emisor o al receptor hasta que desvelen el contenido del mensaje, o si se le ofrece a uno de ellos una gran cantidad de dinero; este tipo de ataques (sobornos, amenazas, extorsión, tortura...) se consideran siempre los más efectivos.

### 2.1.23. Criptografía Simétrica

La criptografía simétrica utiliza la misma clave para cifrar y descifrar el mensaje de datos, es decir se basa en un secreto compartido [?].

Características de la Criptografía simétrica:

- La clave es la misma para cifrar que para descifrar un mensaje, por lo que sólo el emisor y el receptor deben conocerla.
- Se basan en operaciones matemáticas sencillas, por ello son fácilmente implementados en hardware.
- Debido a su simplicidad matemática son capaces de cifrar grandes cantidades de datos en poco tiempo.

[1]

Los algoritmos criptográficos simétricos tienen dos versiones: cifrador en bloque y cifrador en flujo. Una cifra es una palabra para describir un algoritmo de cifrado. El beneficio del uso de un algoritmo simétrico radica en el procesamiento rápido para encriptar y desencriptar un alto volumen de datos. El cifrado simétrico es una eficaz táctica de almacenamiento de información sensible en una base de datos, un registro o archivo [1] El cifrado simétrico puede ser representado con el siguiente diagrama 2.2.

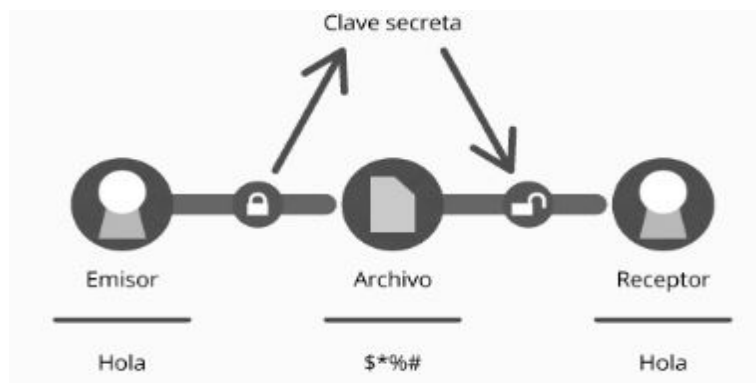


Figura 2.1: Diagrama Criptografía Simétrica

La sintaxis de un esquema de cifrado simétrico, esta dada por la siguiente definición.

**Definición 2.1** *Un esquema de cifrado simétrico está conformado por una tripleta de algoritmos  $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ , definidos como se describe a continuación:*

- *El algoritmo generador de claves **Gen** selecciona una llave  $K$  al azar del conjunto de llaves  $\mathcal{K}$ , esto se denotará como  $K \xleftarrow{\$} \mathcal{K}$ . Esta llave  $K$  será usada por los algoritmos **Enc** y **Dec**, esta llave la compartirán emisor y receptor.*
- *El algoritmo de cifrado **Enc**, toma como entrada un texto en claro  $M \in \mathcal{M}$  y una llave  $K$  generada por **Gen** y regresa un texto cifrado  $C \in \mathcal{C}$ . Usualmente esto se denota como  $C \leftarrow \text{Enc}_K(M)$ .*

- El algoritmo de descifrado  $\text{Dec}$ , toma como entrada un texto cifrado  $C$  y una llave  $K$  y regresa  $M$ . Esta operación se denota por  $M \leftarrow \text{Dec}_K(C)$ . Para que cualquier algoritmo de cifrado simétrico funcione correctamente, se debe garantizar que para todas las llaves posibles en  $\mathcal{K}$  y todos los posibles mensajes  $\mathcal{M}$ ,

$$\text{Dec}_K(\text{Enc}_K(M)) = M.$$

### 2.1.24. Criptografía asimétrica

La criptografía asimétrica es por definición aquella que utiliza dos claves diferentes para cada usuario, una para cifrar que se le llama clave pública y otra para descifrar que es la clave privada. Los algoritmos asimétricos son diferentes a los simétricos en un sentido muy importante [1]. Cuando se genera una clave simétrica, simplemente se escoge un número aleatorio de la longitud apropiada. Al generar claves asimétricas el proceso es más complejo. Los algoritmos asimétricos se llaman asimétricos porque en lugar de usar una sola clave para realizar la codificación y la decodificación, se utilizan dos claves diferentes: una para cifrar y otra para descifrar. Estas dos claves se encuentran asociadas matemáticamente, cuya característica fundamental es que una clave no puede descifrar lo que cifra. [1].

Características de la Criptografía simétrica:

- Se utiliza una clave para cifrar y otra para descifrar. El emisor emplea la clave pública del receptor para cifrar el mensaje, éste último lo descifra con su clave privada.
- Se basan en operaciones matemáticas complejas.
- Se ejecutan de 100 a 1000 veces más lento que los algoritmos simétricos.

[1]

Los beneficios de la criptografía asimétrica son la solución a los problemas de la criptografía simétrica, pues las claves públicas pueden ser distribuidas con toda tranquilidad, no valen de nada sin las claves privadas. El cifrado asimétrico se le emplea muy frecuente para pasar con seguridad una clave privada, que posteriormente, será la que se utilice para cifrar y/o descifrar otra información. El cifrado asimétrico puede ser representado con el siguiente diagrama ??.

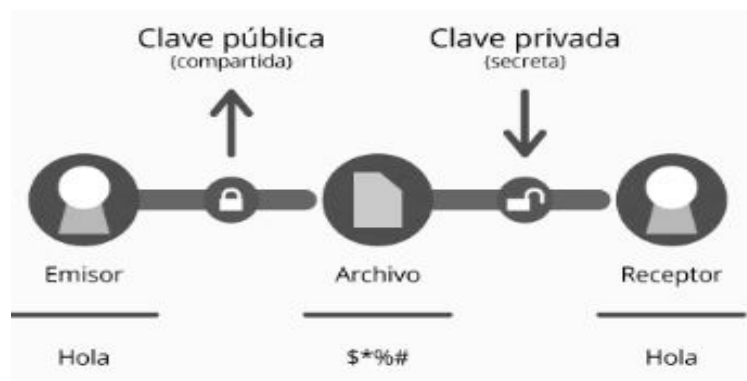


Figura 2.2: Diagrama Criptografía Asimétrica

### 2.1.25. Cifrado por bloques

Los algoritmos de cifrado por bloques toman bloques de tamaño fijo del texto en claro y producen un bloque de tamaño fijo de texto cifrado, generalmente del mismo tamaño que la entrada. El tamaño del bloque debe ser lo suficientemente grande como para evitar ataques de texto cifrado. La asignación de bloques de entrada a bloques de salida debe ser uno a uno para hacer el proceso reversible y parecer aleatoria.

Para la asignación de bloques los algoritmos de cifrado simétrico realizan sustituciones y permutaciones en el texto en claro hasta obtener el texto cifrado.

La sustitución es el reemplazo de un valor de entrada por otro de los posibles valores de salida, en general, si usamos un tamaño de bloque  $k$ , el bloque de entrada puede ser sustituido por cualquiera de los  $2^k$  bloques posibles. La permutación es un tipo especial de sustitución en el que los bits de un bloque de entrada son reordenados para producir el bloque cifrado, de este modo se preservan las estadísticas del bloque de entrada (el número de unos y ceros).

Los algoritmos de cifrado por bloques iterativos funcionan aplicando en sucesivas rotaciones una transformación (función de rotación) a un bloque de texto en claro. La misma función es aplicada a los datos usando una subclave obtenida de la clave secreta proporcionada por el usuario. El número de rotaciones en un algoritmo de cifrado por bloques iterativo depende del nivel de seguridad deseado.

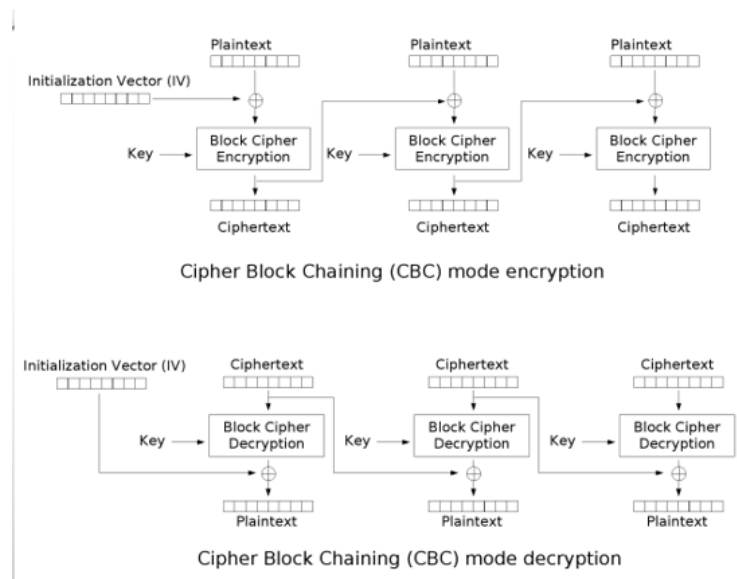
La sustitución es el reemplazo de un bloque de  $n$  bits por otro bloque de  $n$  bits en un espacio de  $2^k$  [11]. Los cifradores por bloques mas usados son AES (Advanced Encryption Standard, por sus siglas en inglés) y DES (Data Encryption Standard, por sus siglas en inglés).

### 2.1.26. Modos de operación

Un modo de operación es una técnica para mejorar el efecto de un algoritmo criptográfico o adaptar el algoritmo para una aplicación, tal como aplicar un cifrador por bloques a una secuencia de bloques de datos o un flujo de datos. Los cuatro modos están destinados a cubrir virtualmente todas las aplicaciones posibles de cifrado para las cuales se podría usar un cifrador por bloques. A medida que han aparecido nuevas aplicaciones y requisitos, el NIST ha ampliado la lista de modos recomendados a cinco en la Publicación Especial 800-38A. Estos modos están diseñados para usarse con cualquier cifra simétrica de bloques, incluyendo DES triple y AES..

*CBC*(Cipher-block chaining): La entrada al algoritmo de encriptación es el XOR de los siguientes 64 bits de texto plano y los 64 bits de cifrado anteriores.

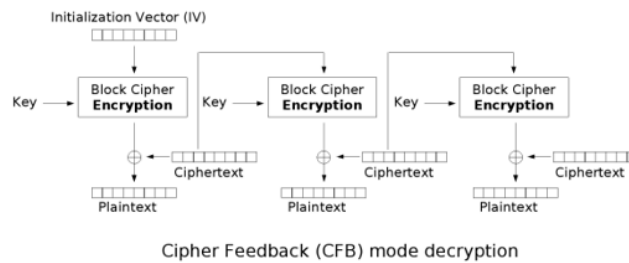
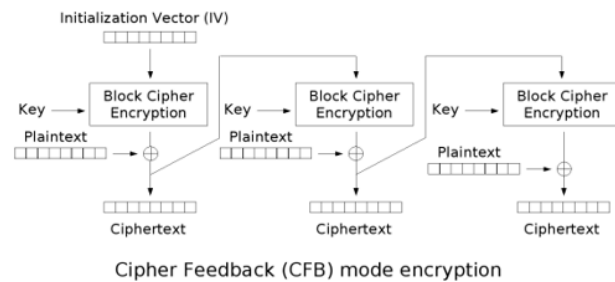
- La salida de uno de los bloques de cifrado se mete a otro bloque de cifrado junto con el siguiente bloque de mensaje.
- Toma como entradas un vector de inicialización (IV) y un bloque de mensaje (m).
- Durante el encriptado la salida del i - ésimo bloque depende del anterior i -1 bloques.
- La salida de cada uno de los bloques depende de todo lo anterior y esto lo hace mas seguro que ECB.
- El descifrado de CBC es no secuencial.



(a) Diagrama CBC Cifrado

*CFB*(Cipher Feedback): La entrada se procesa j bits a la vez. El texto cifrado precedente se utiliza como entrada al algoritmo de cifrado para producir la salida pseudoaleatoria, que se le aplica XOR con el texto sin formato para producir la siguiente unidad de texto cifrado.

- Los bloques de cifrado también están encadenados pero la salida es muy diferente a los demás.
- Para cada bloque, el cifrado es producido haciendo XOR con el mensaje.
- Una ventaja de implementación es que no es necesaria la operación de descifrar no es necesario.

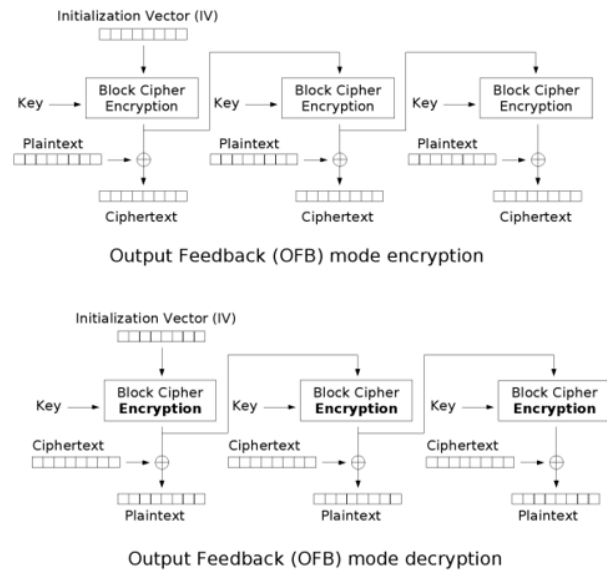


(a) Diagrama CFB Cifrado



*OFB*(Output feedback): Similar a CFB, excepto que la entrada al algoritmo de cifrado es la salida DES anterior.

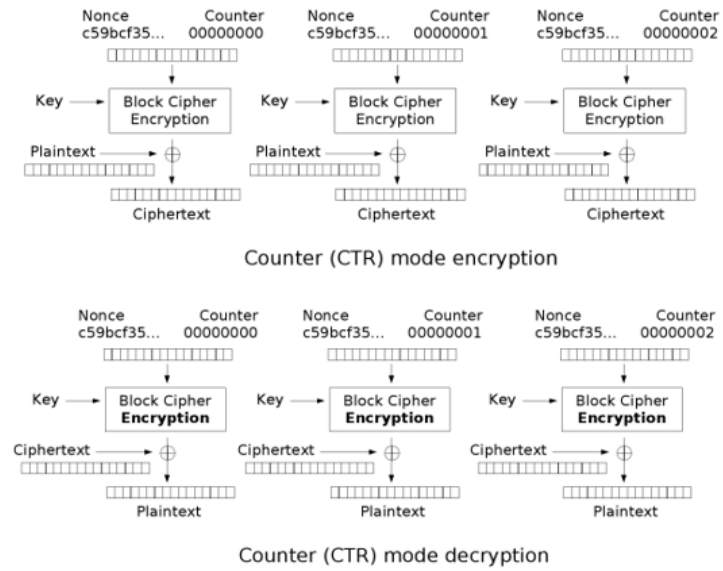
- En OFB la salida del bloque de cifrado es alimentada de nuevo en la siguiente bloque de cifrado.
- El IV es cifrado varias veces para obtener una corriente de bytes aleatorios.
- Estas corrientes de bytes aleatorios se les hace XOR con el texto en plano para generar el texto cifrado.



(a) Diagrama OFB Cifrado

*CTR*(Counter): Cada bloque de texto sin formato se le aplica XOR con un contador cifrado. El contador se incrementa para cada bloque subsiguiente.

- CTR toma un vector de inicialización (IV) y en cada iteración el valor de IV se incrementa en 1 y queda cifrado.
- Para obtener el mensaje cifrado se hace una XOR con el IV y el bloque de mensaje.
- En términos de eficiencia CTR es mejor que CBC, OFB o CFB, ya que en este modo se pueden hacer las operaciones en paralelo ya que no dependen de algo para poder ser cifradas.



(a) Diagrama CTR Cifrado

## 2.1.27. Funciones Hash

A continuación se describirán las características de las *funciones hash*, también conocidas como *funciones de resumen*. Las funciones hash basan su definición en funciones de un solo sentido (*one-way functions*, en inglés). Una función de un sólo sentido es aquella que para un valor  $x$ , es muy fácil calcular  $f(x)$ , pero es muy difícil hallar  $f^{-1}(x)$ . Es complicado en general, hallar funciones de éste tipo y probar que lo son.

**Definición 2.2** *Una función hash, es una función de un sólo sentido cuya entrada  $m$  es un mensaje de longitud arbitraria y la salida es una cadena binaria de longitud fija. Al resumen o hash de un mensaje  $m$ , se le denotará como  $h(m)$ . Una función hash debe tener las siguientes propiedades:*

- *Para cualquier mensaje  $m$ , debe ser posible calcular  $h(m)$  eficientemente.*
- *Dado  $h(m)$ , debe ser computacionalmente difícil, hallar un mensaje  $m'$ , tal que  $h(m) = h(m')$ .*
- *Debe ser computacionalmente difícil, hallar dos mensajes  $m$  y  $m'$  tales que  $h(m) = h(m')$ .*

Entre las funciones hash que se usan para criptografía están: MD2, MD4, MD5, donde MD significa *Message Digest*, y el algoritmo estándar al momento de escribir éstas notas es el *Secure Hash Algorithm* por sus siglas en inglés SHA. La MD5 fue diseñada por Ron Rivest, toma como entrada un mensaje de longitud arbitraria y proporciona como salida una cadena binaria de 128 bits. El mensaje de entrada se procesa por bloques de 512 bits. La SHA fue diseñada por en NIST y se estableció como estándar en 1993. Recibe como entrada un mensaje con longitud menor a  $2^{64}$  bits y como salida se obtiene una cadena binaria de 160 bits. Al igual que el MD5, se procesa en bloques de 512 bits [25].

# Capítulo 3

## Adversarios Clasificadores

# Capítulo 4

## Tecnologías usadas

# Capítulo 5

## Análisis y Diseño

### 5.1. Diagramas de caso de uso

# Capítulo 6

## Desarrollo de prototipos

### 6.1. Prototipo 1

# Capítulo 7

## Conclusiones y Trabajo a Futuro

### 7.1. Conclusiones



# Apéndice A

## Código fuente del prototipo 2

# Referencias

- [1] Cifrado simetrico. Guía de Gnu Privacy Guard, 2015. <https://www.gnupg.org/gph/es/manual/c190.html#AEN201>.
- [2] em client. eM Client web page, 2015. <http://www.emclient.com/>.
- [3] Opera mail. Opera Mail web pages, 2015. <http://www.opera.com/es-419/computer/mail>.
- [4] Post box. Post Box web page, 2015. <https://www.postbox-inc.com/>.
- [5] Thunderbird. Thunderbird web pages, 2015. <https://www.mozilla.org/es-ES/thunderbird/>.
- [6] Zimbra. Zimbra web pages, 2015. <https://www.zimbra.com/>.
- [7] R. Allenby. *Rings, fields, and groups: an introduction to abstract algebra*. E. Arnold, 1983.
- [8] Alonsojpd. Montar un servidor de correo electrónico mail en linux ubuntu. AJPDsoft, 2015. <http://www.ajpdsoft.com/modules.php?name=News&file=article&sid=506>.
- [9] T. Brehm. The perfect server - ubuntu 15.10 (wily werewolf) with apache, php, mysql, pureftpd, bind, postfix, dovecot and ispcconfig 3. How to Forge, 2015. <https://www.howtoforge.com/tutorial/ubuntu-perfect-server-with-apache-php-mysql-pureftpd-bind-postfix-doveot-and-ispc>
- [10] M. Brodsky. Reflexiones jurídicas sobre el e-marketing en Chile. Interactive Advertising Bureau, 2015. <http://www.iab.cl/reflexiones-juridicas-sobre-el-e-marketing-en-chile/>.
- [11] D. Chakraborty and F. Rodríguez-Henríquez. Block cipher modes of operation from a hardware implementation perspective. In Ç. K. Koç, editor, *Cryptographic Engineering*, pages 321–363. Springer, 2009.
- [12] S. Diaz-Santiago and D. Chakraborty. On securing communication from profilers. In P. Samarati, W. Lou, and J. Zhou, editors, *SECRYPT 2012 - Proceedings of the International Conference on Security and Cryptography, Rome, Italy, 24-27 July, 2012, SECRYPT is part of ICETE - The International Joint Conference on e-Business and Telecommunications*, pages 154–162. SciTePress, 2012.

- [13] P. Golle and A. Farahat. Defending email communication against profiling attacks. In *Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society, WPES 2004, Washington, DC, USA, October 28, 2004*, pages 39–40, 2004.
- [14] A. Gulbrandsen and N. Freed. Internet Message Access Protocol (IMAP) - MOVE Extension. RFC 6851, 2015.
- [15] D. Jurafsky and J. H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 2000.
- [16] D. J. C. Klensin. Simple Mail Transfer Protocol. RFC 5321, 2015.
- [17] J. Klensin. Simple Mail Transfer Protocol. RFC 2821 (Proposed Standard), April 2001. Obsoleted by RFC 5321, updated by RFC 5336.
- [18] W. Koch. The gnu privacy guard. GnuPG web page, 2016. <https://www.gnupg.org/index.html>.
- [19] D. P. Martínez. Postgresql vs. mysql. geekWare, 2015. <https://danielpecos.com/documents/postgresql-vs-mysql/>.
- [20] J. Myers and M. Rose. Post Office Protocol - Version 3. RFC 1939 (Standard), 1996. Updated by RFCs 1957, 2449.
- [21] J. Peralta. Anillos y cuerpos. Campus Virtual Univesidad de Almería, 2016. <http://www.ual.es/personal/jperalta/anilloscuerpos.pdf>.
- [22] N. Roshanbin and J. Miller. A survey and analysis of current captcha approaches. *J. Web Eng.*, 12(1-2):1–40, 2013.
- [23] sawiyati. How to install apache, php and mariadb on ubuntu 15.04. Server Mom, 2015. <http://www.servermom.org/install-apache-php-mariadb-ubuntu-15-04/2208/>.
- [24] A. Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, 1979.
- [25] W. Stallings. *Cryptography and Network Security: Principles and Practice*. Pearson Education, 5a edition, 2002.
- [26] D. R. Stinson. *Cryptography Theory and Practice*. Chapman & Hall/CRC, 3a edition, 2006.
- [27] O. Tezer. Sqlite vs mysql vs postgresql: A comparison of relational database management systems. Digital Ocean, 2014. <https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-syst>
- [28] M. R. S. Villanueva. Aritmética del reloj. Departamento de Matemáticas de la Universidad de Puerto Rico en Aguadilla, 2016. <http://math.uprag.edu/milena/4.5%20ARITMETICA%20DEL%20RELOJ.pdf>.

- [29] Wikipedia. Ciphertext-only attack — Wikipedia, the free encyclopedia, 2015. [https://en.wikipedia.org/wiki/Ciphertext-only\\_attack](https://en.wikipedia.org/wiki/Ciphertext-only_attack).
- [30] Wikipedia. Email — Wikipedia, the free encyclopedia, 2015. <http://en.wikipedia.org/wiki/Email>.
- [31] Wikipedia. Pretty good privacy — Wikipedia, the free encyclopedia, 2015. [https://es.wikipedia.org/wiki/Pretty\\_Good\\_Privacy](https://es.wikipedia.org/wiki/Pretty_Good_Privacy).
- [32] L. G. G. y Dr. Sergio Rajsbaum. Critografía. Temas selectos de la web, 2015. [http://www.matem.unam.mx/rajsbaum/cursos/web/presentacion\\_seguridad\\_1.pdf](http://www.matem.unam.mx/rajsbaum/cursos/web/presentacion_seguridad_1.pdf).
- [33] E. A. M. y M.C. Ma. Jaquelina López Barrientos. Fundamentos de critografía. Universidad Nacional Autonoma de México Facultad de Ingenieria, 2015. <http://redyseguridad.fi-p.unam.mx/proyectos/criptografia/criptografia/>.