**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR**


**FACULTAD DE INGENIERÍA**
**ESCUELA DE SISTEMAS**

**DISERTACION PREVIA A LA OBTENCION DEL TÍTULO DE INGENIERO EN SISTEMAS**

# DESARROLLO DE UNA APLICACIÓN PARA ENCRIPTAR INFORMACIÓN EN LA TRANSMISIÓN DE DATOS EN UN APLICATIVO DE MENSAJERIA WEB

**JOHANNA BEATRIZ MOYA CAZA**

**FRANKLIN ANDRÉS ESCOBAR ERAZO**

**QUITO, ABRIL 2015**

# DEDICATORIA

A mí amado Dios por estar a mi lado siempre y permitirme culminar esta meta.

A mis papitos Martha Caza y Carlos Moya por su apoyo incondicional, a mi hermanita Carlita Moya, a mi abuelita Yolita Echeverría, esto es para ustedes con mucho cariño.

Johanna Moya

# DEDICATORIA

Siempre me he sentido maravillado por la linda familia que tengo, se han preocupado de mí desde el momento en que llegué a este mundo, me han formado para saber cómo luchar y salir victorioso ante las diversas adversidades de la vida. Dedico esta tesis a mi madre y hermana quienes son el pilar fundamental de mi vida que siempre han estado apoyándome en todo momento para conseguir un logro más en vida sin que me rinda y siempre siguiendo para adelante y todos los que me apoyaron para escribir y concluir esta tesis. Para ellos es esta dedicatoria de tesis, pues es a ellos a quienes se las debo por su gran apoyo incondicional.

*Andrés Escobar*

# AGRADECIMIENTO

Quiero agradecer de manera muy especial a mi Dios por darme la vida, salud, sabiduría y energía para poder culminar esta meta más trazada en mi vida.

Agradecerle a mi mamita Marthita, por ser mi pilar fundamental en mi vida, por creer siempre en mí, por su apoyo incondicional en todo momento, gracias a todos los sacrificios que hizo para poderme educar en esta noble y prestigiosa universidad, agradecerle a mi papito Carlitos, quien desde pequeña me enseñó a ser valiente a seguir su ejemplo, aprender que todo lo que sucede en esta vida es por algo y a sacar el lado positivo de cada situación que pasa en la vida. A mi hermana Carlita quien gracias a su compañía, risas y porque es la mejor hermana he podido seguir adelante. Como no mencionar a mis abuelitos Yolita, Pablo, Teresa, Michita, Enriqueta, Hilda, quienes con sus mimos y dulzura he crecido llena de amor y he podido seguir adelante con sus bendiciones.

Agradecerles a mis queridos tíos Johnny y Eduardo, gracias a sus consejos y preocupación, quienes han estado con nosotros en las buenas y en las malas siempre apoyándonos y dándome ánimos para siempre seguir adelante. Agradecer a Marianita Paredes, que durante esta vida universitaria nos apoyó y siempre me dio los mejores consejos para seguir adelante.

Agradecer a todos mis ingenieros ya que me brindaron sus conocimientos, sus consejos y las ganas de ser mejor cada día, especialmente quiero agradecer a mi director de tesis Ing. Alfredo Calderón, quien nos aconsejó y nos brindó su tiempo en todo momento, que diosito le de mucha salud y le colme de muchas bendiciones. Agradecerle de una manera muy especial a Ing. Javier Cóndor, quien tomó la batuta de nuestra disertación como nuestro director en la enfermedad del Ing. Calderón, gracias por su apoyo incondicional, por dedicarnos tiempo y por siempre estar presto a solventar nuestras dudas. A nuestra lectora Ing. Beatriz Campos, por todo el tiempo que nos brindó, por compartir su sabiduría y ayudarnos en nuestras dudas y ayudarnos en la parte final de la tesis.

En especial agradecer a mi amigo Andrés Escobar, por el tiempo y ayuda en la elaboración de nuestra tesis, por los ánimos, consejos y las buenas vibras para seguir adelante y terminar con éxito la última tarea de esta meta, gracias al tiempo por permitirnos conocer y aspirar a ser los mejores.

Agradecer a mi equipo Panteras Rugby Club y entrenadores, quienes han sido mi apoyo, mi fortaleza, gracias por sus buenas vibras, gracias por su amistad.

Johanna Moya

# AGRADECIMIENTO

Agradezco a Dios que me dio la fuerza y fe para creer que todo sería posible y terminar mi tesis y así cumplir un logro más en mi vida. A mi Madre y Hermana por ayudarme a seguir adelante cada día de trabajo en el desarrollo de mi tesis y darme su apoyo incondicional cuando lo sentía muy lejano de culminar les agradezco cada momento de su apoyo y tiempo.

A mi compañera y amiga de tesis que pese a momentos difíciles que pasamos en todo el camino de desarrollo siempre estuvo apoyándome y dándome los mejores consejos de seguir adelante.

También expresar mis agradecimientos a todos los ingenieros por su apoyo y tiempo para culminar exitosamente mi tesis.

Andrés Escobar

# TABLA DE CONTENIDO

# INDICE DE ILUSTRACIONES

## INDICE DE TABLAS

# RESUMEN

La vulnerabilidad en la información es muy común hoy en día. Por lo que en los últimos años ha adquirido un auge el estudio e implementación de diferentes modelos de encriptación para asegurar la confidencialidad en el intercambio de información. Convirtiéndola en una "tentación" para toda la adquisición de productos en el mercado que ayuden con este problema; por ello, resulta de suma importancia encontrar e implementar medidas de seguridad que garanticen la integridad de la misma.

Un ejemplo claro que puede suscitarse de gran ayuda para evitar este tipo de vulnerabilidad en la información real de la trasmisión de datos es la aplicación de la encriptación que es el proceso mediante el cual cierta información o texto sin formato es cifrado de forma que el resultado sea ilegible a menos que se conozca los datos necesarios para su interpretación, siendo así una medida de seguridad utilizada para que al momento de almacenar o transmitir información real o sensible ésta no pueda ser obtenida con facilidad por terceros, opcionalmente puede existir además un proceso de "des-encriptación" a través del cual la información puede ser interpretada de nuevo a su estado original, beneficiando así a las organizaciones en el aspecto de costos bajos con beneficios mayores en la seguridad de la información real de trasmisión de datos.

Esta investigación aplicada aborda la protección desde un punto de vista que proporcione información sobre el funcionamiento de algunos algoritmos de cifrado.

Se aplicará toda la información que se ha podido investigar durante el desarrollo de esta disertación, se utilizara metodologías conocidas como el método de SCRUM ya que con este los entregables son pequeños y se pueden revisar periódicamente, haciendo más efectiva la identificación de errores y cambios, además de que Scrum se enfoca en la entrega de productos y no tanto en la calidad del código como Xtreme Programming, como no se va a hacer algo que parta desde cero y más bien se va a adaptarlo a nuestras preferencias además se va a reutilizar herramientas ya existentes en caso de necesitarlas. Para el desarrollo iremos probando herramientas como Dreamweaver, PHP, HTML, Visual Studio, ASP. NET y gestores de bases de datos como MySQL, Microsoft SQL Server, PostgreSQL que nos permiten realizar cambios hasta poder obtener nuestro objetivo final para nuestra disertación.

# INTRODUCCION

En el desarrollo de esta disertación de grado se tratara sobre la encriptación o el cifrado de datos. Nos apasionamos por el tema ya que en la actualidad se tiene mucha vulnerabilidad en cuanto se refiere al envió de nuestra información a otra persona mediante nuestro equipos tecnológicos, ya que tenemos interceptores que pueden utilizar esta información para un mal uso.

En la actualidad todos nosotros tenemos una leve idea sobre los países del primer mundo que nos espían que saben lo que escribimos en nuestros correos electrónicos, saben lo que hablamos pues gracias a esa leve idea, nosotros le podemos afirmar que es cierta. Nos basamos en decir que esto es cierto ya que Julian Assange escribió en su libro y nos informa sobre los criptopunks quienes son los que abogan por el uso de la criptografía, su lema es "privacidad para los pobres, transparencia para los poderosos[1]", además tenemos un claro ejemplo y este es ***Facebook*** es una corporación estadounidense dispone de una penetración casi total en la población de un país entero y fácilmente todos publicamos nuestra información. No es secreto que en lo referente a internet a las comunicaciones telefónicas, todos los caminos desde y hacia América Latina pasan por Estados Unidos. La infraestructura de internet dirige gran parte de tráfico desde y hacia América Latina a través de cables de fibra óptica que físicamente atraviesan las fronteras de Estados Unidos. Nos podemos dar cuenta que sus pilares son cables de fibra óptica que se extienden a lo largo del suelo oceánico, satélites que giran sobre nuestras cabezas, servidores informáticos alojados en edificios ciudades de New York a Nairobi. Talvez estemos hablando de la militarización del espacio cibernético ya que como saben todo lo que hacemos, lo podríamos representar como un tanque en nuestra habitación o un soldado entre tus familiares y tú mientras envías información por internet, mientras hablamos con nuestros seres queridos, con nuestros más íntimos, lo que quiere decir que hemos ingresado a un espacio militarizado.

Creemos que uno de los mejores proyectos para frenar este espacio militarizado es el proyecto TOR[2] este proyecto es un sistema de libre acceso en línea para mantener el anonimato para todas las personas que resistan la vigilancia y eludan la censura a internet.

Otro ejemplo de asegurar porque necesitamos cifrar nuestros datos en Snowden quien fue un antiguo informante de la CIA[3] en sus declaraciones dijo "no pude en conciencia,

[1] Julian Assange, Libro Criptopunks la libertad y el futuro de internet, pág. 19. Para más información consultar en el Glosario

[2] TOR: es un software que funciona mediante un buscador online que evita que alguien que observa su conexión de internet sepa que sitios visita. Para más información consultar en el Glosario

[3] CIA: Agencia central de Inteligencia

permitir al gobierno de Estados Unidos destruir la privacidad, la libertad en internet y las libertades básicas de la gente de todo el mundo con esta gigantesca máquina de vigilancia que están construyendo en secreto[4]". "No quiero vivir en una sociedad que hace este tipo de cosas…. No quiero vivir en un mundo donde se registra todo lo que hago y digo.[5]"

Por estas pocas razones expuestas pero creemos que son unas de las más importantes es por ello que contribuyendo con la privacidad para la gente común justificamos el desarrollo de este tema de disertación de grado.

[4] Snowden, 2 Septiembre del 2013, entrevista, Somos transparentes, son opacos.

[5] Snowden, The guardian News Media 2013

# 1. CAPÍTULO I –MARCO TEÓRICO

La encriptación es básicamente la manera de asegurar información importante, esta normalmente funciona con algoritmos los cuales pueden transformar una pequeña información en una muy extensa y difícil de descifrar a simple vista.

Estos algoritmos de encriptación toman una parte importante ya que se han reconocido. El usuario "no autorizado" al tratar de descifrar se encontrará con una llave la cual tiene dos clases una llamada (Key`S) o la más usada en internet llamada (Public Key) esta se da a conocer a cualquier persona q lo desee y ante cualquier internet, sin embargo existe otra llamada (única) que solo será conocida por un único usuario.

En la encriptación existen varios niveles pero las encriptaciones más comunes son las de 40-512 bits (llave secreta y llave publica) 128 -1024 bits (llave secreta y llave pública) esta última es la más fuerte que existe en el mercado.

La encriptación de 40-512 bits es usada en la gran mayoría de los sitios de internet, la encriptación 128-1024 bits son usados en transacciones de alto riesgo, como las bancarias.

Por ultimo hacemos mención de la encriptación actual de que tan segura puede ser, la llave privada que por lo general trata de 40 bits en ocasiones puede llegar a ser interceptada y la de 128 la cual es 3 veces más poderosa o segura.

En conclusión la encriptación tiene como fin evitar que la o las personas "no autorizadas" se adueñen de información que no les pertenece y se podrá decir que el usuario se encuentra un poco más seguro.

## 1.1 Encriptación o cifrado de datos

Es el proceso por el que la información legible se transforma mediante algoritmo en información ilegible, que se le denomina criptograma[6] o información secreta.

Se la puede denominar una medida de seguridad que es usada para transferir o almacenar información delicada que no debería ser accesible a terceros. Es una tecnología que permite la transmisión segura de la información, ya que al codificar los datos transmitidos se usa una fórmula matemática que "desmenuza" los datos.

Generalmente a este proceso que se indica anteriormente se le denomina "encriptación" pero es incorrecto ya que esta palabra NO existe en castellano se ha importado del inglés "encrypt", que se traduce como "cifrar", y por lo tanto al proceso se lo denomina cifrado.

[6] Criptograma: es un documento escrito en clave

Debemos tener en cuenta conceptos claros como:

**Criptografía:** es la ciencia que estudia la transformación de un determinado mensaje es un código de forma tal que a partir de dicho código solo algunas personas son capaces de recuperar el mensaje original.

Además para utilizar esta transformación, se utilizan las matemáticas para encriptar y desencriptar datos. Una vez que la información es encriptada, puede ser almacenada en un medio inseguro o enviada a través de una red insegura(como internet) y aun así permanecer secreta.

**Cristología:** es la ciencia que estudia los sistemas utilizados para ocultar la información.

### 1.1.2 Historia del cifrado

Las técnicas de cifrado se usan desde la antigua Grecia. La palabra criptografía vienen del griego: **Krypto** "ocultar" y **graphos** "escribir" lo que se denomina escritura oculta.

- Escítala: consistía en envolver en una vara con una tira de papel, y escribir el mensaje longitudinalmente al cual se la ponía una letra en cada vuelta de la tira. Esta tira era enviada al destinatario y este solo podía leerlo envolviéndolo en una vara del mismo diámetro, ya que para distintos diámetros generaría distintas combinaciones de letras.

Figura 1- 1. **(Luringen, 2007)**

- También los hebreos cifraban textos, según menciona La Biblia, esto se hacía mediante el uso del alfabeto invertido. Reemplazaban la primera letra del alfabeto por la última, la segunda por la penúltima a este sistema se lo denominaba Atbash.


Figura 1- 2. Atbash (Public, 2013)

- Cifrado de Cesar: era un algoritmo sencillo de sustitución, se debía asignar a cada letra un valor hacer modificaciones sobre el texto legible de forma que solo el destinatario supiera que modificaciones se habían hecho.

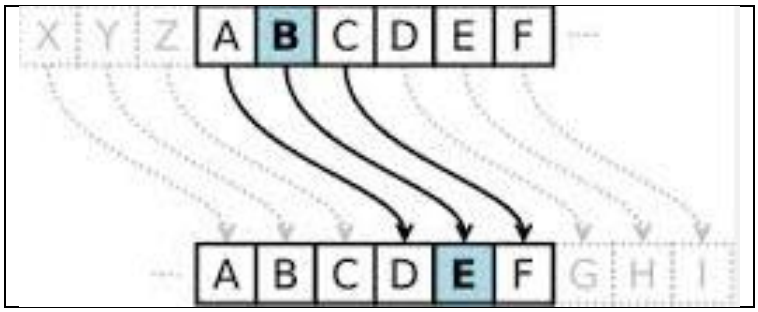



Figura 1- 3. Cifrado de Cesar (Cepheus, 2006)

- Esteganografía: consistía no solo en cifrar texto sino en ocultarlo de forma que nadie sepa que hay información. A continuación se da un ejemplo de cómo los esclavos se les tatuaba la información en la cabeza y después se les dejaba que crezca el cabello, de forma de que nadie sospechase que hubiere tenido información.

Figura 1- 4. Esteganografía (García)

- Enigma: la maquina usada en las dos guerras mundiales por el bando alemán para codificar mensajes sensibles. Usaba algoritmos de sustitución y transposición complejos.

Figura 1- 5. Enigma (Ianturton, 2014)

- Hoy en día podemos decir que el cifrado de datos ha evolucionado hacia sistemas mucho más complicados de poder romper y usar a diario para aplicaciones diversas como las conexiones seguras a ciertas páginas de internet a las que se envía información sensible, almacenamiento de datos seguro , o como ya se tiene en algunas empresas la firma electrónica.

## 1.2 Procedimientos de encriptación

En procedimientos de la encriptación se hará referencia de como paso a paso se puede ir cifrando la información, pero antes de referirnos al procedimiento de encriptación se explicara todo lo que abarca la encriptación.

- Consultoría de seguridad y gestión de riesgos: en este procedimiento se tiene la consultoría que están dedicados a la definición de planes directores de seguridad, planes de seguridad preventiva, políticas y planes de contingencia.
- Arquitectura de seguridad: para este segundo procedimiento podemos encontrar las plataformas de seguridad perimetral tales como (redes privadas, cortafuegos, redes virtuales, etc.) de contenido, de detección y prevención de intrusiones, de análisis y la gestión de vulnerabilidades.
- Certificación y firma electrónica: para la certificación y firma electrónica se ocupa del despliegue de infraestructura de certificación (PKI) corporativas, en donde tienen en cuenta desde la definición de políticas y las prácticas de certificación hasta el despliegue de servicios avanzados como el sellado de tiempo o la validación.
- Gestión de identidades: en este sistema podemos encontrar la seguridad corporativa efectiva y alineada con la realidad del negocio en la cual se garantiza: Disponibilidad de los sistemas de información, la recuperación rápida y completa de los sistemas de información, la integridad de la información y por último la confidencialidad de la información.

El principal procedimiento que tenemos para el cifrado de datos es utilizas un

sistema de clave pública que permite conjuntamente con la firma digital, el aseguramiento de la integridad de los datos que son transmitidos o almacenados. La encriptación con algoritmos de clave pública funciona con un par de llaves, que es una pública y una privada.

Estas claves permiten que el receptor y emisor mantenga una comunicación confiable permitiendo que los datos viajen a través de la red encriptados y que al llegar al receptor, pueda el mismo recomponer la información fácilmente.

Además tenemos el procedimiento matemático que son simples algoritmos pero que tienen una inversa que se cree (pero no se prueba) que es muy complicada.

En cuanto al **algoritmo criptográfico** un algoritmo o cifrado es una función matemática usada en los procesos de encriptación y desencriptación. Un algoritmo criptográfico trabaja en combinaciones con una llave (un número, palabra, frase o contraseña). Para encriptar, el algoritmo combina matemáticamente la información a proteger con una llave provista. El resultado de este cálculo son los datos encriptados. Para desencriptar, el algoritmo hace un cálculo combinado los datos encriptados con una llave provista, siendo el resultado de esta combinación los datos desencriptados.

Si la llave o los datos son modificados el algoritmo se produce un resultado diferente. El objetivo de un algoritmo criptográfico es hacer tan difícil como sea posible desencriptar los datos sin utilizar la llave.

Si se usa un algoritmo de encriptación realmente bueno entonces no hay ninguna técnica mejor que intentar metódicamente cada llave posible. Incluso para una llave de solo 40 bits, debería realizar 2^40 (más de 1 trillón) de llaves posibles.

## 1.3 Algoritmos de encriptación

El algoritmo de encriptación es un procedimiento que transforma un mensaje, sin atender su estructura lingüística o significado, de tal forma que sea incomprensible, o por lo menos difícil de comprender para terceras personas.

### 1.3.1 Algoritmo HASH[7] o de resumen

Es una función para identificar o resumir probabilísticamente un gran conjunto de información, el que da como resultado un conjunto de margen finito generalmente menor. Se refiere a una función o método para generar llaves que representan de una manera precisa a un documento, registro, archivo etc. Como se puede observar en la Figura 1-06. Algoritmo HASH, entra texto plano y se aplica la función HASH, para que me devuelva texto cifrado en el proceso

[7] HASH: Funciones picadillo, funciones de resumen o funciones de digest

lo que realiza es generar claves o llaves que representen casi de manera univoca a un documento, registro archivo, etc. Realizado el hashing[8], no existe ninguna forma de saber exactamente que texto se resumió, incluso si un texto determinado produce siempre la misma huella.

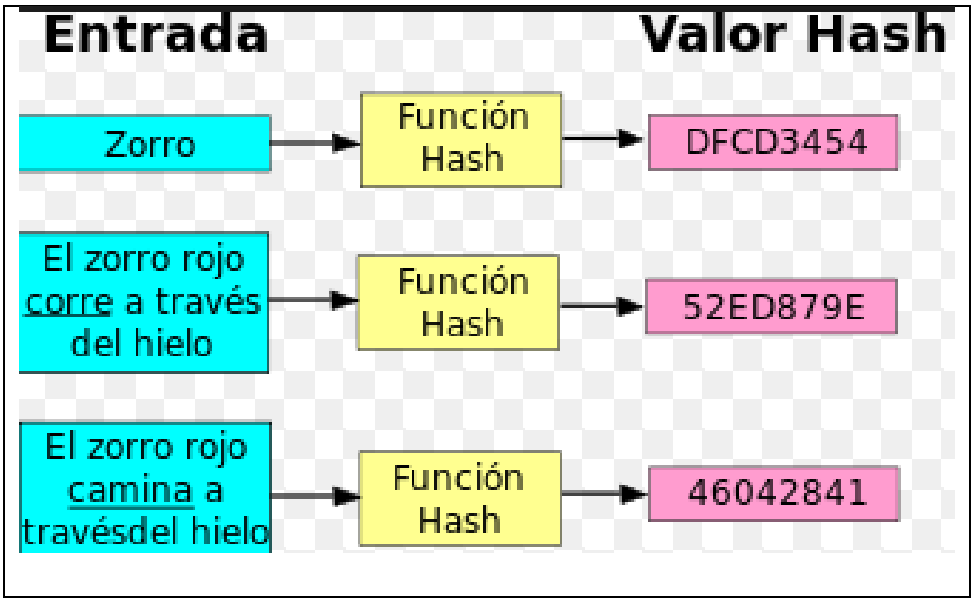


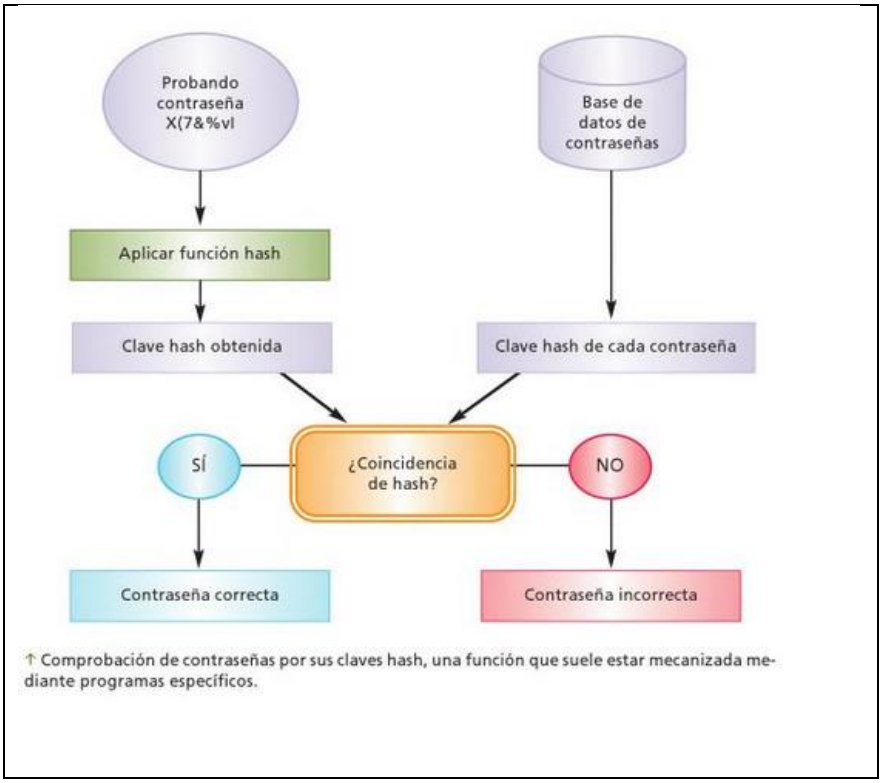

Figura 1- 6. Algoritmo HASH1 (Fercufer, 2011)

Se puede decir que es un método que sirve para generar claves o llaves. Haciendo hincapié en que es una operación matemática que se realiza sobre un conjunto de datos de cualquier longitud, y su salida será en una huella digital, que estará representado en tamaño fijo e independiente de la dimensión del documento original. Cabe señalar que el contenido es ilegible

En conclusión este algoritmo efectúa un cálculo matemático sobre los datos que constituyen el documento y da como resultado un número único llamado MAC.

Como podemos observar en la Figura 1-06. Algoritmo HASH1, se prueba contraseñas, se genera la función hash, y nos entrega una clave hash, luego para verificar se ingresa la clave hash, el gestor de base de datos comprueba que esa clave sea la correcta y nos devuelve un mensaje, diciendo que la contraseña es correcta, o por el contrario contraseña incorrecta y se sigue probando otras contraseñas.

[8] Hashing: Resume el texto en una pequeña huella que no puede descifrase (facebook, 2013)




↑ Comprobación de contraseñas por sus claves hash, una función que suele estar mecanizada mediante programas específicos.

Figura 1- 7. Algoritmo HASH (Solano, 2012)

Entre los algoritmos de HASH tenemos:

- **SHA[9]-1**

  Es un algoritmo de hash seguro, de síntesis que genera un hash de 160 bits de un mensaje de tamaño máximo $2^{64}$ bits. Es una función de un solo sentido que implica que sea sentido único ya que al mensaje se puede calcular su valor hash, pero no se puede recrear a su mensaje original.

- **MD2[10]**

  Es una función criptográfica de hash que fue desarrollado por Ronald Rivest en el año de 1989. Este algoritmo esta optimizado para computadoras de 8 bits, el valor hash de cualquier mensaje se forma haciendo que el mensaje sea múltiplo de la longitud de bloque en el ordenador (128 bits o 16 bytes) y añadiéndole un checksum[11]. Para el cálculo real, se utiliza un bloque auxiliar de 48 bits y una tabla de 256 bytes que contiene dígitos al azar del número pi.

[9] SHA: Secure Hash Algorithm
[10] MD2: Message-Digest Algorithm 2
[11] Checksum: Se basa en la suma de chequeo de internet: se suman todas las palabras de 16 bits que conforman el mensaje y se transmite, junto con el mensaje, el resultado de la suma es el algoritmo

- **MD4**[12]

  El MD4 es un algoritmo de resumen que fue desarrollado por Ron Rivest, y dado a conocer en el año de 1990 a través de un RFC[13] , este algoritmo persigue objetivos como la seguridad, rapidez, sencillo y compacto, El MD4 debe ser capaz de generar un digest[14] único para cada mensaje diferente, aun cuando la diferencia sea mínima, lo que en resumen quiere decir que de ninguna manera dos mensaje generen el mismo bloque de digest. Si nos referimos a la rapidez tiene la finalidad de que su tiempo de procesamiento sea mínimo y entregue a la menor brevedad el bloque de digest. El algoritmo fue implementado para trabajar palabras de 32 bits, lo que significó una mejora con respecto a su antecesor el MD8 que solo opera con 8 bits

- **MD5**[15]

  Fue desarrollado por Ron Rivest en 1992, dicho algoritmo tuvo la finalidad de robustecer el MD4 y a la fecha se trata de un algoritmo seguro. Es de un esquema de hash de 128 bits que es muy utilizado para la autentificación cifrada. Gracias al MD5 se consigue, que un usuario demuestre que conoce una contraseña sin necesidad de enviar la contraseña a través de la red. Además procesa mensajes de cualquier longitud y procesa bloques uniformes de 512 bits a la vez, hasta concluir con el mensaje total a fin de entregar a la salida un digest de 128 bits, que es una longitud fija.

- **SHA-256**[16]

  El primer SHA, fue publicado en 1993, actualmente tenemos cuatro variantes , que se diferencian en el diseño y rangos de salida incrementados, para el SHA-256 es una función criptográfica que es como una firma para un texto o archivo de datos, el algoritmo SHA-256 genera un tamaño fijo de 256 bits (32 bytes), esto hace que sea adecuado para la validación de contraseña, autentificación de hash es un desafío ya que trata de ser anti-sabotaje.

[12] MD4: Message Digest Algorithm 4
[13] RFC: Request For Comments
[14] Digest: función de resumen de los algoritmos de HASH
[15] MD4: Message Digest Algorithm 5
[16] SHA-256: Secure Hash Algorithm

- **RIPEMD[17]-160**

  Es un algoritmo de primitivas de integridad de resumen del mensaje, es un algoritmo que reemplaza a MD4 y MD5 fue diseñado por Hans Dobbertin, Antoon Bosselaers y Bart Preneel. Genera un hash de 20 bytes (160 bits). La función de este algoritmo es disminuir la posibilidad de colisiones hash accidentales.

### 1.3.2 Criptografía de clave secreta o simétrica

El algoritmo también conocido como "Shared Secret[18]" utiliza una clave con la cual se puede encriptar y desencriptar el documento, todo documento encriptado con una clave, deberá desencriptarse en el proceso inverso, con la misma clave. Es muy importante destacar que la clave debería viajar con los datos, lo que hace arriesgado la operación, imposible de utilizar en ambientes donde interactúan varios interlocutores.

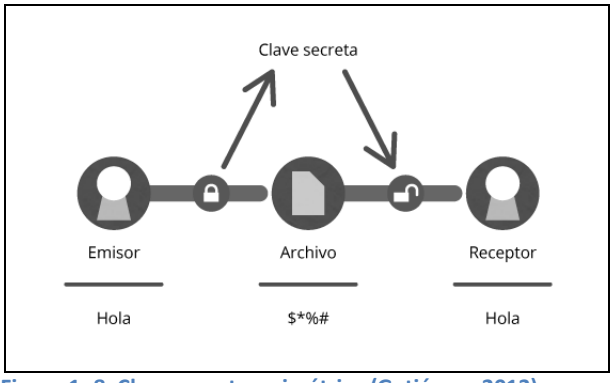



Figura 1- 8. Clave secreta o simétrica (Gutiérrez, 2013)

Los criptosistemas[19] de clave secreta se caracterizan porque la clave de cifrado y el descifrado es la misma, por lo tanto la robustez del algoritmo es permitir mantener el secreto que contiene el documento.

**Las principales características de la clave secreta o simétrica son:**

a) Cada par de usuarios tienen que tener una clave secreta compartida
b) Rápidos y fáciles de implementar
c) Una comunicación en la que intervengan múltiples usuarios la que requiere muchas claves secretas distintas.
d) Clave de cifrado y descifrado son las mismas

[17] RIPEMD: RACE Integrity Primitives Evaluation Message Digest
[18] Shared Secret: Secreto compartido
[19] Criptosistema: es el conjunto de procedimientos que garantizan la seguridad de la información y utilizan técnicas criptográficas

Como se observa en la Figura 1-09. Principales características de la clave secreta, tanto el emisor como el receptor deben conocer la clave que manejaran para el envió y recibimiento de la información.

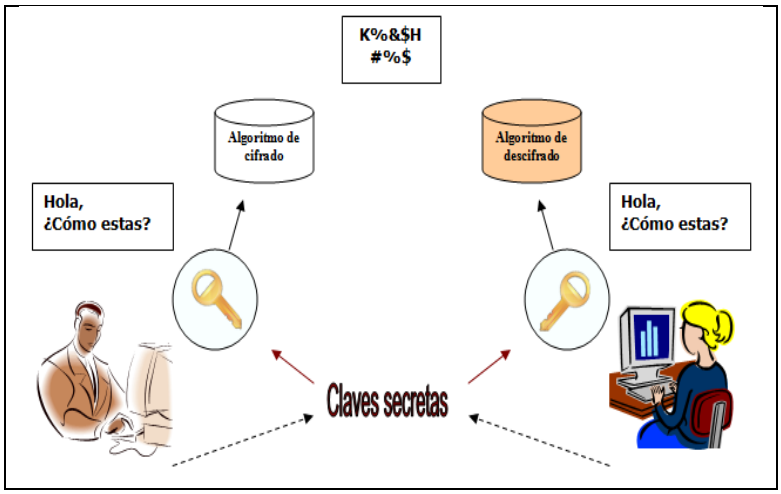



Figura 1- 9. Principales características de la clave secreta (UNAD)

### 1.3.3 Cifrado de flujo

Aquí tendremos como emisor y receptor y estarán representados **A** para emisor y **B** para receptor, A con una clave secreta y un algoritmo determinístico (RKG[20]), genera un secuencia binaria o secuencias binarias cuyos elementos se suman modulo dos con los correspondientes bits de texto claro **m**, dando lugar a los bits de texto cifrado que será representado por **C**, esta secuencia **C** es la que se envía a través del canal. **B** , con la misma clave y el mismo algoritmo determinístico genera la misma secuencia de cifrante(s), que se suma al módulo dos con la secuencia de cifrado **C**, dando lugar a los bits de texto claro .

Algo que se debe tomar muy en cuenta es que los tamaños de la clave oscilan entre 120 y 250 bits. La transformación se aplica sobre cada carácter del mensaje original.

Como podemos observar en la Figura 1-10. Cifrado de flujo, se convierte el texto en claro en texto cifrado bit a bit, esto se logra construyendo un generador de flujo de clave que es una secuencia de bits de tamaño arbitrario

[20] RKG: Random Key Generator o generador de claves pseudoaleatorias

que puede emplearse para oscurecer los contenidos de un flujo de datos combinando el flujo de clave con el flujo de datos mediante la función XOR[21]

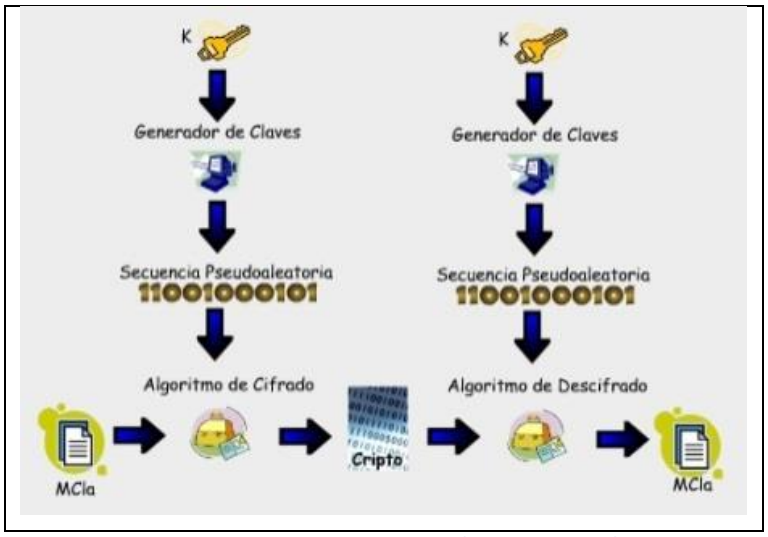



Figura 1- 10. Cifrado de flujo (Adiccion, 2012)

### 1.3.2 Cifrado en bloque

Operan sobre bloques de tamaño mayor que un bit del texto en claro y crean un bloque de tamaño fijo de texto cifrado, en la actualidad se suele trabajar con grupos de bits debido a que los mensajes a cifrar se codifican a esta forma previamente utilizando el código ANSII, para lo cual quiere decir que cada algoritmo encriptado/desencriptado procesa un bloque de tamaño **n** de salida por cada bloque de entrada. La transformación se aplica sobre un grupo de caracteres del mensaje original.

Como se puede observar en la Figura 1-11.Cifrado en bloque, es una unidad de cifrado de clave simétrica que opera en grupos de bits de longitud fija, llamados bloques. Y el proceso de este cifrado es que el texto ingresa como un bloque de texto plano y produce un bloque cifrado de igual tamaño. La transformación exacta es controlada utilizando una segunda entrada, que es la clave secreta provista por el usuario.

[21] XOR: Disyunción exclusiva. Para más información consultar en el Glosario

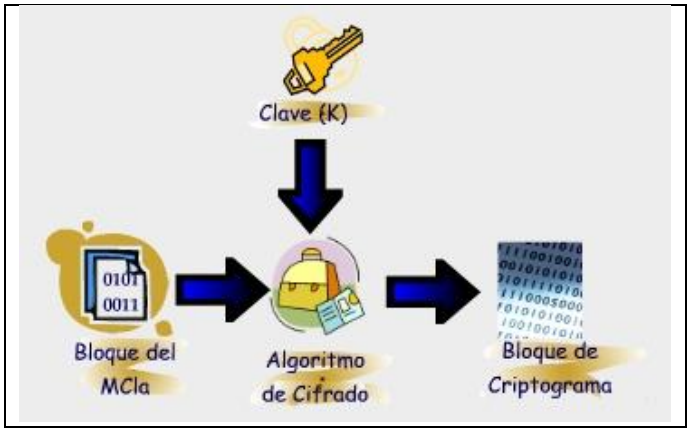



Figura 1- 11. Cifrado en bloque (México U. n., 2012)

### 1.3.4 Algoritmo Asimétrico (RSA)

Es un algoritmo conocido como "Criptografía de llave Pública[22]" que altera los datos de un documento con el objeto de alcanzar algunas características de seguridad como autenticación, integridad y confidencialidad. Este algoritmo no se basa en una única clave sino en un par de ellas: una conocida pública y otra privada.

Las dos claves pertenecen a la misma persona que ha enviado el mensaje. Una clave es pública y se puede entregar a cualquier persona, la clave privada el propietario debe guardarla de modo que nadie tenga acceso a ella.

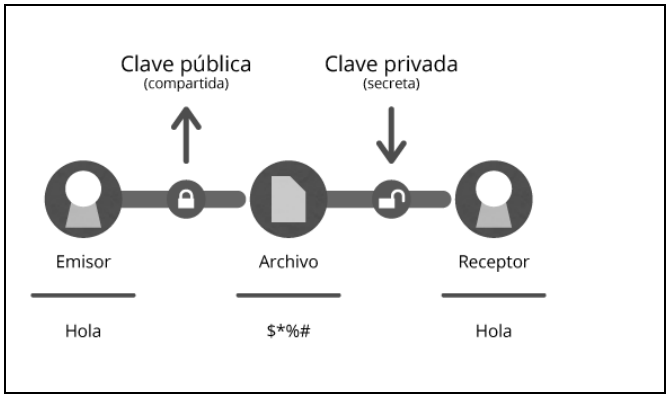



Figura 1- 12. Algoritmo Asimétrico (Hardcode xploit, 2012)

Si el remitente usa la clave pública del destinatario para cifrar el mensaje, una vez cifrado, solo la clave privada del destinatario podrá descifrar este mensaje, ya que es el único que la conoce. Por este motivo se logra la confidencialidad del envío de mensajes, nadie salvo el destinatario puede descifrarlo.

[22] (López, 1998)

**Seguridad.-** En este algoritmo se puede considerar como seguro ya que la clave pública se distribuye gratuitamente a cualquier persona que quisiera enviar un mensaje , la clave privada nunca se distribuye .

**Velocidad**.- Este algoritmo asimétrico es más complejo que sus homólogos simétricos. Las cuales requieren mucha más potencia de procesamiento informático, tanto para cifrar y descifrar mensajes.

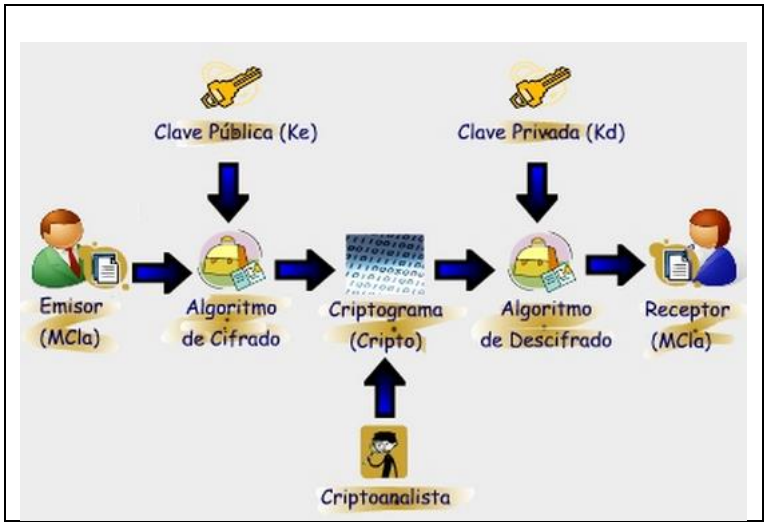



Figura 1- 13. Procedimiento Asimétrico (México U. N., 2012)

### 1.3.5 Diferencias entre algoritmo simétrico y los asimétricos

La criptografía simétrica se puede decir que es más insegura ya que al momento de pasar la clave es una gran vulnerabilidad, pero a la vez se puede cifrar y descifrar en menor tiempo del que tarda la criptografía asimétrica. Por lo cual se dice que el algoritmo de encriptación simétrica mezcla la trasposición y la permutación, mientras que los de clave pública se basan más en complejas operaciones matemáticas.

Los algoritmos asimétricos, que también son conocidos como algoritmos de llave pública, necesitan al menos una llave de 3.000 bits para alcanzar un nivel de seguridad similar al algoritmo simétrico de 128 bits. Estos algoritmos son demasiado lentos, tanto que no pueden ser utilizados para encriptar grandes cantidades de información. Los algoritmos simétricos son aproximadamente 1.000 veces más rápidos que los asimétricos.

Como se puede observar en la Figura 1-14. Diferencias Simétricos vs Asimétricos, tenemos que para el cifrado asimétrico se encripta con la clave

del emisor y para desencriptar, se utiliza la clave del receptor. Lo que no sucede con el cifrado simétrico, se utiliza la misma clave para encriptar como para desencriptar el mensaje.

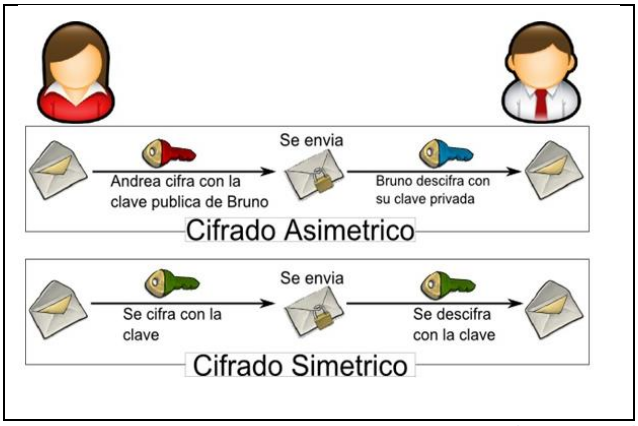



Figura 1- 14. Diferencias simétricas vs Asimétricos (Jessy, 2012)

### 1.3.6 Firma Digital

Es un conjunto de datos asociados o también llamado mecanismo criptográfico, a un mensaje digital que permite garantizar la autoría e integridad de los documentos y del mensaje.

La firma digital es un instrumento que posee normativas y técnicas, esto quiere decir que se debe seguir procedimientos técnicos que permiten la creación y verificación de firmas digitales, y existen documentos normativos que respaldan el valor legal que poseen dichas firmas.

Que realiza la firma digital:

- Validad identidad
- Evitar falsificaciones
- Seguridad de datos confidenciales
- Gestiones ante administraciones publicas
- Factura digital

El funcionamiento de la firma digital es el siguiente:

Se basa en dos claves numéricas una privada y una publica con una relación matemática entre ellas que generan a partir de un certificado digital solicitado por el usuario a una entidad de certificación acreditada electrónico, la clave privada debe ser conocida únicamente por su titular y se almacena normalmente en el disco duro o en una tarjeta criptográfica, la clave pública en cambio se distribuye junto con el mensaje firmado. Si la clave de cifrado solo

la tiene una persona lo que cifra esa persona podrá ser utilizado como su firma electrónica ya que solo ella puede realizar.

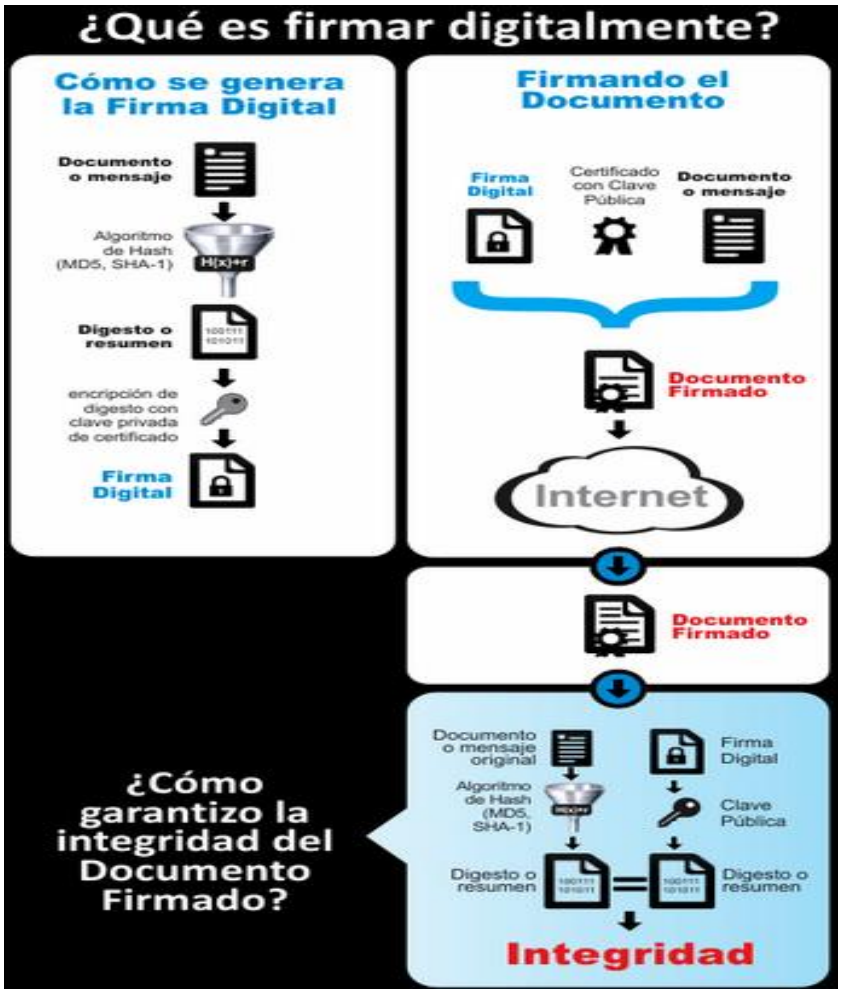



Figura 1- 15. Firma Digital (Hermes, 2011)

Como se puede observar en la Figura 1-15. Firma Digital, se encuentra de una manera resumida lo que es una firma digital y que se realiza primero en un algoritmo matemático aplicando la función hash, el cual se cifra con la clave privada del firmante, lo que permite que terceras personas puedan reconocer la identidad del firmante y asegurarse de que los contenidos no han sido modificados.

### 1.3.7 Protocolos criptográficos o de seguridad

Un protocolo criptográfico es un protocolo abstracto o concreto que realiza funciones relacionadas con la seguridad, aplicando los métodos criptográficos.

No es suficiente estudiar la seguridad de los algoritmos de base solamente, como tampoco las debilidades en un protocolo o aplicación de más alto nivel se pueden traducir en cuan insegura es una aplicación o que tan bueno es el algoritmo criptográfico de base. El análisis de los protocolos es generalmente difícil porque las aplicaciones que implementan dichos protocolos pueden conducir a problemas adicionales. De esa manera un buen protocolo no es suficiente, se debe tener una buena y robusta implantación.

- **Secure Socket Layer (SSL[23]):** Es una capa con protocolos criptográficos que proporcionan conexiones seguras por una red, utiliza la criptografía asimétrica para autentificar a la contraparte con quien se está comunicando. Está sesión es luego usada para encriptar el flujo de datos entre las partes, lo que permite la confidencialidad del dato/mensaje. Es uno de los dos protocolos para conexiones seguras WWW[24], la seguridad de www se ha vuelto importante con el incremento de información sensible, como números de tarjeta de crédito.
- **Transport Layer Security (TLS[25]),** es una capa de transporte seguro mediante el cual se establece conexión segura a un canal cifrado entre el cliente y servidor. Así el intercambio de información se realiza en un entorno seguro y libre de ataques.

  **Objetivos:**
  - Seguridad Criptográfica se debe establecer una conexión segura entre dos partes
  - Interoperabilidad aplicaciones diferentes deberían intercambiar parámetros criptográficos sin necesidad que ninguna de las dos conozca el código de la otra
  - Extensibilidad permite la integración de nuevos algoritmos criptográficos
  - Eficiencia el protocolo contiene un esquema de cache de sesiones para reducir el número de sesiones que deben inicializarse desde cero
- **Domain Name Server Security (DNSSEC[26]). -** Protocolo para servicios de distribución de nombres seguros de dominio DNS[27] y lo protegen de las amenazas en línea. A lo que refiere que es un conjunto de extensiones al DNS que proporcionan a los clientes la autentificación del origen de datos DNS, la negación autenticada de la existencia e integridad de datos, pero no disponibilidad o confidencialidad.

[23] SSL: Secure Sockets Layer
[24] WWW: expression inglesa World Wide Web, red informática mundial
[25] TLS: Transport Layer Security
[26] DNSSEC: Domain Name Server Security
[27] DNS: Sistema de nombre de dominios. Para más información consultar en el Glosario

- **Generic Security Services API (GSSAPI[28]).-** Provee una interface de autenticación, intercambio de claves y encriptación para diferentes algoritmos de cifrado y sistemas
- **Secure Hypertext Transfer Protocol (SHTTP[29]). -** Es el protocolo usado para transacciones seguras en la Web los hackers no son capaces de interceptar el mensaje que contiene los datos confidenciales mientras se dirige al servidor.
- **Estándares de encriptación de llave pública (PKCS[30]).-** Este algoritmo proporciona un estándar para derivar una clave secreta de una contraseña
- **Especificaciones sobre el estándar criptográfico de llave publica(IEEE P1363[31]).-** Son varios algoritmos de llave publica para encriptación y firma digital
- **Publius Censor-Resistent Publishing Protocol[32]-** Protocolo resistente a censura, permite a un grupo de autores y lectores compartir documentos en una serie de servidores web en el cual no se les obliga a revelar su identidad, ya que los documentos se encuentran certificados. Los documentos no pueden ser removidos o modificados
- **Secure Shell.-** Es utilizado para asegurar sesiones de terminal y conexiones arbitrarias de .

## 1.4 Técnicas matemáticas de encriptación de datos

Desde la óptica matemática, siempre un mensaje se vería transmitir a maneras de números. Ya que internamente las computadoras representan todos los caracteres como números binarios de acuerdo al código ASCII[33].

**CÓDIGO ASCII**

| Decimal | Signif. | Código Binario | Decimal2 | Signif.3 | Código Binario4 |
|---|---|---|---|---|---|
| **32** | Espacio | 10 0000 | 95 | _ | 101 1111 |
| **33** | ! | 10 0001 | 96 | ` | 110 0000 |
| **34** | " | 10 0010 | 97 | a | 110 0001 |
| **35** | # | 10 0011 | 98 | b | 110 0010 |

[28] GSSAPI: Generic Security Services API
[29] SHTTP: Secure Hypertext Transfer Protocol
[30] PKCS: Public-Key Cryptography Standards
[31] IEEE P1363: Institute of Electrical and Electronics Engineers Traditional public-key cryptography
[32] Publius Censor-Resistent Publishing Protocol: Protocolo resistente a censura. Para más información consultar en el Glosario
[33] ASCII: American Standard Code for Information Interchange

| Decimal | Signif. | Código Binario | Decimal2 | Signif.3 | Código Binario4 |
|---|---|---|---|---|---|
| **36** | $ | 10 0100 | 99 | c | 110 0011 |
| **37** | % | 10 0101 | 100 | d | 110 0100 |
| **38** | & | 10 0110 | 101 | e | 110 0101 |
| **39** | ' | 10 0111 | 102 | f | 110 0110 |
| **40** | ( | 10 1000 | 103 | g | 110 0111 |
| **41** | ) | 10 1001 | 104 | h | 110 1000 |
| **42** | * | 10 1010 | 105 | i | 110 1001 |
| **43** | + | 10 1011 | 106 | j | 110 1010 |
| **44** | , | 10 1100 | 107 | k | 110 1011 |
| **45** | - | 10 1101 | 108 | l | 110 1100 |
| **46** | . | 10 1110 | 109 | m | 110 1101 |
| **47** | / | 10 1111 | 110 | n | 110 1110 |
| **48** | 0 | 11 0000 | 111 | o | 110 1111 |
| **49** | 1 | 11 0001 | 112 | p | 111 0000 |
| **50** | 2 | 11 0010 | 113 | q | 111 0001 |
| **51** | 3 | 11 0011 | 114 | r | 111 0010 |
| **52** | 4 | 11 0100 | 115 | s | 111 0011 |
| **53** | 5 | 11 0101 | 116 | t | 111 0100 |
| **54** | 6 | 11 0110 | 117 | u | 111 0101 |
| **55** | 7 | 11 0111 | 118 | v | 111 0110 |
| **56** | 8 | 11 1000 | 119 | w | 111 0111 |
| **57** | 9 | 11 1001 | 120 | x | 111 1000 |
| **58** | : | 11 1010 | 121 | y | 111 1001 |
| **59** | ; | 11 1011 | 122 | z | 111 1010 |
| **60** | < | 11 1100 | 123 | { | 111 1011 |
| **61** | = | 11 1101 | 124 | \| | 111 1100 |
| **62** | > | 11 1110 | 125 | ¦ | 111 1101 |
| **63** | ? | 11 1111 | 126 | ~ | 111 1101 |
| **64** | @ | 100 0000 | 127 | ¦ | 111 1110 |
| **65** | A | 100 0001 | 128 | Ç | 1000 0000 |
| **66** | B | 100 0010 | 130 | é | 1000 0010 |
| **67** | C | 100 0011 | 144 | É | 1001 0000 |
| **68** | D | 100 0100 | 157 | Ø | 1001 1101 |

| Decimal | Signif. | Código Binario | Decimal2 | Signif.3 | Código Binario4 |
|---|---|---|---|---|---|
| **69** | E | 100 0101 | 160 | á | 1010 0000 |
| **70** | F | 100 0110 | 161 | í | 1010 0001 |
| **71** | G | 100 0111 | 162 | ó | 1010 0010 |
| **72** | H | 100 1000 | 163 | ú | 1010 0011 |
| **73** | I | 100 1001 | 164 | ñ | 1010 0100 |
| **74** | J | 100 1010 | 165 | Ñ | 1010 0101 |
| **75** | K | 100 1011 | 166 | ª | 1010 0110 |
| **76** | L | 100 1100 | 167 | º | 1010 0111 |
| **77** | M | 100 1101 | 168 | ¿ | 1010 1000 |
| **78** | N | 100 1110 | 169 | ® | 1010 1001 |
| **79** | O | 100 1111 | 171 | ½ | 1010 1010 |
| **80** | P | 101 0000 | 172 | ¼ | 1010 1100 |
| **81** | Q | 101 0001 | 173 | ¡ | 1010 1101 |
| **82** | R | 101 0010 | 181 | Á | 1011 0101 |
| **83** | S | 101 0011 | 184 | © | 1011 1000 |
| **84** | T | 101 0100 | 214 | Í | 1101 0110 |
| **85** | U | 101 0101 | 224 | Ó | 1110 0000 |
| **86** | V | 101 0110 | 225 | ß | 1110 0001 |
| **87** | W | 101 0111 | 230 | µ | 1110 0110 |
| **88** | X | 101 1000 | 233 | Ú | 1110 1001 |
| **89** | Y | 101 1001 | 241 | ± | 1111 0001 |
| **90** | Z | 101 1010 | 243 | ¾ | 1111 0011 |
| **91** | [ | 101 1011 | 246 | ÷ | 1111 0110 |
| **92** | \ | 101 1100 | 248 | ° | 1111 1000 |
| **93** | ] | 101 1101 | 252 | ³ | 1111 1100 |
| **94** | ^ | 101 1110 | 253 | ² | 1111 1 |

Tabla 1- 1. Código ASCII (Johanna Moya P. , 2014)

### 1.4.1 Funciones de una sola vía

La teoría de los algoritmos de clave pública se dice que una computadora puede realizar una operación matemática en el mejor tiempo posible, lo que no sucede cuando se tiene que realizar una operación inversa ya que requiere más tiempo para el procesamiento respectivo y lo que se volvería prácticamente imposible.

Ejemplo si tenemos dos números primos grandes x,y, es muy fácil calcular Z, Z=x,y.

Pero en el caso que se conozca Z, resulta imposible calcular el valor tanto para x como para y, ya que al inicio del ejemplo se expuso que son primos grandes.

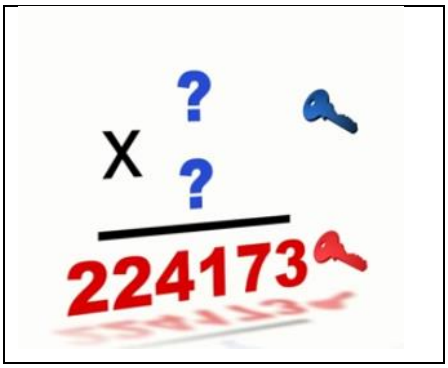



Figura 1- 16. Técnica matemática una sola vía (Departamento de electrónica, 2012)

### 1.4.2 Exponencial Modular

El par de algoritmos utiliza como función de una sola vía, la función exponencial modular que consiste en obtener la representación del exponente *n* en dígitos binarios y hallar los distintos cuadrados sucesivos. En donde:

$x_{\rightarrow a^x (mod\ n)}$

X → devuelve el resto de $a^x$ módulo n

Para calcular $a^x$ se utiliza el método de cuadrados repetidos

***Ejemplo***

$1001^{19}$ (mod301)

El exponente colocamos en números binarios

**Binario 19**

| | | | | |
|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 1 |
| 2^4 | 2^3 | 2^2 | 2^1 | 2^0 |

Tabla 1- 2. Ejemplo Binario 19 (Johanna Moya P. , 2014)

$1001^2 \equiv 273 (mod\ 301)$

$1001^4 \equiv 273^2 \equiv 182 \pmod{301}$

$1001^8 \equiv 182^2 \equiv 14 \pmod{301}$

$1001^{16} \equiv 14^2 \equiv 196 \pmod{301}$

$1001^{18} \equiv 1001^{16} * 1001^2 \equiv 196*273 \equiv 231 \pmod{301}$

$1001^{19} \equiv 1001^{18 * 1001} \equiv 63 \pmod{301}$

### 1.4.3 Raíces Primitivas

Se dice que *x* es un numero primo, en donde *j* es una raíz primitiva modulo *p*, si $j^n$ módulo de *x* recorre los valores 1,2,3,4……,x-1 cuando n recorre esos mismos valores.

Ejemplo:

*p*=23; *j*=5

Tabla de valores

| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| j^n | 5 | 2 | 10 | 4 | 20 | 8 | 17 | 16 | 11 | 9 | 22 | 18 | 21 | 13 | 19 | 3 | 15 | 6 | 7 | 12 | 14 | 1 |

Tabla 1- 3. Técnicas Matemáticas raíces Primitivas (Johanna Moya P. , 2014)

Se tiene que todo primo de *p*, existen raíces primitivas.

### 1.4.4 Algoritmo de Diffie-Hellman

Para este algoritmo se utiliza el emisor y el receptor, en donde las partes del mensaje dan a conocer una clave pública que será un número primo grande **p** y una base **j**, la cual será una raíz primitiva de **p**.

***Ejemplo***

*p*=23; *j*=5.

El emisor selecciona una clave secreta, *e* =6, con esta clave calculamos $j^e$(modulo p)

$5^6 \equiv 8 \pmod{23}$

El receptor también selecciona una clave secreta, *f*=15, realizamos la misma operación

$5^{15} \equiv 19 \pmod{23}$

Por lo tanto el emisor y el receptor pueden calcular $j^{ef}$ modulo $p$: se puede decir que se tiene un secreto en común.

Emisor $(j^f)^e \equiv j^{ef} \pmod{p}$

$19^6 \equiv 2 \pmod{23}$

Mientras que el receptor (que conoce $j^a$ y f) calcula:

$(j^e)^f \equiv j^{ef} \pmod{p}$

$8^{15} \equiv 2 \pmod{23}$

Pues para decir que el algoritmo de Diffie-Hellman depende de que se conozca $y=j^a$ o($j^b$) no sea posible determinar $a$ o $b$, no sea posible resolver la ecuación de congruencia.

$J^x \equiv y \pmod{p}$

Se puede ver que si j es una raíz primitiva de p este problema siempre tiene una solución, pero no se conoce ningún algoritmo eficiente (que sea de complejidad polinomial en el número de bits de los datos)

**Ejemplo** simulado del algoritmo de Diffie-Hellman para generar una clave simétrica para el cifrado de mensajes obtenido y conocido únicamente por el emisor y el receptor Ana y Borja seleccionan y se intercambian dos números, a y b que se utilizarán para la generación de la clave simétrica secreta compartida.

Ana selecciona después su número secreto xa. (Su clave privada). Borja realiza el mismo proceso eligiendo su clave privada xb. A partir de las dos números anteriores (a y b) y de su clave privada xa, Ana calcula ya (su clave pública) y se lo envía a Borja.

Borja calcula igualmente yb (su clave pública) y se lo envía a Ana. A partir de estos valores Ana y Borja pueden calcular la clave simétrica secreta que utilizaran para cifrar/descifrar la información que se intercambien.

**Veámoslo con un ejemplo:**

Ana elige 7 como su número inicial (a=7) y 3 como clave privada (xa=3)

Borja elige 5 como su número inicial (b=5) y 2 como clave privada (xb=2)

Ana calcula ya (su clave pública) mediante la fórmula *ya*=(*b* ** *xa*) % a y se lo envía a Borja. (Donde ** representa "elevado a", y % "resto de la división")

*ya*=(5 ** 3) % 7 = 6 (resto de 5 elevado a 3 dividido entre 7) Borja realiza el mismo proceso *yb*=(*b***xb*) % a, enviándoselo a Ana *yb*=(5 ** 2) % 7 = 4

Usando el número recibido de Borja (yb=4), Ana calcula *ka*

*ka* = (*yb* ^ *xa*) % a ; *ka* = (4 ** 3) % 7 = 1

De igual modo, Borja calcula *kb* a partir del número enviado por Ana (*ya*=6):

*kb* = (*ya* ^ *xb*) % *a*

*kb* = (6 ** 2) % 7 = 1

Ana y Borja han completado el proceso de encriptación Diffie-Hellman. Ana ha aplicado su número secreto *xa* al valor *yb* de Borja y ha calculado *ka*. Borja ha aplicado su número secreto *xb* al valor ya de Ana y ha calculado *kb*. Como se comprueba *ka = kb*, un número conocido ahora por Ana y Borja únicamente. Alicia y Borja pueden ahora usar el valor 1 para encriptar mensajes.

Si alguien "escuchara" la transmisión no le sería fácil descubrir el valor de *ka*

Con los valores *a, b, ya e y b*, "escuchados" por la red, puede escribir la ecuación

*ya* = ( *b* ** *xa*) % a y probar todos los posibles valores para encontrar *xa* (la clave privada de Ana), hasta encontrar uno que generara el valor conocido de *ya*.

Siguiendo con nuestro ejemplo donde ( *a=7 b=5 ya=6 yb=4*)

6= (5 ** 1) % 7 --> FALSO

6= (5 ** 2) % 7 --> FALSO

6= (5 ** 3) % 7 --> VERDADERO

*xa* =3 Usando este valor de xa en la ecuación

*ka* = (*yb* ** *xa*) % a

*ka* = ( 4 ** 3) % 7 = 1

Eva podría calcular el valor de ka=1. Este método de ataque de "fuerza bruta", solo es posible si Alicia y Borja han usado números pequeños (como en nuestro ejemplo).

Usando preferentemente números primos grandes, Ana y Borja pueden hacer el ataque de "fuerza bruta" de Eva impracticable

### 1.4.5 Algoritmo de Euclides

El algoritmo de Euclides permite obtener el máximo común divisor de dos números. La tarea principal de este algoritmo es verificar que dos números son primos relativos.

Ejemplo: $j$ = 231; $h$=640

| A | B | C | D |
|---|---|---|---|
| | | | |
| | 178 | | |
| | 53 | | |
| | 19 | | |
| | 15 | | |
| | 4 | | |
| | 3 | | |
| | | | |
| | | | |

Tabla 1- 4. Algoritmo de Euclides (Johanna Moya P. , 2014)

=RESIDUO (640;231)

=RESIDUO (231;B2)

=RESIDUO (B2;B3)

=RESIDUO (53;19)

=RESIDUO (B4;B5)

=RESIDUO (B5;B6)

640 = 231*2+178

231 = 178*1+53

178 = 53*3+19

53 = 19*2+15

19 = 15*1+4

15 = 4*3+3

4 = 3*1+3

3 = 1*3+0

Como consecuencia al calcular el máximo común divisor entre j y h se puede escribir en la siguiente forma *aj+dh*

En donde:

*a* y *d* son números enteros: como lo tenemos en nuestro ejemplo

178 = 1*640+ (-2)*231

53 = (-1)*640 + 3*231

19 = 4*640+3*231

15 = (-9)*640+ (-11)*231

4 = 13*640+ (-36)*231

3 = (-48)*640+133*231

1=61*640+ (-169)*231

Ahora podemos ver que existen enteros *a, d* tales que:

*aj+dh= mcd(j,h)*=1

Para el ejemplo: *a* =61, *d*= -169. Por lo que existirá un entero *d* tal que:

$de \equiv 1 (mod\ h)$

Lo que se tiene en secreto es ***d*** esto representa su clave privada.

Cuando se tiene números negativos, se puede realizar lo siguiente

-169 ≡ -169+640 ≡ 471(mod 640) por lo tanto si elegimos d = 471 se seguirá verificando la relación

$de \equiv 1 (mod\ 640)$

## 1.5 Modelo encryptionstring

Este modelo como todos los anteriores se encuentran basados en un algoritmo, la función básica de este algoritmo es codificar la información, para que sea indescifrable por terceros. Este modelo toma el mensaje y una clave del usuario,

se realiza un acoplamiento de estas dos variables y como resultado se obtiene una cadena codificada.

Como podemos observar en la Figura 1-17. Modelo Encriptionstring, vemos que se ingresa un texto, lo separamos por caracteres, lo transformamos a código ASCII, colocamos una contraseña, sumamos los caracteres especiales tanto de la contraseña como del texto ingresado y tenemos caracteres ilegibles, a lo que se le denomina texto cifrado o codificado.

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Texto a codificar | **UNIVERSIDAD** | | | | | | | | | | |
| Caracteres del texto | U | N | I | V | E | R | S | I | D | A | D |
| Códigos ASCII | 85 | 78 | 73 | 86 | 69 | 82 | 83 | 73 | 68 | 65 | 68 |
| Contraseña Key | J | O | H | A | J | O | H | A | J | O | H |
| Caracteres de Key | 74 | 79 | 72 | 65 | 74 | 79 | 72 | 65 | 74 | 79 | 72 |
| Sumas de Códigos ASCII | 159 | 157 | 145 | 151 | 143 | 161 | 155 | 138 | 142 | 144 | 140 |
| En caracteres | Ÿ | | ‘ | — | | ¡ | › | Š | Ž | | Œ |
| Texto codificado | **Ÿ ‘— ¡›ŠŽ Œ** | | | | | | | | | | |

Figura 1- 17. Modelo Encryptionstring (Johanna Moya P. , 2014)

Se puede decir que la ventaja principal de este algoritmo es flexible para usar y muy intuitivo. En cuanto respecta a la seguridad, se lo define como muy segura.

- **Código de Encryptionstring en Visual Basic**

Como se puede observar en la Figura 1-18.Codigo de Encriptación Visual Basic, en la primera parte definimos constantes, pasa saber que método utilizar en la función encryptstring, cuando declaramos n, estamos obteniendo la clave del usuario, el siguiente paso a desarrollar es obtener la cadena de texto y esta la transformamos al código ASCII, cada caracter, para utilizar las funciones de encriptar o desencripta, según el parámetro en la función, si seleccionamos la función encriptar lo que realiza es sumar el texto más la clave carácter por carácter y entregar una suma que pueda realizar o pueda ser transformado a un código ASCII, y para desencriptar se realiza el proceso reverso. Para poder tener el texto plano o legible.

La ventaja de aplicar este código en visual basic es que es muy flexible de usar e intuitiva. Sin tener la máxima seguridad, es muy segura.

```
Public Function EncryptString(ByVal UserKey As String, Text As String, Action As Single) As String

Dim UserKeyX As String
Dim Temp As Integer
Dim Times As Integer
Dim i As Integer
Dim j As Integer
Dim n As Integer
Dim rtn As String

n = Len(UserKey)
ReDim UserKeyASCIIS(1 To n)
For i = 1 To n
UserKeyASCIIS(i) = Asc(Mid$(UserKey, i, 1))
Next
ReDim TextASCIIS(Len(Text)) As Integer
For i = 1 To Len(Text)
TextASCIIS(i) = Asc(Mid$(Text, i, 1))
Next
If Action = ENCRYPT Then
For i = 1 To Len(Text)
j = IIf(j + 1 >= n, 1, j + 1)
Temp = TextASCIIS(i) + UserKeyASCIIS(j)
If Temp > 255 Then
Temp = Temp - 255
End If
rtn = rtn + Chr$(Temp)
Next
ElseIf Action = DECRYPT Then
For i = 1 To Len(Text)
j = IIf(j + 1 >= n, 1, j + 1)
Temp = TextASCIIS(i) - UserKeyASCIIS(j)
If Temp < 0 Then
Temp = Temp + 255
End If
rtn = rtn + Chr$(Temp)
Next
End If
EncryptString = rtn
End Function
```

Figura 1- 18. Código de Encriptación Visual Basic (Moya, 2015)

## 1.6 Modelo chrtran

Para este modelo la mayoría de los expertos la denomina que es la más segura, ya que las probabilidades para poder descifrar los datos son del orden 255! ( ! = Factorial).

Un **factorial** es el producto de todos los números desde 1 hasta el numero factorial. Ejemplo 5! = 5*4*3*2*1= 120. La función de los factoriales es determinar las cantidades de combinaciones y permutaciones, y averiguar las probabilidades.

$$n! = n * (n-1)!$$

| n | n! | | |
|---|---|---|---|
| 1 | **1** | 1 | 1 |
| 2 | 2 × **1** | = 2 × **1!** | = 2 |
| 3 | 3 × **2 × 1** | = 3 × **2!** | = 6 |
| 4 | 4 × **3 × 2 × 1** | = 4 × **3!** | = 24 |
| 5 | 5 × **4 × 3 × 2 × 1** | = 5 × **4!** | = 120 |
| 6 | etc | etc | |

Tabla 1- 5. Factorial (Moya, Factorial, 2015)

CHRTRAN la función de este modelo es transponer caracteres, usa dos claves de 255 caracteres (ASCII) con esto genera un texto codificado de origen aleatorio. Selecciona cada carácter del texto, encuentra su posición en la primera clave, e intercambia este carácter por el carácter en la misma posición de la segunda clave.

## 1.7 Encriptación de 40 bits, 128 bits o 1024 bits

Cuando se habla de cifrado de 40,128 o 1024 bits, se está refiriendo a la clave que será utilizada para el cifrado simétrico que es utilizado para la protección de datos mediante SSL/TLS. Antiguamente existían unas leyes de exportación que limitaban fuera de EEUU el cifrado simétrico a 40 bits, pudiendo obtener cifrado de 128 bits mediante un cifrado especial. Por ello se habla certificados de 40 bits y certificado de 128. Para la encriptación de 40 bits se la conoce como llave secreta, para las de 128 bits se la conoce como la llave publica y llave secreta. En la actualidad ya se dispone en la gran mayoría de los sitios de internet utilizan encriptación 40-512, y para 128-1024 bits se utiliza para transacciones de alto riego.

### 1.7.1 Clase de llaves

Las llaves generalmente son una larga secuencia de números protegidos por un mecanismo común de autenticación como contraseñas, tokens o biométricos, como la huella digital.

Hay dos clases de algoritmos de encriptación basados en llaves:

- **Llave Pública**

  Algoritmo asimétrico (o de llave pública) utilizan una llave distinta para la encriptación y para la desencriptación, y ninguna de las llaves puede ser derivada a partir de la otra, con esta llaves pública se verifica, autentifica la identidad de un usuario, y se determina la integridad de los datos, para no repudio, o combinación de los mismos.

- **Llave Privada**

  Algoritmo simétrico (o de llave privada) utilizan la misma clave para encriptar y desencriptar. Cuando se maneja adecuadamente, las claves de firma privadas se pueden utilizar para proporcionar autenticación, integridad y no repudio.

## 1.8 Niveles de encriptación

Tenemos varios niveles de encriptación, para el cifrado de datos, pero las más comunes son nivel de enlace, nivel de transporte y nivel de red, arquitectura, direcciones IP[34].

### 1.8.1 Encriptación a nivel de enlace

Este nivel es una protección criptográfica la más transparente tanto para los controladores de los dispositivos como para las aplicaciones. La protección que posee solo afecta a un enlace individual lo que quiere decir que el paquete es encriptado por completo lo que incluye las direcciones de origen y destino lo que deja fuera de riesgo la comunicación. El problema que tiene en particular es que solo protege un enlace.

***Ejemplo:*** Si el mensaje debe atravesar más de un enlace, será vulnerable en el nodo intermedio y en el siguiente enlace en caso de que tampoco este protegido.

### 1.8.2 Encriptación a nivel de transporte y a nivel de red

El protocolo de seguridad de la capa de red (Network Layer Security Protocol) - (NLSP) y el protocolo de seguridad de la capa de transporte (Transport Layer Security Protocol) - TLSP esto permite a los sistemas comunicarse de forma segura por internet.

Los 2 protocolos están basados en el concepto de id de clave o key_id; esto se transmite sin encriptar junto con el paquete encriptado. Lo que permite controlar el comportamiento de los mecanismos de encriptado y desencriptado: especifica el algoritmo de cifrado, el tamaño del bloque de cifrado, el mecanismo de control de integridad usado, el periodo de validez de la clave entre otras cosas más. Además utiliza el mecanismo de administración de claves para intercambiar claves de ids.

TLSP está limitado a conexiones individuales tales como circuitos virtuales creados en TCP. Diferentes circuitos entre el mismo par de host pueden ser protegidos con diferentes claves. El segmento TCP completo es encriptado, lo que hace que este nuevo segmento es enviado al protocolo IP, con un identificado de protocolo diferente. Al recibir el paquete, IP envía el paquete a TLSP, que luego de descifrar y verificar el paquete, lo pasa a TCP.

[34] IP: Internet Protocol

### 1.8.3 IPsec- Arquitectura de seguridad IP

La función de IPsec está diseñada para proveer seguridad basada en criptografía, de alta calidad e interoperable para IPv4 e IPv6.

**Dirección IPv4:** es un número de 32 bits formado por 4 octetos en una notación decimal, separados por puntos. Los posibles valores de un octeto en una dirección IP van de 0 a 255(cada número de IPv4 representa 8 bits, o lo que es lo mismo 1 byte). Por lo tano están formadas en total por 32 bits o 4 bytes.

IPv4 está limitado a 4.3 mil millones de direcciones.

***Ejemplo: 192.168.7.1***

**Dirección IPv6:** surgió está dirección ya que IPv4 estaba "quedándose corto". Porque empezaban a acabarse las IPs que sirven para identificar a los miles de millones de equipos y dispositivos de las redes mundiales e Internet. Usando la matemática tenemos $2^{128}$ (IPv6 están basados en 128 bits) y así se encuentra el total de las direcciones IPv6 está compuesto por 8 secciones de 16 bits, separadas por dos puntos (:) cada sección es de 16 bits, se tiene 2 elevado a la 16 variaciones.

***Ejemplo: 2607:f0d0:4545:2:100:f8ff:fe22:65cf***

Los servicios de seguridad ofrecida incluyen control de acceso, integridad en comunicaciones sin conexión, autenticación del origen de datos, protección contra ataques de repetición, confidencialidad mediante encriptado, entre otro. Estos servicios son provistos en la capa IP ofreciendo protección para esta y las capas superiores.

### 1.8.4 Encriptación a nivel de aplicación

Este procedimiento no es transparente involucra al usuario en las tareas necesarias, aunque es el más flexible ya que el alcance y la intensidad de la protección puede ser diseñada para cubrir las necesidades específicas de cada aplicación.

- **Protocolo Telnet**

  Está es una de las áreas más críticas porque tienen la necesidad de prevenir que las contraseñas sean enviadas sin protección a través de internet. Una de las soluciones seria encriptar la sesión telnet pero la desventaja de esta solución es que se está utilizando la capacidad de procedimiento innecesaria ya que solo las contraseñas estarían protegidas.

  Se debe tener en cuenta que el encriptado requiere la distribución de las claves,

para lo cual se necesita algún tipo de autenticación.

El mecanismo de encriptado propuesto para telnet tiene integrada la opción de autenticación. Cuando se inicializa una conexión, los dos lados negocian que algoritmo de encriptado y autenticación será usado y por quien.

- **Servicio de correo electrónico**

  El funcionamiento de este servicio es para los usuarios, ya que esta parte es la más importante ya que desean que su información no este expuesta.

  Uno de los principales es PEM[35] es el estándar oficial de la familia de protocolos de TCP/ IP, PGP[36] y RIPEM basados en una implementación libre de RSA, habilitado con el consentimiento de los dueños de la patente. Todos estos utilizan un sistema simétrico para encriptado y un sistema de distribución de claves basado en los sistemas de criptografía asimétrica.

  Lo que cabe aclarar es que la seguridad del correo electrónico depende críticamente de la seguridad del sistema operativo subyacente. Cualquier sistema de correo debe ser ejecutado en un sistema final que esté protegido físicamente y electrónicamente.

- **Cifradores de Bloque**

  La función de un cifrador en bloque es dividir el texto en bloques relativamente largos, que normalmente son de 4 o 128 bits, y codificar cada bloque y es la clave de encriptado que determina el orden en el que se llevan a cabo la sustitución, el transporte y demás funciones matemáticas.

  El funcionamiento del cifrado de bloque tiene diferentes variantes. La más simple es el libro de códigos electrónicos ECB.

- **ECB (Electronic Code Book):** el funcionamiento de este cifrado es el que divide el mensaje en bloques de k bits, rellenando el último si es necesario y se encripta cada bloque.

[35] PEM: Privacy-Enhanced Electronic Mail

[36] PGP: Pretty Good Privacy la finalidad es proteger la información distribuida a través de internet mediante el uso de criptografía de clave pública.

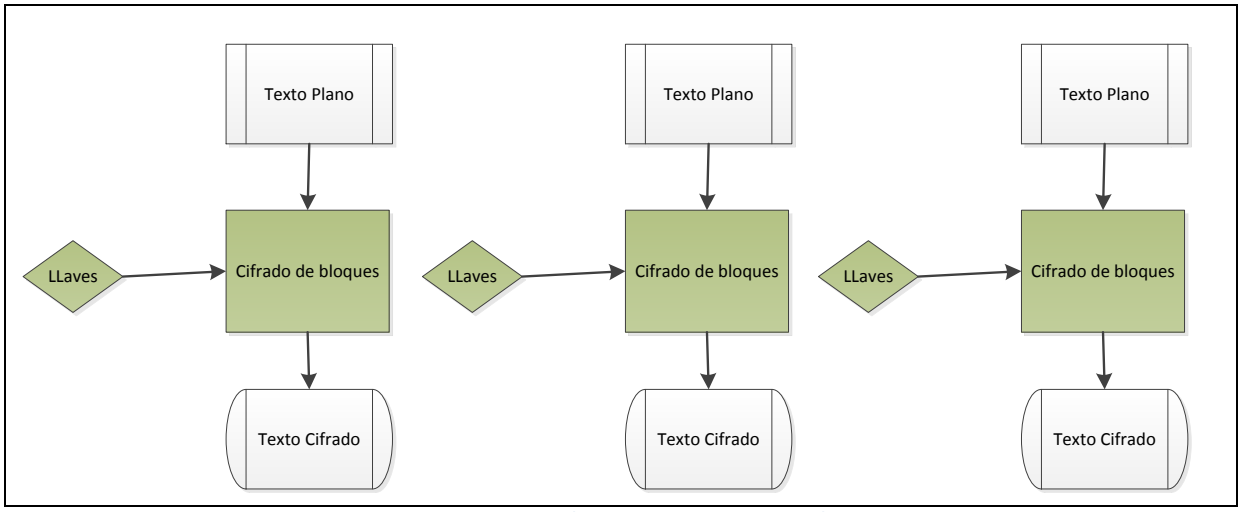



Figura 1- 19. ECB (Moya, ECB Cifrado, 2015)

Como se puede observar en la Figura 1-19 ECB Descifrado, nos indica que se genera una llave, luego ingresa el texto cifrado, que ingresa a un cifrado en bloques, que será del mismo tamaño de texto en plano, se coloca la llave y la información vuelve a ser legible.

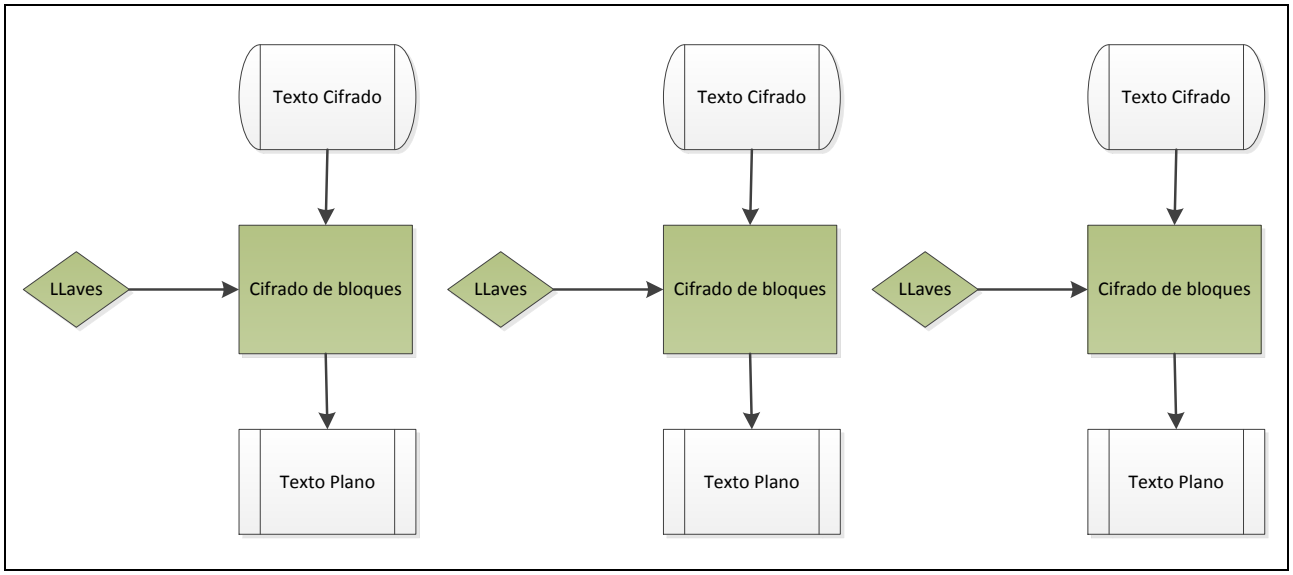



Figura 1- 20. ECB Descifrado (Moya, ECB Cifrado, 2015)

Otra manera para los algoritmos en bloque es CBC:

- **CBC (Cipher Block Chaining)** es un cifrado de encadenamiento de bloques el funcionamiento para el primer bloque de texto plano se aplica XOR[37] (operación matemática) con el bloque cifrado anterior antes de que sea cifrado. Lo que permite que cada bloque de texto cifrado dependa de todo el texto en claro y procesado hasta ese punto.

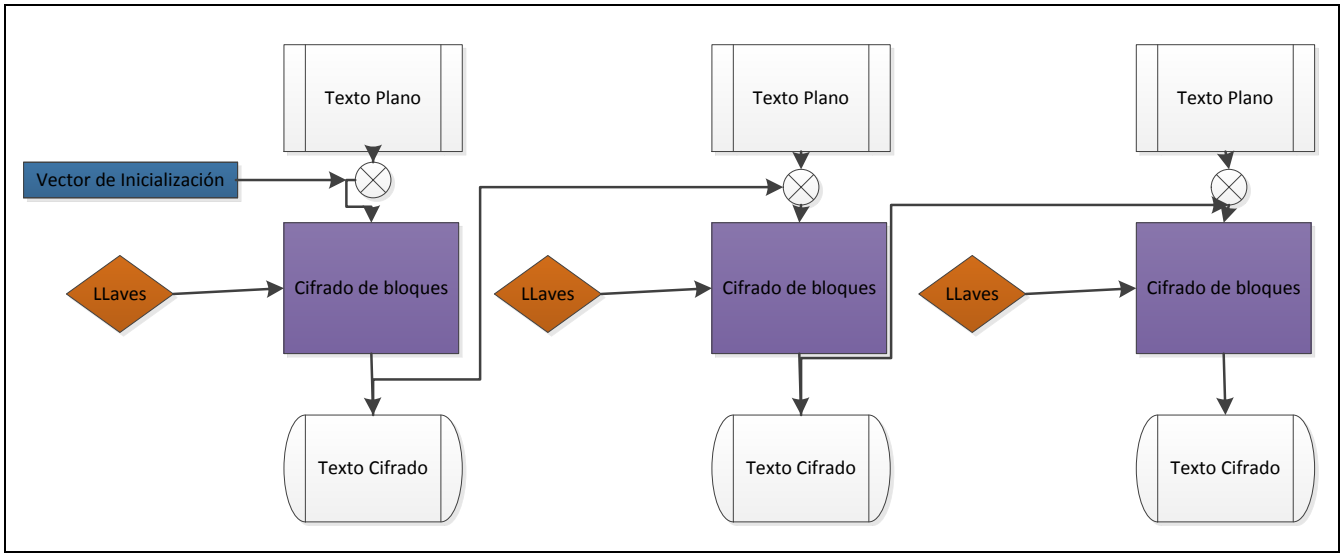



Figura 1- 21. CBC Cifrado (Moya, ECB Cifrado, 2015)

Se Observa en la Figura 1-21. CBC Descifrado, se utiliza una llave, para poder descifrar el texto en bloque que se tiene, pero este contiene una variable que se suma, es un vector de inicialización, con la ayuda de esta variable, permite que el texto se pueda descifrar más rápido y si se tiene la clave correcta, devuelve el texto plano. Caso contrario, regresa al cifrado en bloque y solicita la clave, mientras no se ingrese la clave correcta no se puede observar el texto plano, para estos casos si se controla el número de veces que intenta ingresar es decir un número máximo de intentos para poder ingresar y luego se bloquea, pero eso se debe controlar cuando se desarrolla la aplicación. Si no tiene esta restricción se puede intentar n veces el poder ingresar y obtener el texto descifrado.

[37] XOR: realiza la función booleana A'B + AB'. Se puede definir esta puerta como aquella que da por resultado uno, cuando los valores en las entradas son distintas.

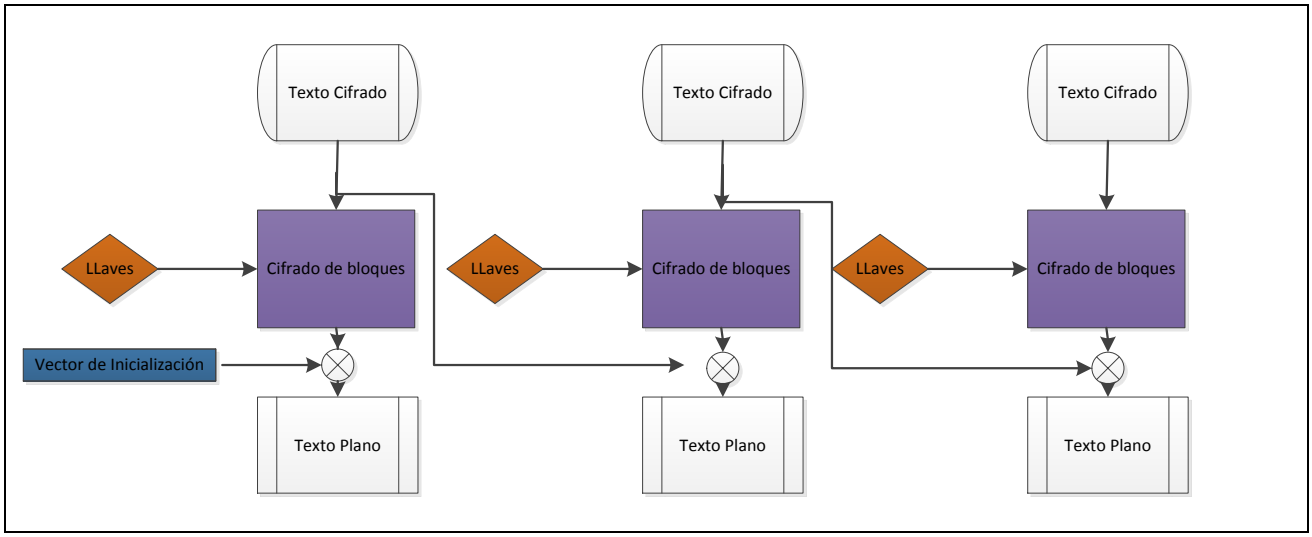



Figura 1- 22. CBC Descifrado (Moya, ECB Cifrado, 2015)

- **OFB (Output Feedback Mode):** este sistema emplea la clave de sesión para crear un bloque pseudoaleatorias grande que se aplica en o-exclusiva al texto en claro para generar el texto cifrador.

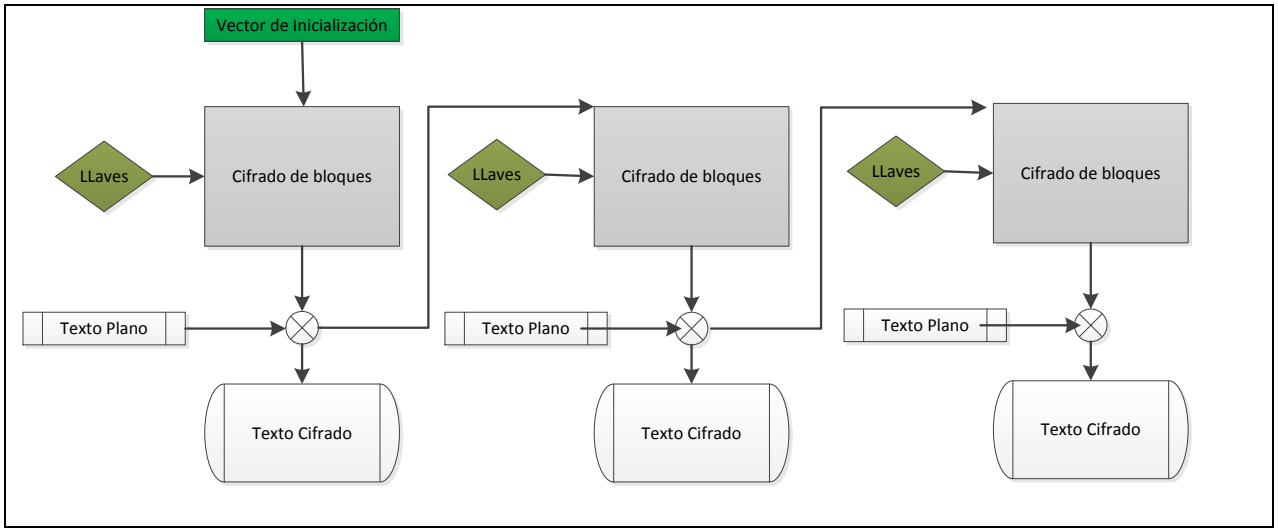



Figura 1- 23. OFB Cifrado (Moya, ECB Cifrado, 2015)

Como se observa en la Figura 1-24. OFB descifrado, se utiliza la llaves, el vector de inicialización, entra en una operación matemática para poder generar el texto de descifrado o también llamado vector de inicialización, realiza varias comprobaciones y si es correcto accede al texto, caso contrario regresa al punto de partida y tiene que realizar todo el procedimiento.

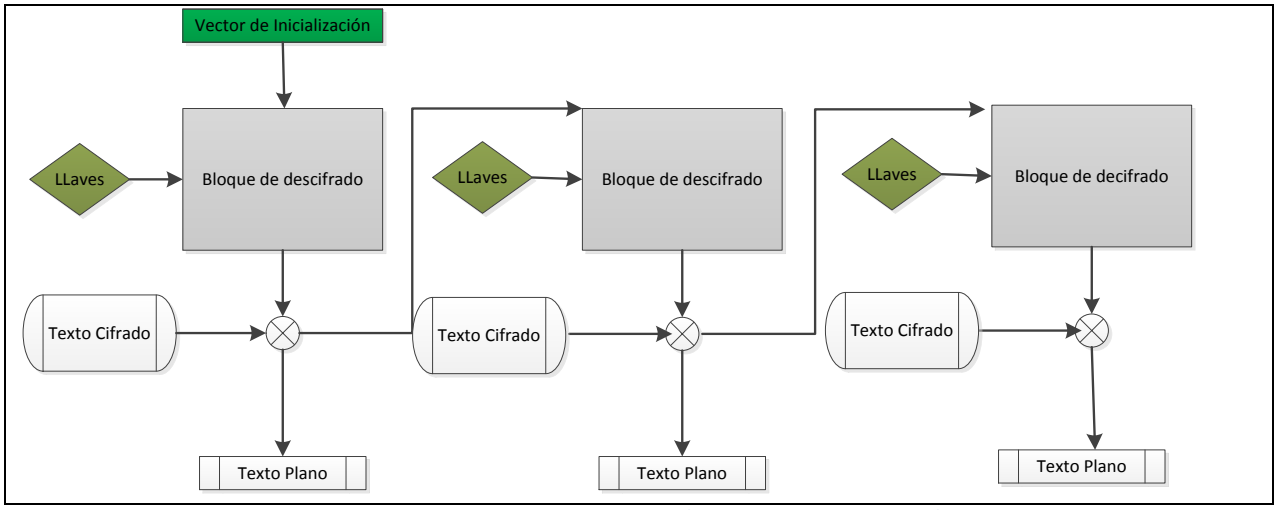



Figura 1- 24. OFB Descifrado (Moya, ECB Cifrado, 2015)

- **CFB (Cipher Feedback Mode):** en caso de que los mensajes sean muy largos se ocupa este algoritmo, su funcionalidad es igual a OFB

### 1.8.5 Cifradores de Flujo

La transformación sobre cada carácter del mensaje original soporta llaves con longitudes de 80 y 128 bits. Además de poder procesar hasta 8 flujos de bits en paralelo, este cifrado realiza operaciones elemento a elemento, es decir que el algoritmo de cifrado se va aplicando un elemento de información de MCIa con un elemento de la clave para obtener así el criptograma.

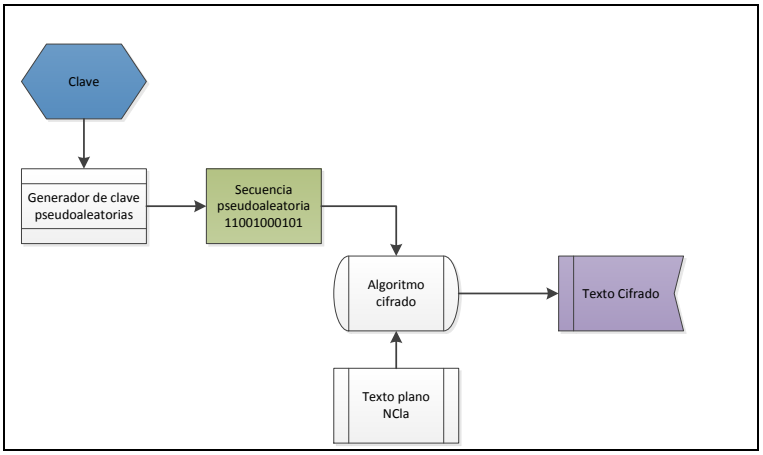



Figura 1- 25. Cifrado de flujo (Moya, ECB Cifrado, 2015)

- **Cifrado en flujo síncrono**

  Para realizar el proceso de cifrado correctamente tanto emisor como receptor deben tener señales de sincronización lo cual otorga la ventaja de que si se tiene un ataque en donde se inserten mensajes erróneos, es posible detectarlos ya que

ellos interrumpen la sincronía.

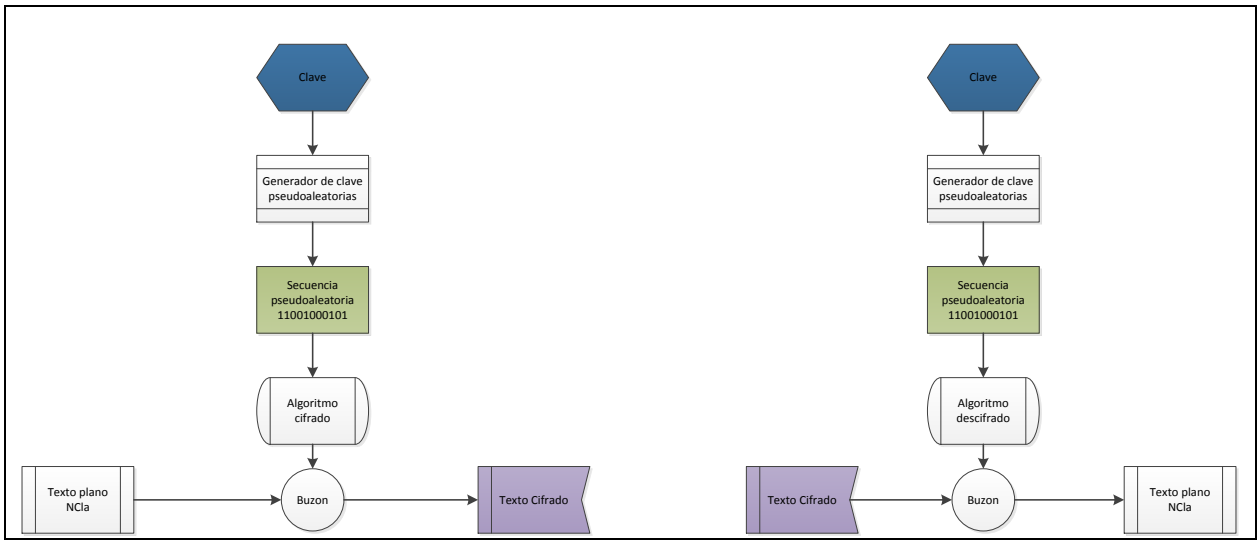



Figura 1- 26. Cifrado Síncrono (Moya, ECB Cifrado, 2015)

- **Cifrado en flujo autosincronizante**

  Para este caso no se necesita señales de sincronización entre el emisor y el receptor ya que en caso de pérdida de sincronía, esta se recupera debido a la retroalimentación. Su principal desventaja es que son altamente vulnerables a un ataque de inserción de mensajes, pero esto se puede evitar enviando mensajes adicionales de identificación de mensaje.

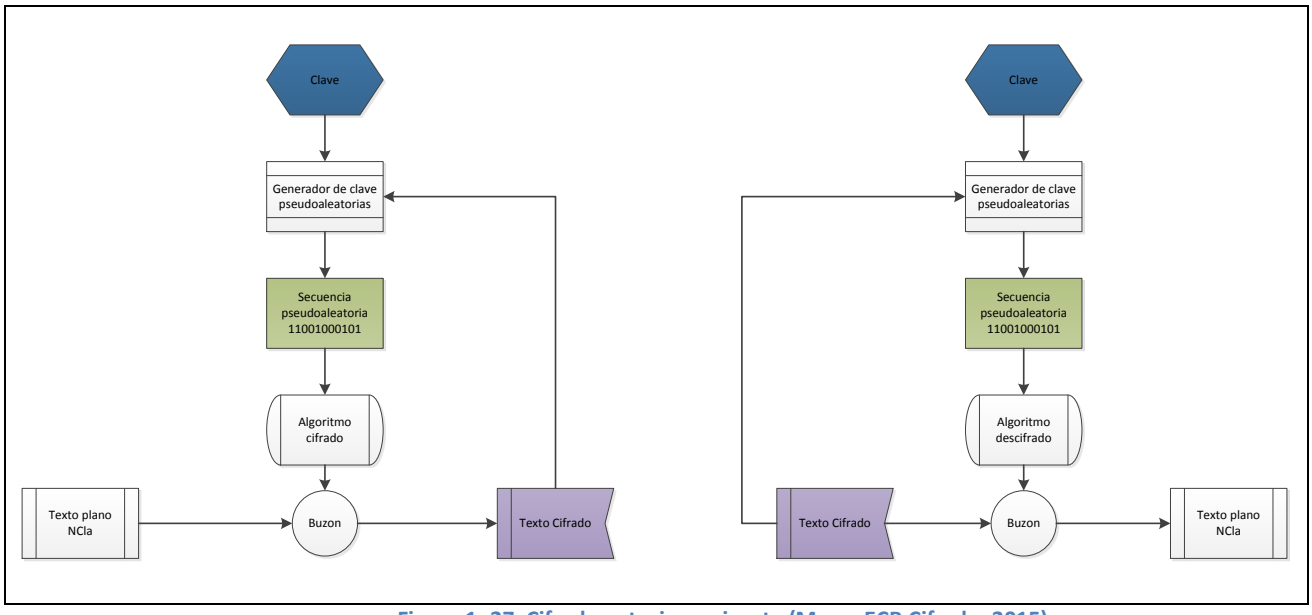



Figura 1- 27. Cifrado autosincronizante (Moya, ECB Cifrado, 2015)

## 1.9 Privacidad

Se dedicara un subcapítulo de privacidad debido a que se encontró datos muy importantes que vienen de ***Microsoft Research*** y nos dice que para el año 2025 internet tendrá 4700 millones de usuarios y tendremos 150 mil millones de cosas conectadas: computadoras, heladeras, televisores, autos, ropa, casa, etc. Todo estará conectado y controlado a través de internet entonces definiremos que es **Privacidad:** es el derecho y propiedad de la propia intimidad y mida privada.

- **Privacidad de la información en internet**

  Se define como la capacidad de controlar tanto la cantidad de información personal que se proporciona como quién tiene acceso a esta información en otras palabras es un derecho humano básico. Esta sirve de fundamento a la dignidad y a otros valores tales como la libertad de asociación y la libertad de expresión.

- **El derecho a la privacidad**

  A nivel internacional podemos encontrar la Declaración Universal de Derechos Humanos, la cual específicamente protege la privacidad territorial y de las comunicaciones. Artículo 12.

  "Nadie será objeto de injerencias arbitrarias en su vida privada, su familia, su domicilio o su correspondencia, ni de ataques a su honra o a su reputación. Toda persona tiene derecho a la protección de la ley contra tales injerencias o ataques"[38]

- **La evolución de la protección de datos**

  La primera ley de protección de datos en el mundo fue promulgada en el Estado Federado de Hessen en Alemania en 1970, con el advenimiento de la tecnología de la información fue seguido por las leyes nacionales en Suecia (1973), los estados Unidos de América (1974), Francia (1978). En las cuales se fijaron normas específicas que se ocupan del manejo de datos electrónicos. Estas normas describen a la información personal como datos a los que se les suministra protección, desde su recolección hasta su almacenamiento y difusión. La locución protección de datos varía en distintas declaraciones y leyes. Sin embargo, todas exigen que la información personal deba ser:

  ❖ Obtenida de manera justa y legal;

[38] Declaración universal de derechos humanos, artículo 12

- Utilizada solo para el propósito original especifico;
- Exacta y actualizada;
- Adecuada, relevante y no excesiva;
- Mantenida de forma segura;
- Destruida después de que se haya logrado su propósito

## 1.10 Transparencia

Se dice que transparencia en una persona es cuando se muestra tal como es y no tiene secretos. En un sentido similar para las organizaciones son aquellas que hacen pública su información, la cual nos permite tener confianza.

Además se dice que es la obligación de los sujetos dar a conocer públicamente la información derivada de su actuación, en ejercicio de sus atribuciones. Tiene por objeto generar un ambiente de confianza, seguridad y franqueza entre el gobierno y la sociedad, de tal manera que todos estemos informados y conozcamos las responsabilidades, procedimientos, reglas, normas y demás información generada por el gobierno público, en un marco de abierta participación social y escrutinio público.

En términos generales, la transparencia se refiere al acceso que tienen los ciudadanos a la información interna; el alcance, la precisión y la puntualidad de dicha información.

La transparencia busca formar usuarios sensibles, responsables y participativos, que conozcan y ejerzan sus derechos y obligaciones, y colaboren activamente en el fomento a la integridad y combate a la corrupción.

### 1.10.1 Los beneficios de la transparencia

Se tiene varios beneficios cuando somos transparentes, tenemos democracia, participación, gobernanza y armas contra la corrupción.

En el siguiente Figura1-28. No democracia, podemos observar que somos seguidos la pista, saben lo que hacemos a donde ingresamos, que enviamos, con quien conversamos de que hablamos, que hacemos. Siempre controlados , pero en todo caso tendríamos muchos beneficios si jugáramos para ambos bandos de la misma manera, transparencia para los dos lados, así tendríamos democracia y responsabilidad de todo lo que manejamos y sabemos.


Figura 1- 28. No democracia (Assange, 2012)

- **Democracia, responsabilidad y participación**

  Se dice que la información por sí sola no implica poder, pero es un primer paso fundamental en el ejercicio del poder político y económico. El usuario solo puede tener una participación real en el proceso democrático cuando tiene información sobre las actividades y políticas de su gobierno y cuando sabe que beneficios y servicios tiene derecho y si está recibiendo lo que debería.

- **Gobernanza**

  Con la transparencia y claridad permitirá fomentar las capacidades de los pobres de desempeñar un papel a la hora de formular y poner en práctica políticas, influencias las decisiones que afectan a sus vidas y alentar a los responsables de las políticas y la toma de decisiones a que ejerciten su poder por el bien común.

- **Arma contra la corrupción**

  El acceso a la información libre y garantizada permite que los usuarios, medios de comunicación y organismos de aplicación de la ley utilicen documentos oficiales con el fin de sacar a la luz las cosas de corrupción y mala administración.

  Con una mayor transparencia aumentan las posibilidades de detectar prácticas corruptas.

## 1.11 Beneficios de encriptar

Debido a que la humanidad ha inventado una variedad de métodos para guardar nuestros secretos como se puede ver en este capítulo.

Cada uno de nosotros nos preguntamos si tenemos algo que esconder y nuestra respuesta será "Yo no tengo nada que ocultar"[39], en si lo que queremos decir con este dicho es que "No creemos que nadie esté interesado en hurgar nuestra información"[40]. Pero creemos nosotros que estamos preparados para enseñar nuestras cartas, fotos y documentos a nuestra familia. Y aún más estamos preparados para mostrar al mundo nuestra información. Talvez diremos que nuestra información no contiene nada malo. Pero qué hay de nuestras tarjetas de crédito, nuestros PIN's, nuestras claves, etc. Siempre encontraremos gente que quiera hacernos daño con esta información en su poder. Es por esa razón que hemos encontrado varios beneficios que a continuación se dan a conocer.

- Cuando se envía nuestra información encriptada tenemos la seguridad que dicha información solo llegara a la que va destinado el correo y pueda leerlo.
- Sirve para comprobar la integridad del mensaje
- Se puede comprobar automáticamente la autenticidad
- Es transparente para el usuario, lo único que debe hacer es saberse la clave.
- Garantiza la total privacidad de la información enviada

[39] Johanna Moya, Andrés Escobar primera respuesta cuando empezamos la disertación de grado
[40] Johanna Moya, Andrés Escobar aclaración de nuestro punto de vista en la fase inicial

# 2. CAPITULO II – HERRAMIENTAS Y METODOLOGÍAS

En este capítulo se señalara la metodología que utilizaremos para el desarrollo de la disertación de grado, los ciclos respectivos, los roles, los elementos, herramientas que se utilizara, gestores de base de datos y justificaciones. Con lo explicado en el marco teórico tratada en el capítulo I podremos aplicar la encriptación, en donde un emisor quiere enviar información al receptor de manera segura, se lo pueda realizar, sin que terceros sepan cual es la información que está viajando. Mediante algoritmos, ya vistos anteriormente.

## 2.1 Metodología scrum

Está metodología fue aplicada por primera vez por Takeuchi y Nonaka en el año de 1986 en la cual se dio a conocer una nueva forma de gestionar proyectos en donde la agilidad, flexibilidad, y la incertidumbre se hacían presentes, ellos se basaban en empresas como Honda, HP, Canon entre otras de que el producto no seguía unas fases en las que había un equipo especializado en cada uno de ellas, sino que se partía de un requisito muy general y lo realizaba un equipo multidiciplina que trabajaba desde el principio hasta el final.

Se comparó esta forma de trabajo en equipo, con la colaboración que hacen los jugadores de rugby y la utilización de la formación denominada scrum.

Scrum al ser una metodología de desarrollo ágil tiene como base la idea de creación de ciclos breves para el desarrollo, que comúnmente se llama iteraciones y que en scrum se llamaran "Sprints[41]".

La metodología scrum es adecuado para empresas en las que el desarrollo de los productos se realiza en entornos que se caracterizan por tener:

1. **Incertidumbre:** se plantea el objetivo que se quiere alcanzar sin proporcionar un plan detallado del producto.
2. **Auto- organización:** los equipos son capaces de organizarse por sí solos, no se necesita roles.
3. **Control moderado:** se establecerá un control suficiente para evitar descontroles.
4. **Transmisión del conocimiento:** todo el mundo aprende de todo el mundo.

### 2.1.1 Ciclo de desarrollo Scrum

- **Concepto:** se define de forma general las características del producto.
- **Especulación:** se hacen disposiciones con la información obtenida y se establecen los límites que marcara el desarrollo del producto

[41] Es un documento, donde se presentan resultados del ciclo de desarrollo

- **Exploración:** se incrementa el producto en el que se añaden las funcionalidades de la fase de especulación.
- **Revisión:** se revisa todo lo que se ha construido y se contrasta con el objetivo deseado.
- **Cierre:** se entrega en la fecha acordada una versión del producto deseado.

Scrum se gestiona los Sprints a través de reuniones diarias.

Como se puede observar en la Figura 2-1. Ciclo de desarrollo scrum, es una metodología 'ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión de la empresa. Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación. El ciclo de desarrollo Scrum, va desde el concepto, especulación, exploración, revisión, cierre parcial, para seguir el procedimiento hasta terminar el proyecto.

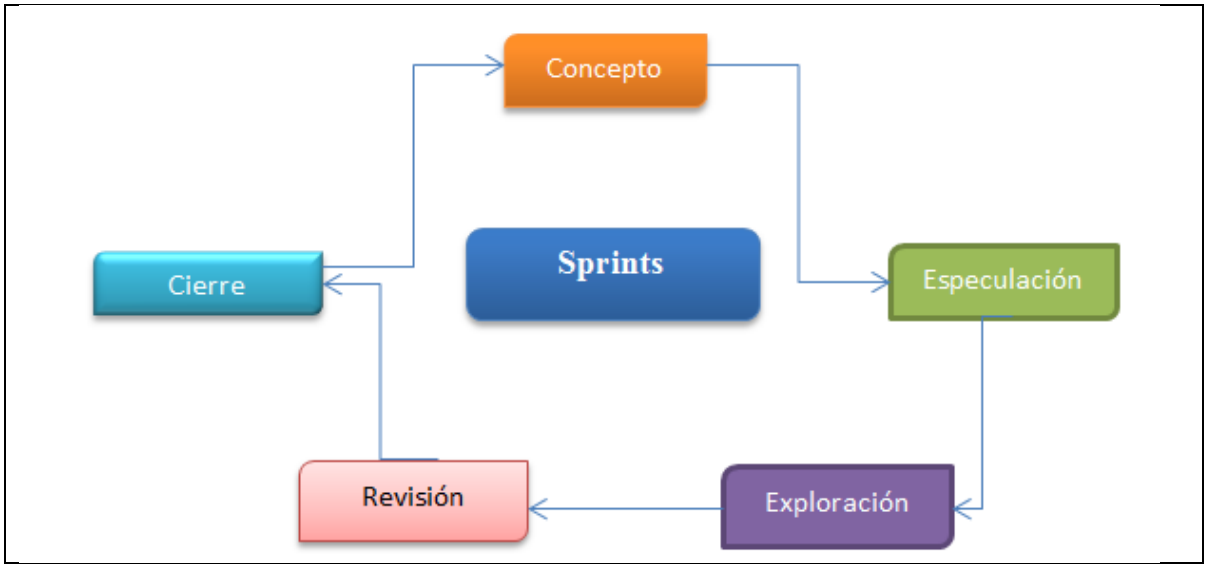



Figura 2- 1. Ciclo de desarrollo scrum (Moya, Ciclo de desarrollo Scrum, 2015)

### 2.1.2 Las Reuniones

a) **Planificación del backlog:** se define un documento en que se reflejan los requerimientos del sistema por prioridades se debe obtener un sprint backlog, que es la lista de tareas y que es el objetivo más importante de sprint.

b) **Seguimiento del sprint:** la fase de reuniones diarias en la que se realizan 3 preguntas principales.

1. ¿Qué trabajo se realizó desde la reunión anterior?
2. ¿Qué trabajo se hará hasta una nueva reunión?

3. Inconvenientes que han surgido y que hay que solucionar para continuar.

**c) Revisión del sprint:** cuando se finaliza el sprint se presentan los resultados finales y un demo o versión esto ayudara a mejorar el feedback con el cliente.

### 2.1.3 Los Roles

**A. Los cerdos:** son las personas que están comprometidas con el proyecto y el proceso de scrum

**A.1 Product Owner:** es la persona que toma las decisiones, conoce al cliente y su visión, escribe las ideas del cliente las ordena y las coloca en el product backlog

**A.2 Scrum Master:** Debe comprobar que el modelo y la metodología funcional

**A.3 Equipo de Desarrollo:** suele ser un grupo pequeño y tienen autoridad para organizar y tomar decisiones.

**B. Las Gallinas:** no son parte del proceso scrum, pero es necesario que parte de la retroalimentación se salida del proceso y así poder revisar y planear cada sprint

**B.1 Usuarios:** destinario final del producto

**B.2 Stakeholders:** personas a quienes el proyecto les producirá un beneficio

**B.3 Managers:** toma las decisiones finales participación en la selección de los objetivos y de los requisitos.

### 2.1.4 Elementos de la metodología Scrum

- **Product Backlog**: lista de necesidades del cliente.

  Contendrá objetivos del producto, para cada objetivo se indicara el valor que le da el cliente, el costo estimado[42], se tendrá que indicar posibles iteraciones y liberaciones que se ha indicado al cliente, además debe incluir posibles riesgos e incluir las tareas necesarias para solventarlo

- **Sprint backlog:** lista de tareas que se realizan en un sprint.

  **Las historias de Usuario:** serán el resultado de la colaboración entre el cliente y el equipo e irán evolucionando durante toda la vida del ciclo. Se componen en tres fases denominadas **"Las 3 C"**

  **CARD:** Breve descripción escrita que servirá como recordatorio.

  **CONVERSATION:** es una conversación para asegurarse que se ha entendido todo bien.

  **CONFIRMATION:** test funcional para figar detalles que sean relevantes e indicar cuál va a ser el límite.

[42] **Estimación:** evaluación del coste de implementación en unidades de desarrollo representa el tiempo teórico( desarrollo/hombre) que se haya estimado al comienzo del proyecto

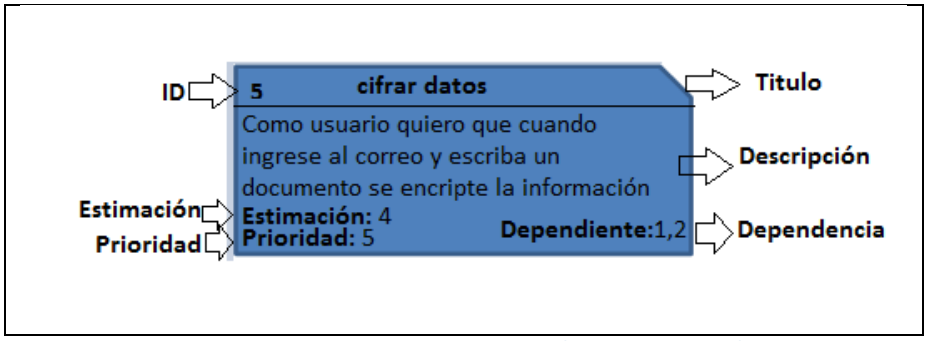



Figura 2- 2. Card Sprint Backlog Anverso (Escobar A. , 2015)

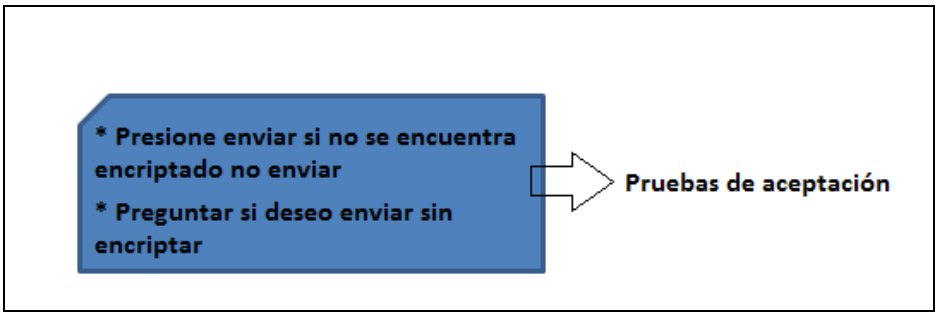



Figura 2- 3. Card Sprint Backlog Reverso (Escobar A. , 2015)

**Prioridad: Must** (M)= se debe completar el requerimiento, **Should** (S)= se debe completar, pero el éxito del proyecto no depende de ese requerimiento. **Could** (C)= se debería completar el requerimiento, **Would** (W)= se puede completar si sobra tiempo.

**Dependencia:** una historia del usuario, no debería ser dependiente de otra historia.

- **Incremento:** parte añadida o desarrollada en un sprint, es una parte terminada y totalmente operativa.
  Representa los requisitos que se han completado a una iteración y que son perfectamente operativos. Según los resultados que se obtenga, el cliente puede ir haciendo los cambios necesarios.

## 2.2 Herramientas de software

Se llama herramientas de software a los lenguajes de programación que se vaya a utilizar en el desarrollo, se tratara de utilizar los más apropiados y los que nos dé como resultado nuestra encriptación de datos en un aplicativo web, además se usara los gestores de bases de datos que es fundamental para el funcionamiento del aplicativo.

### 2.2.1 HTML[43]

Hace referencia al lenguaje marcado para la elaboración de páginas web. Es un estándar que sirve de referencia para la elaboración de páginas web en sus

[43] Hypertext markup language ( lenguaje de marcas de hipertexto)

diferentes versiones, define una estructura básica y un código denominado HTML para la definición de contenido de una página web, como texto, imágenes, videos, entre otros.

Formulario HTML es el elemento que permite recoger datos introducidos por el usuario.

El formulario posee los siguientes atributos:

- Atributo action: indica una dirección de correo electrónico a lo que envía el formulario, o la dirección del programa que se encarga de procesar el contenido del formulario.
- Atributo enctype: indica el modo en que será cifrada la información para su envió. Por defecto tiene el valor application/x-wwwform-url/encoded.
- Atributo method: indica el método mediante el que se transfieren las variables del formulario, su valor puede ser get o post.

**Ejemplo:**

```
<!DOCTYPE html>
<html>
<body>

<form action="demo_post_enctype.asp" method="post" enctype="multipart/form-data">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Figura 2- 4. HTML (Moya, HTML, 2015)

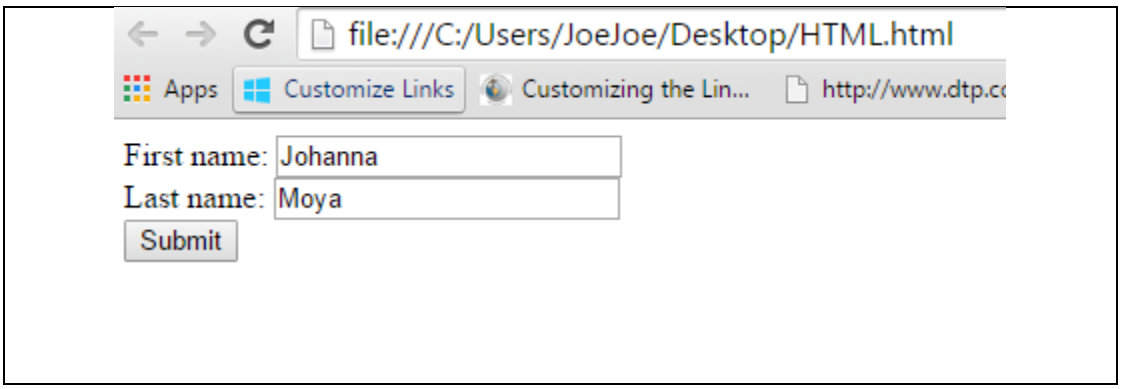



Figura 2- 5. Encriptar en HTML (Moya, HTML, 2015)

Como se puede observar en la figura 2-5. HTML, se usa una palabra específica en HTML, que es enctype y cifra el texto que se va ingresar y nos devuelve caracteres diferentes a lo que ingresamos.

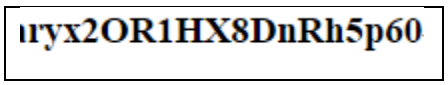



Figura 2- 6. Resultado de encriptación en HTML (Moya, HTML, 2015)

### 2.2.2 PHP

Lenguaje de programación de uso general de código del lado del servidor, por lo cual podemos utilizar el modelo de almacenamiento que se puede realizar en PHP, se encuentra la forma más sencilla para evitar el problema del cifrado, primero se debería crear un paquete de encriptación propio y utilizarlo en los scripts de PHP, con esto podemos utilizar extensiones de PHP que pueden ser de gran ayuda como Mcrypt, Mhash, Lo que realiza estas extensiones encripta los datos antes de insertarlos en la base de datos, y los desencripta cuando los obtiene.

- **Encriptación en 2 sentidos o simétrica**

  Se refiere, cuando se tiene un texto plano que el usuario quiere enviar al receptor, puede realizar la encriptación y la desencriptación, es decir obtener los dos sentidos del cifrado con la misma clave.

- **Encriptación en 1 único sentido o asimétrica**

  Significa que no se permite la desencriptación del texto encriptado previamente. Dado que recuperar la cadena original se hace imposible, esto nos permite que la encriptación asimétrica sea más segura que la encriptación simétrica.

  A continuación se coloca un ejemplo de cifrado y descifrado de texto en php para lo cual se crea una clase Encryter con dos métodos sencillos: Encrypt, Decrypt.

  El objetivo principal de la clase encrypter es fortalecer la seguridad de la encriptación, se tendrá una clave secreta con el atributo Key. Para la desencriptación se utilizara la misma clave secreta.

  Como se puede observar en la Figura 2-7.Encriptación en PHP, realiza la encriptación en la primera línea, podemos observar la variable de $texto. Y en la segunda fila se desencripta la variable $texto encriptado.

```php
/**
* Description of Encrypter
*
* @author JoeJoe
*/
class Encrypter {

    private static $Key = "dublin";

    public static function encrypt ($input) {
        $output = base64_encode(mcrypt_encrypt(MCRYPT_RIJNDAEL_256, md5(Encrypter::$Key), $input, MCRYPT_MODE_CBC, md5(md5(Encrypter::$Key))));
        return $output;
    }

    public static function decrypt ($input) {
        $output = rtrim(mcrypt_decrypt(MCRYPT_RIJNDAEL_256, md5(Encrypter::$Key), base64_decode($input), MCRYPT_MODE_CBC, md5(md5(Encrypter::$Key))), "\0");
        return $output;
    }

}

$texto = "estamos a punto de graduarnos";

$texto_encriptado = Encrypter::encrypt($texto);

if ($texto == $texto_original) echo 'Encriptación / Desencriptación realizada correctamente.';
```

Figura 2- 7. . Encriptación en PHP (Moya, PHP, 2015)

### 2.2.3 Visual Studio 2013

Es un entorno para sistemas operativos Windows, soporta múltiples lenguajes de programación tales como C++, C#, visual Basic.NET, F#, Java, Python, Ruby, además posee servicios y herramientas de última generación que puede utilizar para crear grandes aplicaciones destinadas a dispositivos, la nube, entre otras muchas opciones.

- **ASP.NET WEB**

  Es un framework para aplicaciones web que es desarrollado y comercializado por Microsoft, está diseñado para crear sitios dinámicos, aplicaciones web y servicios web XML. Fue creado en el año 2002 la cual tenía la versión 1.0 del .NET Framework.

  Además podemos acotar que es un programa que funciona en un servidor web y que sirve para la construcción de páginas web dinámicas. Y utilizaremos que sea dinámica ya que se actualiza en la base de datos y permite que el usuario haga cambios y este actualizado sus datos e información en el aplicativo.

  Se utilizara para realizar las pruebas de desarrollo en ASP. NET versión 4.51,

con los lenguajes presentados anteriormente no se pudo llegar a realizar el proyecto, porque encripta o solo frases, o el documento, o palabras claves.

Intentaremos que en este lenguaje no permita cifrar el mensaje que deseamos enviar y podamos aplicar alguno de los algoritmos ya tratados anteriormente.

Además este lenguaje hoy por hoy tiene un sistema de pertenencia de ASP.NET, incluye un nuevo sistema de scaffolding[44], compatibilidad con los servidores web, nuevas plantillas de proyecto, mejoras de rendimiento.

## 2.3 Gestores de base de datos

Es un sistema de software que permite la definición de base de datos, así como la elección de las estructuras de datos necesarios para el almacenamiento y búsqueda de los datos, ya sea de forma interactiva o a través de un lenguaje de programación. Un SGBD[45] relacional es un modelo de datos que facilita a los usuarios describir los datos que serán almacenados en la base de datos junto con un grupo de operaciones para manejar los datos.

- **SQL SERVER 2008**

  Es un gestor de datos creado por Microsoft y tiene varias características entre ellas es que es muy potente, soporta procedimientos almacenados, incluye un entorno grafico de administración, permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y los terminales o clientes de la red solo acceden a la información.

- **MySQL**

  Encriptación de datos en MySQL en ocasiones no es suficiente encriptar las comunicaciones. Si nuestros datos son altamente sensibles, queremos encriptar el mismísimo almacenamiento físico.

  **Contraseñas:** se dice que las contraseñas de nuestros usuarios, constituyen un dato critico que no queremos almacenar en "claro".

  Además tampoco nos gustaría que el administrador de la base de datos tenga la opción de conocer las contraseñas, para ello se debe guardar las contraseñas encriptadas, pero en lugar de utilizar algún algoritmo de encriptación que con el

[44] Scaffolding: es un método para construir aplicaciones basadas en base de datos.
[45] SGBD: DataBase Management System

tiempo y recursos se podrá desencriptar, las contraseñas deben o preferiblemente ser almacenadas en un hash.

Almacenado un hash de la contraseña, la única forma de romperlas será probando todas las entradas posibles hasta dar con una que genere el mismo hash. Para lo cual se requiere de bastante tiempo y recursos.

**MySQL** tiene varias funciones de hash: PASSWORD, OLD _PASSWORD, ENCRYPT, SHA1 y MD%

**Para la inserción de una contraseña seria:** INSERT INTO USUARIO (nombre, clave) VALAUES ('Johanna', PASSWORD('escobar')); de esta forma, la tabla almacena el hash, no la clave.

Para el caso de aplicaciones informáticas, es preferible que el hash se realice en el cliente, evitando así que la clave en claro viaje por la red.

**Comprobación de la clave de un usuario seria:** SELECT * FROM usuario WHERE nombre = $nombre AND clave= PASSWORD($clave)

**NOTA:** Si se desea almacenar resultados de una función de cifrado que pueda contener valores arbitrarios de bytes, se usa una columna BLOB[46] en lugar de CHAR o VARCHAR para evitar inconvenientes potenciales con eliminación de espacios finales que puedan cambiar los valores de los datos.

- **Funciones de cifrado**

Con estas funciones a continuación se cifran y se descifran valores:

- **AES_ENCRYPT(str, key_str), AES_DECRYPT(crypt_str, key_str):** esta función permite el cifrador y descifrado de datos usando algoritmo AES ( Advanced Encryption Standard)

AES( Advanced Encryption Standard) era conocida anteriormente como "Rijndael" se usa un cifrado con una clave de 128- bits, pero se lo puede ampliar hasta 256 bits modificando las fuentes. Generalmente se elige el de 128 bits porque es mucho más rápido y al momento es seguro.

AES es un algoritmo a nivel de bloques, se usa relleno para cadenas de longitud impar y así la longitud de la cadena resultante puede calcularse como 16*(trunc(string_length/16)+1.

[46] **BLOB:** las columnas BLOB se tratan como cadenas de caracteres binarias, no tienen conjunto de caracteres y la ordenación y la comparación se basan en los valores numéricos de los bytes.

Si AES_ENCRYPT() y AES_DECRYPT() se las considera las funciones de cifrado criptográficamente más seguras y disponibles en MySQL.

- **DECODE (cript-str, pass_str):** descifra la cadena cifrada crypt_str usando pass_str como contraseña, crypt_str debe ser una cadena retornada de ENCODE()
- **ENCODE (str, pass_str):** cifra str usando pass_str como contraseña. Para descifrar el resultado, use DECODE().
- **DES_DECRYPT (str[ , key_str]):** descifra una cadena cifrada con DES_ENCRYPT(). En caso de error, está función retorna null

*****Esta función solo funciona si MySQL se configura con soporte SSL.

**MySQL** incluye soporte para conexiones seguras (cifrados) entre los clientes MySQL y el servidor, utilizando el protocolo SSL (Secure Sockets Layer).

La configuración de MySQL tiene la misión de ser tan rápida como le sea posible, así que no se usan las conexiones cifradas por defecto, en caso de hacerlo, haría que el protocolo cliente/servidor fuese mucho más lento.

MySQL permite que el cifrado sea activado para conexiones individuales. Puede escoger entre una conexión normal sin cifra, o una segura cifrada mediante SSL dependiendo de los requerimientos de las aplicaciones individuales.

## 2.4 Justificación de herramientas a utilizar

Se utilizara el método Scrum para poder desarrollar la aplicación para encriptar información en la transmisión de datos en un aplicativo de mensajería web, ya que esta metodología permite que nos compromete y entusiasma dado que se verá crecer iteración a iteración. Asimismo nos permite en cualquier momento realinear el software con los objetivos que tenemos para esta disertación de grado, ya que podemos introducir cambios funcionales o de prioridad en el inicio de cada nueva iteración sin ningún problema.

Esta metodología de trabajo nos promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que encontramos un ámbito propicio para desarrollar nuestras capacidades.

Utilizaremos además algoritmos ya desarrollados y lo acoplaremos a nuestras necesidades y que se vio en el capítulo 1, el algoritmo seleccionado es asimétrico que necesita de una clave pública y otra privada. Contaremos con estos algoritmos ya

estudiados y que nos servirán para ir aplicando en el cifrado y descifrado de los mensajes, con lo cual podremos darle una mayor calidad al software, se tendrá una versión funcional.

El desarrollo de la página web, se realizara en Visual Studio 2013 con ASP.NET 4.5.1, usaremos paginas master para el contorno interno, nos permite tener mayor productividad, ya que tiene la alta capacidad de reacción ante los cambios de requerimientos generados por nuestras necesidades o evoluciones que va teniendo nuestro proyecto.

Además utilizaremos como gestor de base de datos SQL SERVER 2008, estas dos herramientas serán muy importantes ya que la pagina que crearemos es dinámica lo que significa que la base de datos se actualizara y así el usuario podrá disfruta de información actualizada para los mensajes, como para las claves.

En resumen tendremos herramientas que nos permitan cumplir con nuestras expectativas, tendremos flexibilidad a cambios solicitados tanto internos como externos, una reducción de tiempo ya que se podrá utilizar las funcionalidades más importantes del proyecto antes de que esté finalizado por completo, una mayor calidad en el software, mayor productividad, maximiza el retorno de inversión, predicciones de tiempo nos permitirá conocer la velocidad media con la que vamos desarrollando la disertación, utilizaremos puntos historia, con lo que consecuentemente, sería posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad. Y una reducción de riesgos ya que nos permitirá despejar riegos eficazmente de manera anticipada.

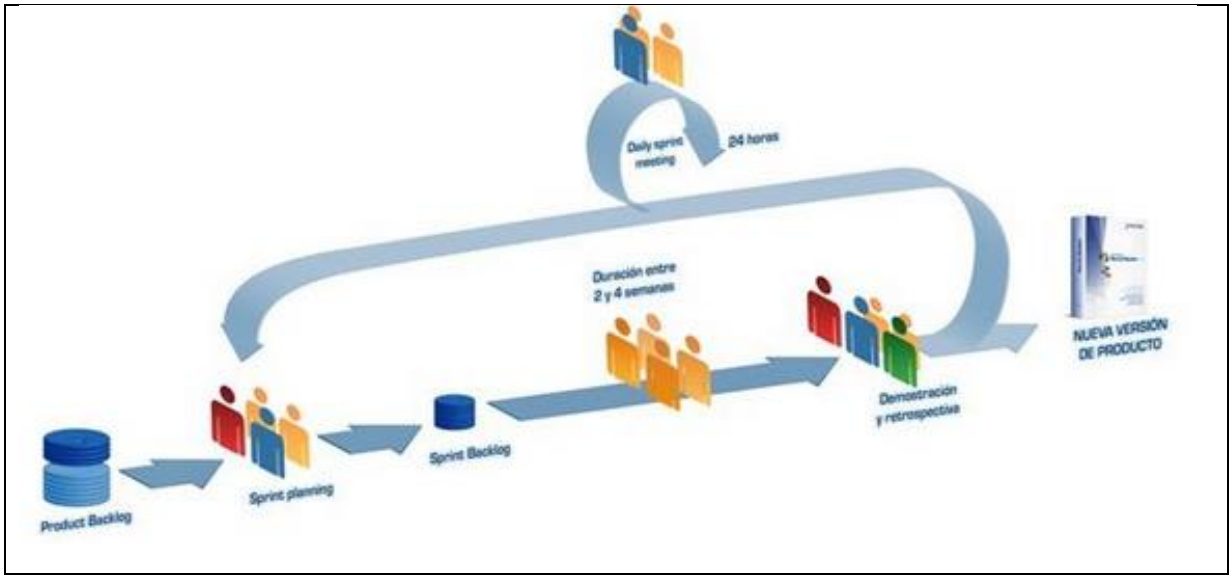



Figura 2- 8. Scrum (Softeng, 2010)

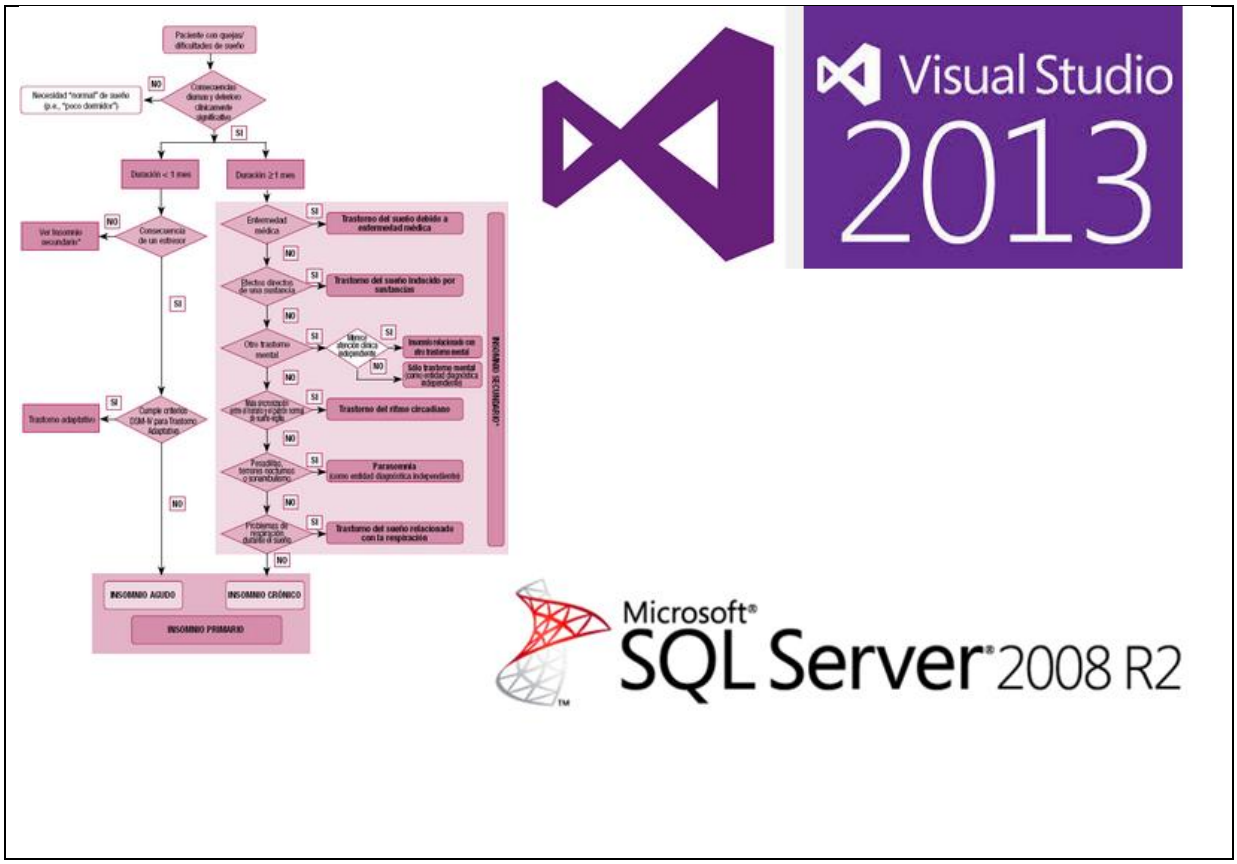



Figura 2- 9. Herramientas (Moya, 2015)

# 3. CAPITULO III –ITERACIONES DE ENCRIPTAMIENTO

En este capítulo, se describirá paso a paso como fue el desarrollo de la aplicación el lenguaje de programación que se utilizó, el gestor de base de datos que se usó para conectar y herramientas extras para el desarrollo.

Aplicamos la metodología SCRUM, y la primera fase fue el análisis.

| Id. | Nombre de tarea | Comienzo | Fin | Duración |
|---|---|---|---|---|
| 1 | Análisis del tema que vamos a desarrollar, mas recopilación de información | 10/03/2014 | 14/03/2014 | 5d |
| 2 | Análisis de los objetivos | 17/03/2014 | 17/03/2014 | 1d |
| 3 | Análisis del objetivo principal | 19/03/2014 | 19/03/2014 | 1d |
| 4 | Análisis de la informacion recopilada | 20/03/2014 | 26/03/2014 | 5d |
| 5 | Análisis en word del resumen de la informacion recopilada | 01/04/2014 | 04/04/2014 | 4d |
| 6 | Análisis, para las tareas a realizar durante el desarrollo | 07/04/2014 | 07/04/2014 | 1d |
| 7 | Análisis, de los roles a desempeñar durante el desarrollo del aplicativo | 10/03/2014 | 11/03/2014 | 2d |
| 8 | | | | |
| 9 | | | | |
| 10 | | | | |
| 11 | | | | |

Figura 3- 1. Análisis de los requerimientos para el desarrollo (Moya, SCRUM, 2015)

Después del análisis del tema y recopilación de la información, entramos a la definición de roles, el primero fue scrum master que es la persona que lidera el proyecto el que permite que se cumpla las reglas y procesos de la metodología. Se gestiona la reducción de impedimentos del proyecto.

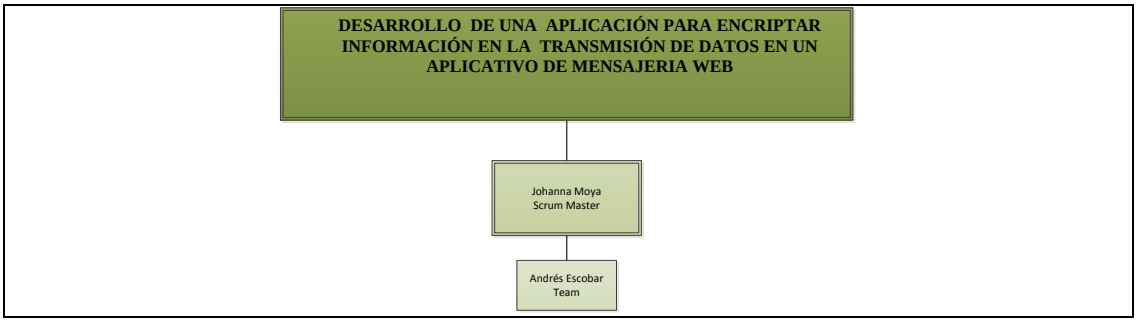



Figura 3- 2. Scrum Master 1era semana (Moya, SCRUM, 2015)

Cada día el equipo realiza una reunión de sincronización (tiempo de duración como mínima 45 min). El segundo paso a realizar por la persona que fue asignada como el Scrum Master es que inspecciona el trabajo que se está realizando en esta fase y además se entra al estudio de historia de la encriptación, los tipos de cifrado que existe y se valora los primeros algoritmos matemáticos para cifrar. Como parte fundamental del desarrollo es comprender como están estructurados y cuál es su función, para después redactar en los documentos previos, se crea las tarjetas de resumen para luego redactar en el informe final de la semana, la tarea de la siguiente semana es sentar la información en el documento Word que se presentara al final. El scrum master será cambiado cada dos semanas se realizara una revisión de lo hecho anteriormente y verifica las dependencias entre las tareas, el progreso hacia el objetivo de la iteración, y los obstáculos que pueden impedir este objetivo. Se realizara las debidas adaptaciones que permitan cumplir con el compromiso adquirido. Esto se realizara en 4 días en caso de encontrar cambios.

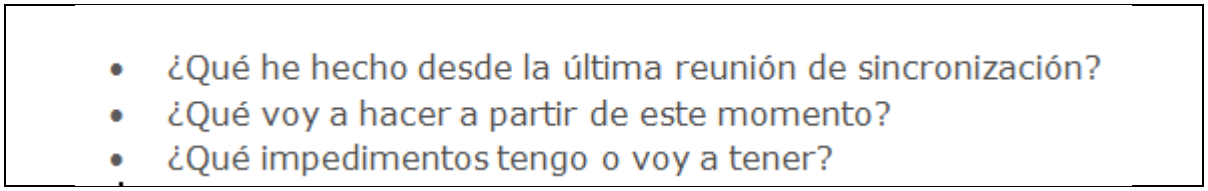



Figura 3- 3. Preguntas a realizar Scrum (Escobar A. , 2015)

Durante la iteración el scrum master se encarga de que el equipo pueda cumplir con su compromiso y de que no merme su productividad.

Para la fase de inspección y adaptación se realiza la reunión de revisión de la iteración, la cual consta de dos partes: Demostración y retrospectiva, para la primera el equipo presenta los requisitos completados en la iteración, en forma de incremento de la disertación para ser entregado con el mínimo esfuerzo. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, nosotros realizamos las adaptaciones necesarias de manera objetiva, lo que se puede decir es que desde la primera iteración se puede ir re planificando el proyecto. Para la segunda que es la retrospectiva, analizamos como ha sido nuestra manera de trabajar y cuáles son los problemas que nos ha ido impidiendo progresar adecuadamente, mejorando de manera continua nuestra productividad. Para la cual el Scrum Master se encargara de ir eliminando los obstáculos identificados.

Como se puede observar en la Figura 3-4. Análisis de algoritmos, se ha ido estudiando que algoritmos aplican en el tema que estamos resolviendo en nuestra disertación, entramos en el ámbito de analizar y estudiar para en un futuro poder aplicarlo, utilizando el que nos ayude en nuestro desarrollo.

| Id. | Estudio de algoritmos | Comienzo | Fin | Duración |
|---|---|---|---|---|
| 1 | Algoritmo HASH o de resumen | 02/05/2014 | 06/05/2014 | 3d |
| 2 | SHA-1 | 06/05/2014 | 07/05/2014 | 2d |
| 3 | MD2 | 07/05/2014 | 08/05/2014 | 2d |
| 4 | MD4 | 08/05/2014 | 08/05/2014 | 1d |
| 5 | MD5 | 09/05/2014 | 09/05/2014 | 1d |
| 6 | SHA-256 | 09/05/2014 | 12/05/2014 | 2d |
| 7 | RIPEMD-160 | 12/05/2014 | 12/05/2014 | 1d |

Figura 3- 4. Análisis de algoritmos (Moya, 2015)

Después de realizar esta etapa, en donde se encontraron varios algoritmos e información de cuál era la parte fundamental de cada una de estas, se entendió que se puede cifra por bloque o de carácter en carácter después de haber realizado esta etapa, y de ver si se encuentra complejidad para poder avanzar, se rectificó algunos procedimientos que se estaba haciendo mal y se trató profundizar en algoritmos que no se entendía para llegar a lo que era utilización de llaves tanto públicas como privadas.

Para cuando termino este ciclo, se entró a profundizar en las técnicas matemáticas, de cómo se va desarrollando cada algoritmo y que es lo que hace por debajo de nuestros algoritmos ya estudiados, con esto sumamente claro se dio paso a la siguiente etapa que era modelos de encriptación que fue fácil ya que con lo analizado anteriormente con nuestros resúmenes e informes finales de cada semana, se pudo aplicar en papel lo que tratábamos de realizar, entramos a la fase de inspección y verificación de cumplimento de objetivos, observamos que en el análisis de algoritmos, técnicas matemáticas nos tomó aproximadamente un mes y medio. Entramos a la fase de inspección que fue hecha por nuestro tutor, se realizó cambio que no fueron profundos simplemente de forma en cuanto era la historia, y organización de los algoritmos y la interpretación matemática en el documento final.

## 3.1 Etapa de inicialización en el desarrollo del aplicativo

A la par del análisis de la información recopilada, empezamos a realizar el levantamiento de requerimientos para el desarrollo de nuestro aplicativo, así podemos ir mezclando el conocimiento y aplicando a la práctica.

- Primer intento se realizó en HTML, la creación de la portada el inicio de sesión.

| Id. | Inicializacion del desarrollo del aplicativo HTML | Comienzo | Fin | Duración |
|---|---|---|---|---|
| 1 | Definicion de necesidades | 01/08/2014 | 05/08/2014 | 3d |
| 2 | Analisis de requerimientos | 05/08/2014 | 07/08/2014 | 3d |
| 3 | Diseño | 08/08/2014 | 12/08/2014 | 3d |
| 4 | Codificacion | 12/08/2014 | 18/08/2014 | 5d |
| 5 | Pruebas | 19/08/2014 | 21/08/2014 | 3d |
| 6 | Validacion | 21/08/2014 | 29/08/2014 | 7d |

Figura 3- 5. HTML encriptación (Moya, 2015)

Como conclusión del primer intento que realizamos en HTML es que para este encriptar código HTML está mal dicho, ya que no es la forma correcta de referirse a esto, pues aquí no usamos ninguna clave para esconder ningún mensaje, sino que utilizamos JavaScript que accede a nuestro HTML de otra forma y que es algo más complejo para leer.

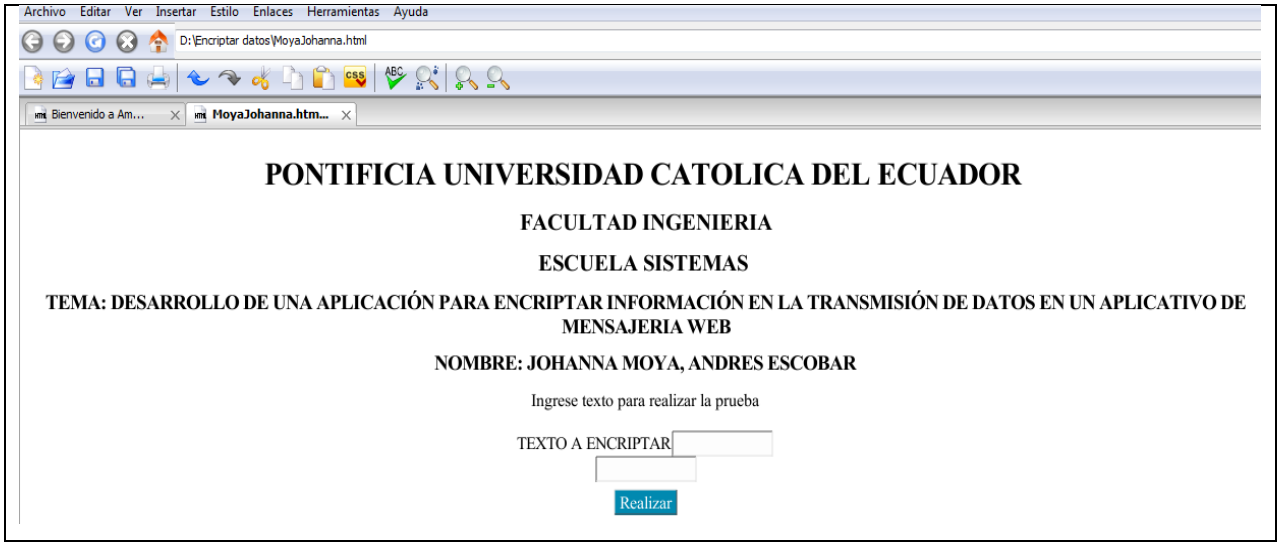



Figura 3- 6. HTML (Moya, HTML, 2015)

Incluso un usuario experimentado en tecnología web no tardaría más de un minuto en ver el mensaje en claro. Como podemos observar en la Figura3-6. Encriptar, lo que hacemos es utilizar un script mas no algún comando especial.

```
<script>
<!--
document.write(unescape("hola"));
//-->
</script>
```

Figura 3- 7. Encriptar HTML (Moya, 2015)

Realizamos un estudio de análisis de requerimientos en cuanto a la página web, para ver si por este medio podíamos atacar y poder realizarlo con HTML o PHP, nuestro segundo intento fue PHP pero primero les indicamos los requerimientos WEB.

## 3.2 Análisis de requerimientos

**3.2.1 Web Services Business Model:** en este diagrama podemos encontrar como el proveedor de servicios se enlaza al igual que lo hace el usuario y porque procesos pasa.

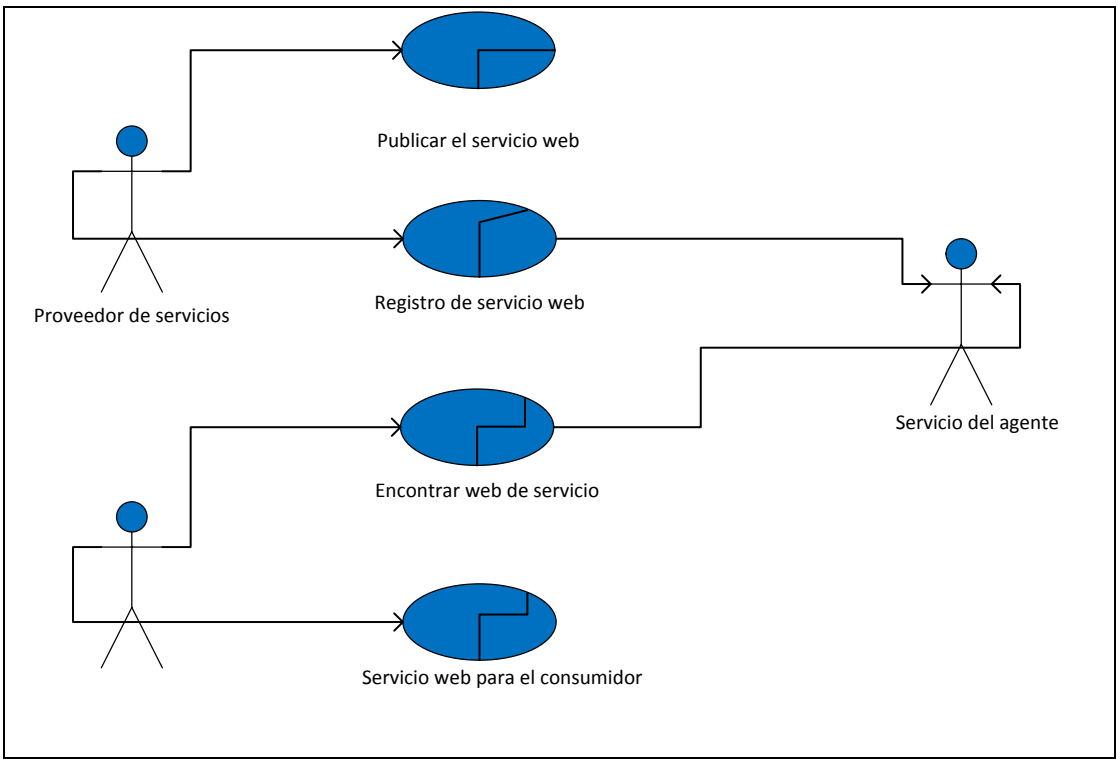



Figura 3- 8. Requerimiento WEB (Moya, 2015)

En esta Figura 3-8. Se entiende que el proveedor se enlaza con el usuario pero el

algoritmo interno no tiene nada que ver con este proceso, así que no aplica que exista algún método especial mientras la información viaja al receptor.

**3.2.2 Conceptos fundamentales del servicio web:** Observamos que el agente solo encuentra el solicitante de servicio, se enlaza al servicio de proveedor y publica, caso contrario regresa al agente y no hace nada. En cuanto a utilización de algún agente tampoco se puede utilizar en la implementación ya que depende únicamente de lo que hace internamente el emisor, ya que el receptor solo recibe información.

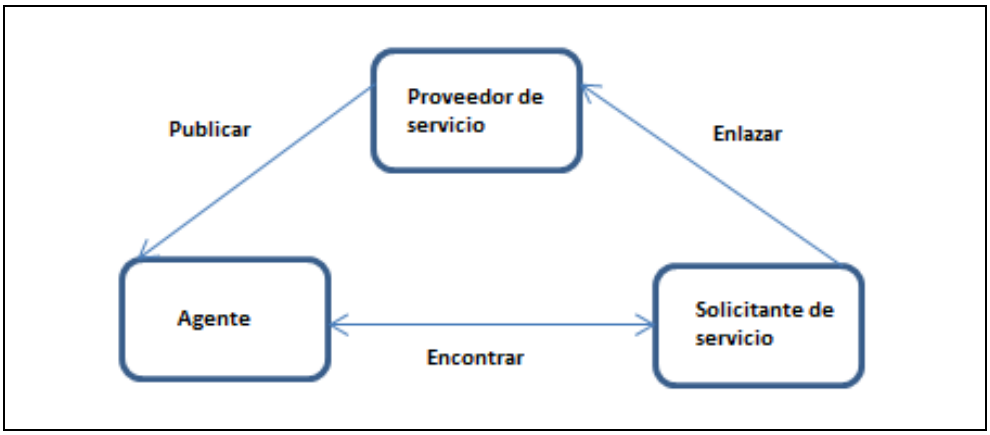



Figura 3- 9. Conceptos fundamentales del servicio web (Moya, 2015)

| Servicio | Estándar |
|---|---|
| Publicar | UDDI |
| Encontrar | UDDI, WSDL, DISCO |
| Enlazar | WSDL, SOAP |

Tabla 1. Servicio-Estándar Conceptos fundamentales del servicio web

**UDDI[47]:** Define un modo de publicar y encontrar información sobre servicios Web, tiene 2 funciones la primera es un protocolo basado en SOAP, que define como se comunican los clientes. La segunda es UDDI con registros es un conjunto en particular de registros duplicados globalmente. A que nos referimos es que funciona como un directorio que proporciona un mecanismo para localizar y registrar servicios web en Internet.

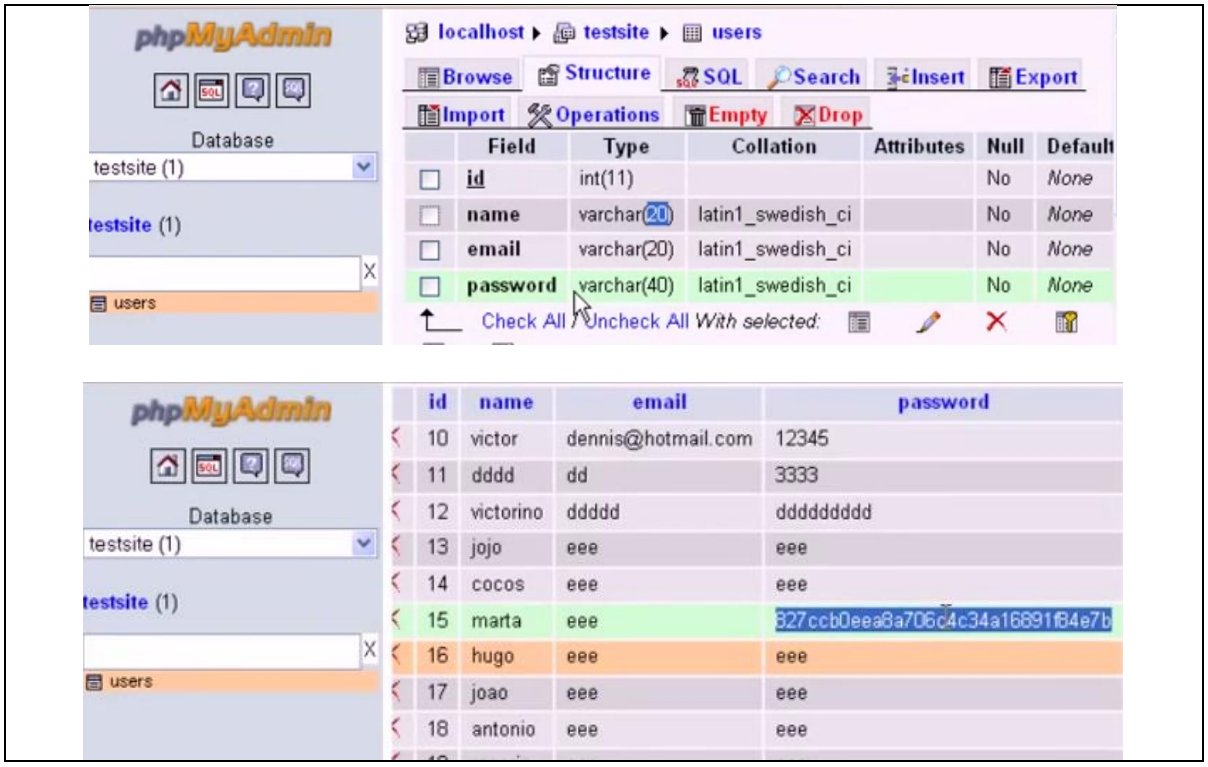
**WSDL[48]:** Es una especificación estándar basada en XML para describir servicios web. Un formato XML describe servicios de red como un conjunto de puntos finales que operan en mensajes que contienen información orientada a documentos u orientados a procedimientos. En resumen WSDL es un lenguaje de marcado que describe el servicio web.

[47] **UDDI:** Universal Description Discovery and Integration
[48] **WSDL:** Web Services Description Language

**DISCO**[49]**:** Es usada para descubrir la URL de servidores web XML localizados en un servidor web y guarda los documentos relacionados con cada servicio XML en un disco local.

**SOAP**[50]**:** Es un protocolo ligero el cual nos permite el intercambio de información en un entorno descentralizado y distribuido. Un mensaje SOAP es una transmisión de información desde un emisor a un receptor. Además se puede combinar para realizar los patrones de petición/respuesta SOAP es independiente del transporte, pero se realiza con mayor frecuencia a través de HTTP con el fin de ejecutar con la infraestructura de internet existente. SOAP permite la unión y el uso de servicios web descubiertas mediante la definición de una ruta de mensajes para los mensajes de enrutamiento. SOAP se utiliza para consultar UDDI para los servicios web.

- Pero antes de aplicar el segundo procedimiento a la par realizábamos el Fundamento Teórico que era simplemente aplicar lo que ya se estaba haciendo en el camino, que era el ciclo de desarrollo, las reuniones lo roles y los elementos de la metodología Scrum. Utilizando PHP y un gestor de base de datos MySql para realizar un cifrado que lo habíamos estudiado es MD5.




[49] **DISCO:** The Web Service Discovery Tool
[50] **SOAP:** Simple Object Access Protocol


Figura 3- 10. PHP, MySql MD5 (Moya, 2015)

En la Figura 3-10. PHP, MySql MD5, se tiene la base que nos permite ingresar al aplicativo.

Incluso nos permite cifrar el texto que se encuentra en la parte interna del aplicativo, pero no lo que es nuestro objetivo es que el mensaje que queremos enviar al receptor sea encriptado.

```
if ($count != 0) {

    echo "This name is already registered! Please type ano
} else {

    $passwordmd5 = md5($passwd);
    mysql_query("INSERT INTO users(name,email,password) VA
    echo "You have succefully registered!";
```

Figura 3- 11. Password Encriptado (Moya, 2015)

Como se puede observar en la Figura 3-11. Password Encriptado, lo que nos permite es encriptar la clave en el gestor de base de datos.

| ID | NAME | EMAIL | PASSWORD |
|---|---|---|---|
| 58 | balboa1 | balboa1@hotmail.com | 84e0991a380d2a451e9a7787e56e2b53 |
| 59 | rockybalboa | rocky@hotmail.com | 5bab541acd761a3093d7c4202b6e1da9 |

Figura 3- 12. Password Encriptado 2 (Moya, 2015)

- Encriptación con JAVA, por medio de una contraseña se realizara la encriptación de datos, pero tiene la particularidad que el usuario no sabe la contraseña, genera el programa a media que se va ejecutando. Dado una cadena de caracteres, se toma un carácter en un carácter, se lo convierte en código ASCII, los cuales saldrán 8

dígitos en 0 y 1, y después de manera aleatoria de los 8 números habrá dos parejas de dígitos que se las va a cambiar entre sí, es decir dos parejas estarán alteradas por lo tanto cuatro dígitos de los ocho por cada letra no se encontraran en sus puestos originales, y de esas parejas las posiciones que ocupe , si es entre el segundo con el quinto digito o el primero con el sexto, esas parejas de dígitos son las que iremos guardando en una variable string y nos entregara la llave, que es una secuencia de números donde me dirá cuáles son las posiciones de cada letra que tengo que ir cambiando para después poder desencriptarlo.

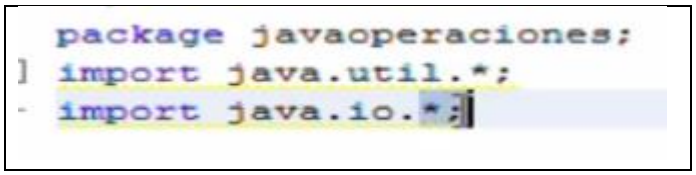



Figura 3- 13. Librería Java (Moya, 2015)

Como se puede observar en la Figura 3-13. Encriptar en JAVA, importamos la librería import java.util. *; Que nos permitirá solicitar al usuario que ingrese el texto que desee a ser encriptado.

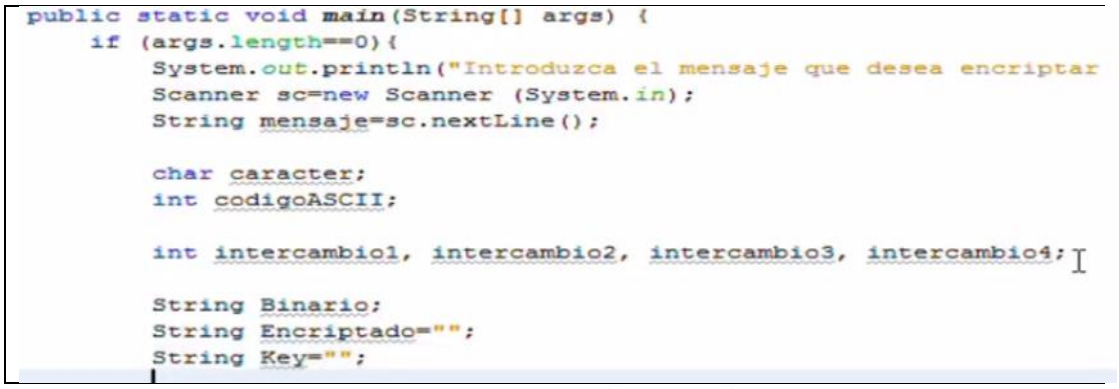



Figura 3- 14. Texto que se ingresa JAVA (Moya, 2015)

Como se puede observar en la Figura 3-14. Texto que se ingresa, creamos un arreglo que se llama leer, llamamos la función main que es leer.

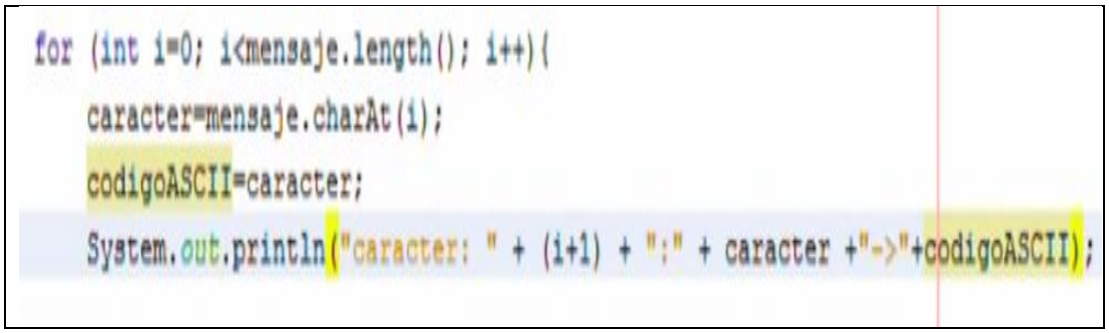



Figura 3- 15. Algoritmo JAVA (Moya, 2015)

```
Introduzca el mensaje que desea encriptar :
hola mundo
caracter: 1:h->104
caracter: 2:o->111
caracter: 3:l->108
caracter: 4:a->97
caracter: 5: ->32
caracter: 6:m->109
caracter: 7:u->117
caracter: 8:n->110
```

Figura 3- 16. Prueba JAVA (Escobar A. , 2015)

Como se puede observar en la Figura 3-16. Prueba JAVA, se ingresa la palabra hola y cada uno de sus caracteres se han transformado en código ASCII.

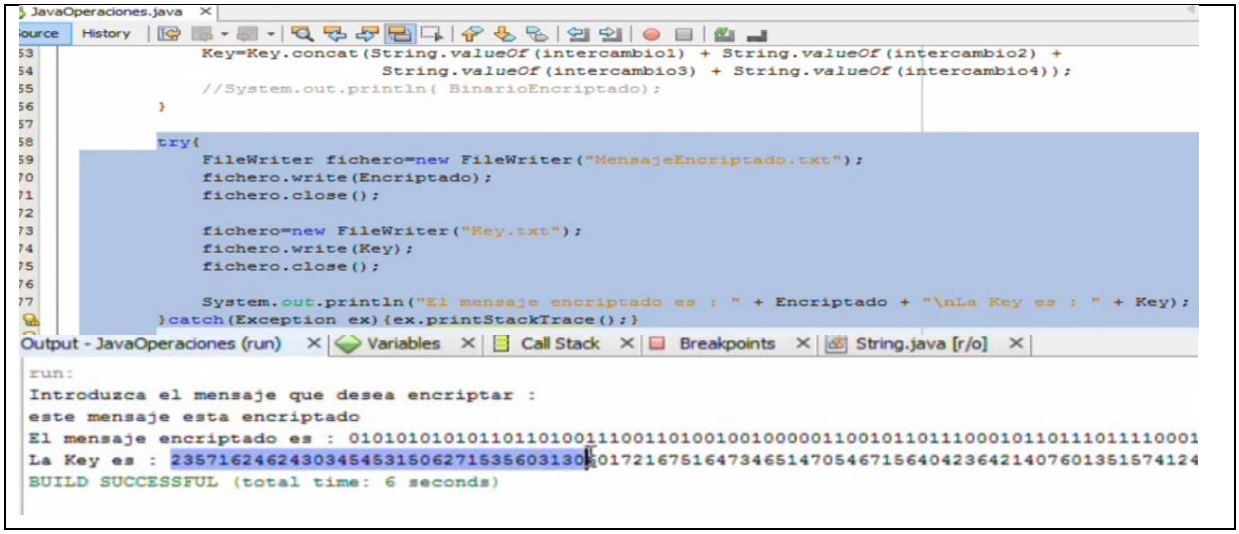



Figura 3- 17. Modelo de encriptación en un fichero (Moya, 2015)

Como se puede observar en la Figura 3-17. Modelo de encriptación en un fichero, que este método no lo podemos utilizar ya que se crea un archivo en notepad, con el archivo encriptado lo que causaría un manejo no apropiado del aplicativo a desarrollar y lleva mucho tiempo en realizar estos pasos.

- **En el análisis de requerimientos para el desarrollo del aplicativo web en ASP.NET:** encontraremos los casos de uso, los cuales representan los requerimientos que necesitamos para desarrollar este proyecto, se puede decir que aquí encontraremos la parte fundamental del desarrollo de la aplicación web paso a paso. Y tenemos los siguientes casos de usos:

**A. El usuario ingresa al aplicativo de mensajería web**

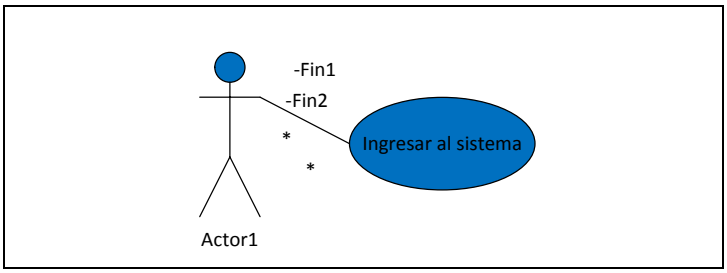



Figura 3- 18. Análisis de requerimientos (Moya, 2015)

1. Ingresar al navegador http://encriptacion.azurewebsites.net/index.aspx
2. Ingresar el usuario (en caso de no tener usuario seguir los pasos B.).
3. Colocar la contraseña.
4. Dar clic en "*Ingresar"*.

**B. El usuario debe registrarse en el aplicativo de mensajería web**

1. Presione en "*Registrarse*" en caso de no tener cuenta en el aplicativo.
2. Ingrese nombre.
3. Ingrese apellido.
4. Ingrese una contraseña.
5. Ingrese nombre de usuario o login.
6. Ingrese la ciudad.
7. Ingrese número de teléfono.
8. Ingrese una dirección de correo electrónico.
9. Se auto genera un numero primo para encriptar los mensajes.
   Presione "*Ingresar"*.
   Presione "*Cancelar*" si no desea usar nuestros servicios de encriptación.

**C. El usuario en caso de olvidarse la contraseña**

1. Dar clic en "*Recuperar Pass*".
2. Se enviara un correo con la clave provisional para que pueda ingresar.
3. Presiona la tecla "Continuar".
4. Revise su correo.
5. Abra el mensaje que le enviamos.
6. Ingrese la clave provisional que le enviamos.
7. Presione "*Ingresar"*.

### D. El usuario en caso de olvidarse el usuario

1. Dar clic en "*Recuperar Pass*".
2. Se enviara un correo con la clave provisional para que pueda ingresar.
3. Presiona la tecla "Continuar".
4. Revise su correo.
5. Abra el mensaje que le enviamos.
6. Ingrese la clave provisional que le enviamos.
7. Presione "*Ingresar".*

### E. Tiempo de duración de la clave de encriptación

1. Después que se valida su usuario y contraseña.
2. Ingresara a una pantalla en la cual le despliega su nombre y un tiempo determinado.
3. El tiempo que le despliega la pantalla es el número de sesiones que le quedan para utilizar el número primo que utilizaremos para encriptar o desencriptar los mensajes.
4. En caso de que le queden (0 días) se re-direccionara al cambio de clave y se le asignara otro número primo.

### F. Ingresar al aplicativo

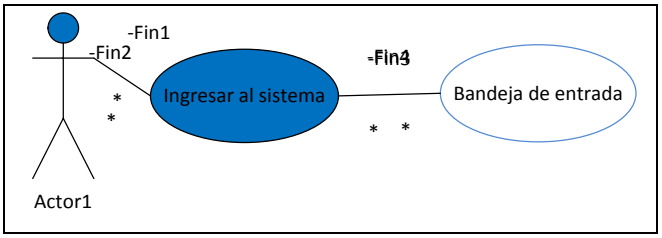



Figura 3- 19. Ingresar al aplicativo (Escobar A. , 2015)

1. Ingrese con su usuario y contraseña al aplicativo web.
2. Tiempo de duración de la contraseña y presionar "Ingresar".
3. Encontrará como presentación inicial, crear mensaje.
4. Usted puede crear el mensaje y enviarlo.
5. O puede revisar las pestañas que tenemos como "Entrada", "Salida", "Cifrado", "Cambio de clave".

## G. Lista de mensajes

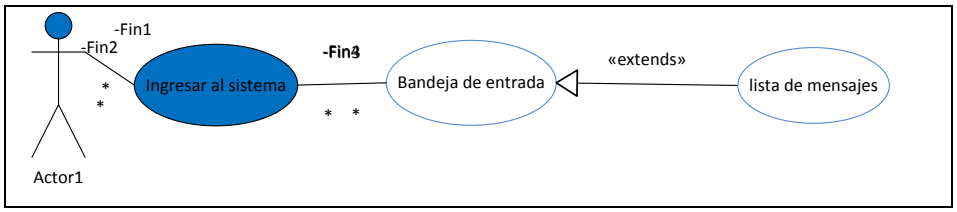



Figura 3- 20. Lista de Mensajes (Escobar A. , 2015)

1. Ingresamos con el usuario y contraseña respectivo
2. Tiempo de duración de la contraseña y presionar "Ingresar".
3. Al ingresar al aplicativo web, encontraremos la pestaña "Entrada"
4. Dicha bandeja de entrada contiene una lista de mensajes que se ha recibido en nuestra cuenta.
5. La lista de mensajes será una lista de 5 filas
6. Se encontrara en las columnas del listado el título, descripción y la fecha en la que enviaron el mensaje.

## H. Leer mensaje.

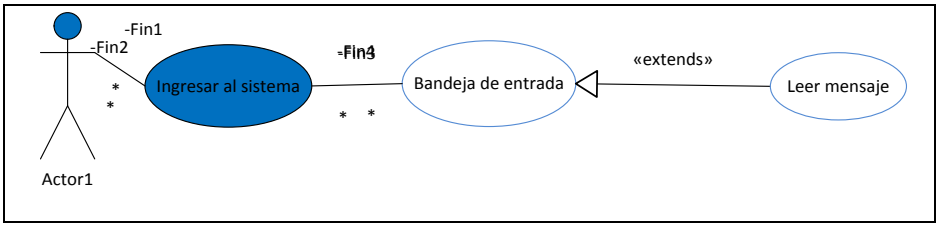



Figura 3- 21. Leer mensaje (Escobar A. , 2015)

1. En la bandeja de "Entrada", encontramos un listado de mensajes
2. Seleccionamos el que deseamos leer.
3. Aquí encontraremos, el usuario que lo envió, descripción, fecha.
4. Podemos seleccionar la opción "Responder" o "Ver Codificación"
5. Si seleccionamos "Ver Codificación", de debe colocar el numero primo junto a la palabra "Leer"
6. Si el numero primo que agrego es correcto, el mensaje se desencripta casa contrario le saldrá un mensaje "Código erróneo"

## I. Borrar mensaje

1. El usuario podrá borrar el mensaje, desde la bandeja de entrada o después de haberlo leído
2. Para borrar el mensaje lo que debe hacer es buscar la opción eliminar que se encontrará en la parte superior darle clic y borrar

3. Al igual desde la lista de mensajes podrá encontrar la opción "eliminados", selecciona el mensaje que va a ser borrado y lo elimina.

### J. Escribir un nuevo mensaje

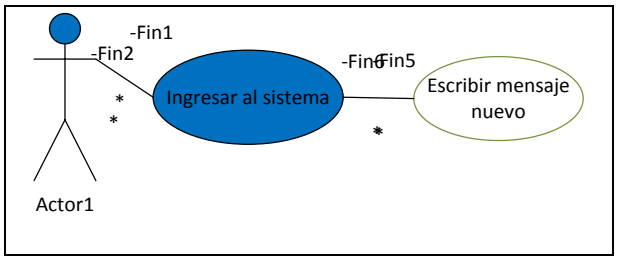



Figura 3- 22. Escribir nuevo mensaje (Escobar A. , 2015)

1. Después de haber ingresado al aplicativo web, en la bandeja de entrada a lado izquierdo encontrara la opción **"***Nuevo***"**
2. Dar clic en crear un nuevo mensaje, pero por default tenemos la opción "Nuevo"
3. Ingresaremos para quien es el mensaje, tenemos habilitado la opción buscar en el lado derecho de "Para" podrá colocar la inicial o nombre para quien va el correo en caso de no recordar puede pulsar en buscar y se desplegara los usuarios que tenemos registrado en la base de datos con esas iniciales o nombres.
4. Ingresamos el asunto del correo. Para el texto podemos colocar 1500 palabras
5. Presione "Enviar Mensaje"
6. Usted podrá observar un mensaje en la pantalla que el mensaje que guardo exitosamente.

### K. Desencriptar información

1. Selecciona la opción "Entrada".
2. Se desplegara una lista de mensajes que usted ha recibido en su cuenta.
3. Seleccione el que desea leer.
4. Encontrará un cuadro que le solicita ingresar el número primo de la persona que le envía el mensaje.
5. Si ingresa correctamente el número primo, usted podrá leer el mensaje.
6. Caso contrario le saldrá un mensaje notificando que el número primo que ingreso es erróneo.

### L. Ver mensajes enviados

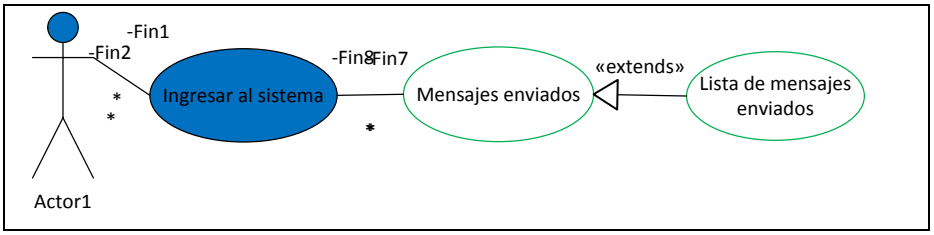



Figura 3- 23. Mensajes Enviados (Escobar A. , 2015)

1. El usuario podrá ver la lista de los mensajes que fueron enviados ( título, descripción, fecha) de su cuenta
2. El usuario podrá borrar los mensajes (seleccionando el icono vacío) y dar clic en borrar.

### M. Salir del sistema

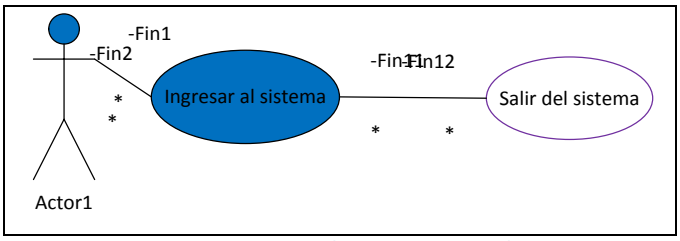



Figura 3- 24. Salir del sistema (Escobar A. , 2015)

1. El usuario encontrará en la parte superior derecha "Salir".
2. El usuario da clic en "Salir".

## 3.3 Diagrama general

En este diagrama general encontramos el total de los casos de usos o requerimientos, que se necesita para el desarrollo de nuestro aplicativo web. Se puede observar los requerimientos en forma macro.

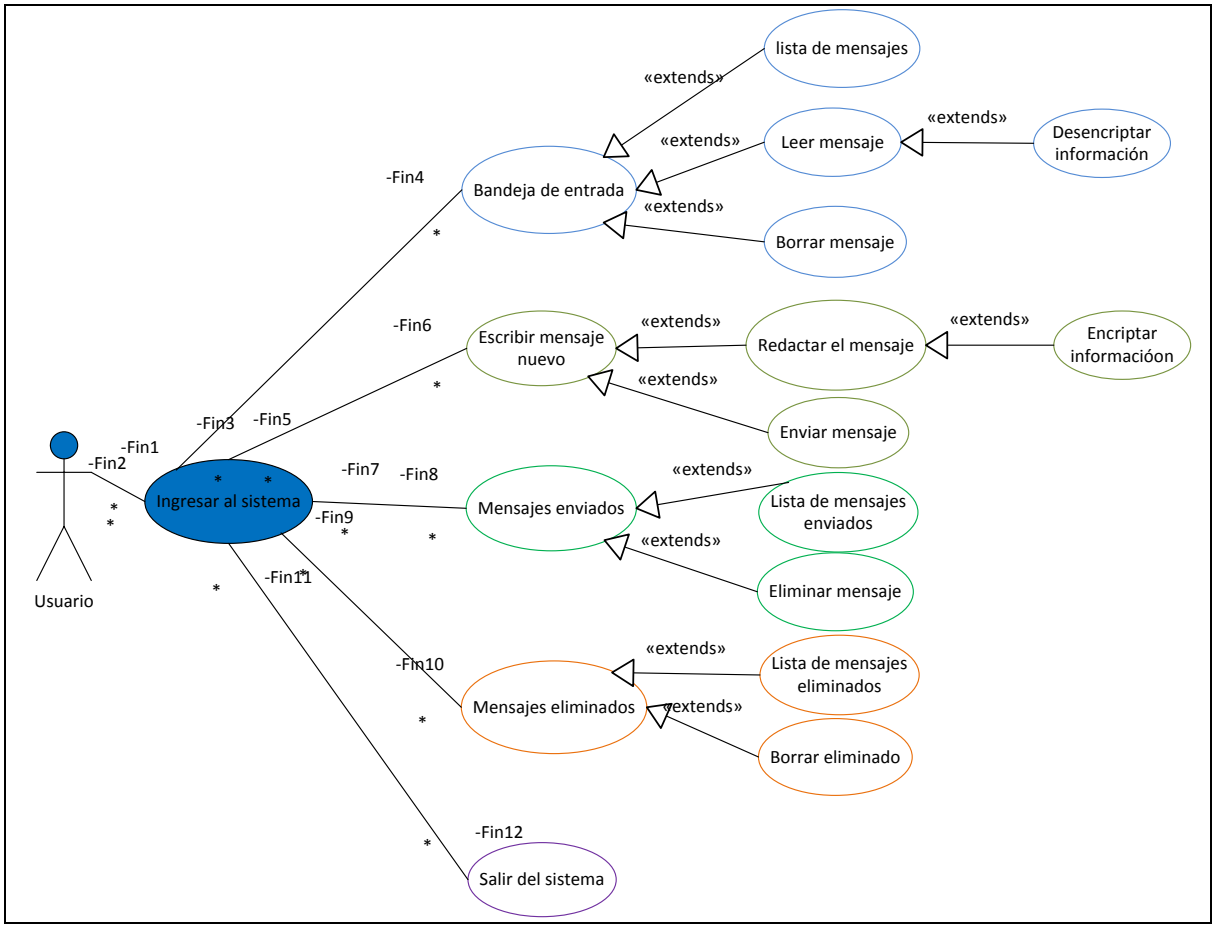



Figura 3- 25. Diagrama General (Moya, 2015)

Como podemos observar en la Figura 3-25. Diagrama General, nos permite ver a manera macro como se ira realizando el aplicativo WEB y cuáles son las funcionalidades que debe tener.

## 3.4 Diagrama a nivel conceptual del sistema

En este diagrama encontraremos la entidad y atributos de cada una de las clases que se detallan a continuación:

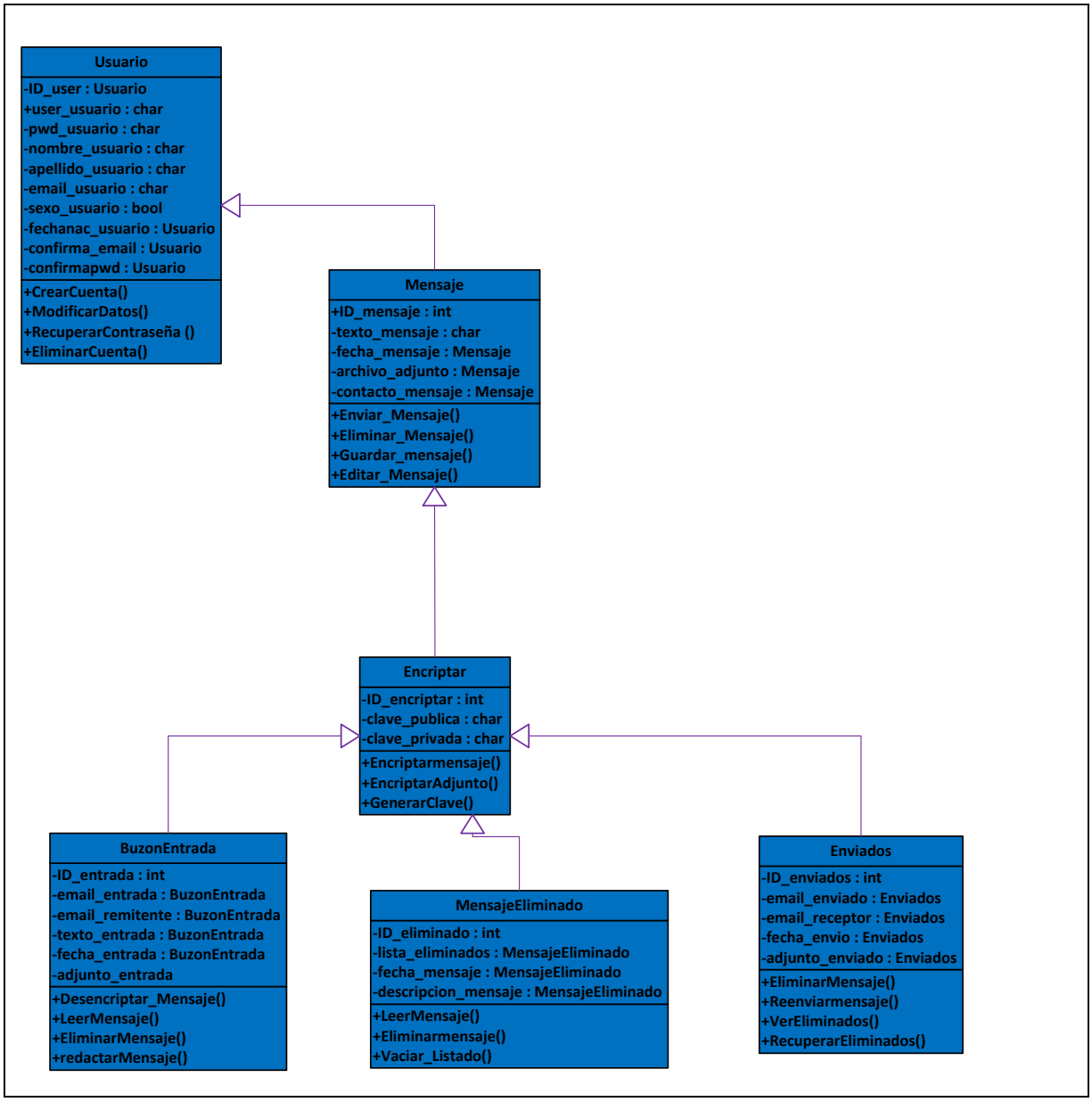



Figura 3- 26. Diagrama general conceptual (Escobar A. , 2015)

En esta Figura 3-26. Diagrama general conceptual, se puede observar las tablas que nos permiten ver de una manera amplia como esta nuestro gestor de bases y que tipo de conexiones utilizara al ser un usuario, el mensaje, la función propia que es encriptar entre otras.

Encontraremos, las tablas principales para el desarrollo del aplicativo web, dichas entidades se encuentran con sus respectivos atributos.

| Entidad | Atributos |
|---|---|
| Usuario | ID_user |
| | User_usuario |
| | pwd_usuario |
| | nombre_usuario |
| | apellido_usuario |
| | email_usuario |
| | sexo_usuario |
| | fechanac_usuario |
| | cofirma_email |
| | confirmapwd |

Tabla 7. Usuario

| Entidad | Atributos |
|---|---|
| Mensaje | ID_mensaje |
| | texto_mensaje |
| | fecha_mensaje |
| | |
| | archivo_mensaje |
| | contacto_mensaje |

Tabla 8. Mensaje

| Entidad | Atributos |
|---|---|
| Encriptar | ID_encriptar |
| | clave_publica |
| | clave_privada |

Tabla 9. Encriptar

| Entidad | Atributos |
|---|---|
| BuzonEntrada | ID_entrada |
| | email_entrada |
| | email_remitente |
| | texto_entrada |
| | fecha_entrada |
| | adjunto_entrada |

Tabla 10. Buzon de entrada

| Entidad | Atributos |
|---|---|
| MensajeEiminado | ID_eliminado |
| | lista_eliminado |
| | fecha_mensaje |
| | descripcion_mensaje |

Tabla 11. Mensaje eliminado

| Entidad | Atributos |
|---|---|
| Enviados | ID_enviados |
| | email_enviado |
| | email_receptor |
| | fecha_envio |
| | Adjunto_enviado |

Tabla 12. Enviados

Figura 3- 27. Entidad Atributo (Escobar A. , 2015)

Se puede observar en la Figura 3-27. Entidad Atributo, Se define a una entidad relación, como una herramienta de modelado de datos que permite presentar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades por lo cual, cada uno de los atributos que se utiliza en este grafico se refiere a las diferentes entidades que tenemos en el proyecto, lo que nos permitirá manejar cada uno de ellos en la programación y en el gestor de datos, declarando sus tamaños y respectivas funciones.

# 4. CAPITULO IV – CONSTRUCCIÓN DE LA APLICACIÓN

La construcción de aplicación se la ira mostrando paso a paso, como se realizó, que cambios se hizo, explicando cómo se realizó la programación, cuáles fueron los estándares en la programación y en diseño, la documentación que se pudo recopilar y manuales que se fueron realizando en la iteración de pruebas, que sintaxis y librerías utilizamos entre otras cosas.

## 4.1 Lenguaje de programación

Para realizar el aplicativo de encriptación utilizaremos ASP.NET, ya que como se ha indicado en el capítulo anterior , no se puede utilizar ninguno de los implementados , porque no permiten realizar lo que deseamos que es cuando el usuario envía un mensaje, dicho mensaje se encripte y llegue al receptor y solo lo pueda leer en caso de que él tenga la clave, caso contrario solo vera texto cifrado.

Lo que en resumen decimos es que crearemos un algoritmo que nos permita realizar esta tarea, y como ASP.NET es una herramienta muy amigable y ya con todos los conceptos claros será más sencillo.

## 4.2 Programación

Función para encriptación, primero veo la longitud que se ingresa para el mensaje, entonces con nuestro "for" sabemos que empieza en 1 hasta el tamaño de la cadena, sabemos que es una letra y se transforma en su código ASCII, luego le multiplico por el numero primo que se genera cuando se ingresa al aplicativo y ese valor lo separo con un ("/") y este valor agrego a mi cadena.

```
0 references
Public Function encriptacion(ByVal cadena As String, ByVal numero As Integer) As String

    Dim longitud, cont, valor As Integer
    Dim res As String

    longitud = Len(cadena)
    res = ""

    For cont = 1 To longitud
        valor = Asc(Mid(cadena, cont, 1))
        valor = valor * numero
        res += CStr(valor)
        res += "/"
    Next


    Return res

End Function
```

Figura 4- 1. Encriptación (Moya, 2015)

Función para desencriptación, primero veo la longitud que se ingresa para el mensaje, entonces con nuestro "for" sabemos que empieza en 1 hasta el tamaño de la cadena, sabemos que es una código ASCII, busca ("/") esta barra de división de caracteres, luego divide para el numero primo que en registrado al inicio de la aplicación, recibe el fragmento lo transforma y vuelve a ver el texto plano.

```
0 references
Public Function desencriptacion(ByVal cadena As String, ByVal numero As Integer) As String

    Dim longitud, cont, valor As Integer
    Dim res, tope, fragmento As String

    longitud = Len(cadena)
    res = ""
    fragmento = ""

    For cont = 1 To longitud
        tope = Mid(cadena, cont, 1)


        If tope = "/" Then
            valor = fragmento
            valor = valor / numero
            fragmento = ""
            res += Chr(valor)
        Else
            fragmento += tope
        End If


    Next

    Return res

End Function
```

Figura 4- 2. Desencriptación (Moya, 2015)

Función de validación, en esta validación nos conectamos con nuestro gestor de datos, en donde el usuario ingresa su usuario y contraseña, si el usuario está registrado e ingresa correctamente su usuario y contraseña podrá acceder al aplicativo, caso contrario le retornara valores como el usuario no es válido, usuario no registrado, contraseña incorrecta. Se necesita de manera obligatoria que nuestro gestor de bases de datos se encuentre operativo, ya que sin eso no podemos acceder al aplicativo, el mismo que no podrá validar si los usuarios y contraseñas que están ingresando son correctas o incorrectas.

```
0 references
Public Function validacion(ByVal login As String, ByVal pass As String) As String
    Try
        coneccion()
        _sql = "Select count(1) as existe from TblUsuario where usu_login='" + login + "'"
        oregi = ocone.Execute(_sql)

        If Not oregi.Fields("existe").Value = "0" Then
            _sql = "Select * from TblUsuario where usu_pass='" + pass + "' and usu_login='" + login + "'"
            oregi = ocone.Execute(_sql)
            If Not oregi.RecordCount = "0" Then
                cod_per = oregi.Fields(0).Value
                Return ("Usuario Valido")
            Else
                Return ("Contraseña incorrecta")
            End If
        Else
            Return ("Usuario no registrado")
        End If
        ocone.Close()

    Catch ex As Exception
        Return "mensaje: " + ex.Message
    End Try

End Function
```

Figura 4- 3. Función de Validación (Escobar A. , 2015)

Para la Función numero primo creamos un Random entre 1001 y 9999, teniendo en claro que un número primo es un natural mayor que 1 que tiene únicamente dos divisores distintos: el mismo y el 1 ya que con este número primo aplicaremos nuestro cifrado de datos de manera asimétrica utilizando una clave pública y una privada. Se realiza mediante banderas comprobando si su divisor es para sí mismo o para 1.

```
    0 references
    Public Function numero_primo() As String

        Dim res As String = ""
        Dim modulo, cont, bandera, bandera2, numero As Integer


        bandera2 = 1

        While bandera2 = 1

            Dim Random As New Random()
            numero = Random.Next(1001, 9999)
            bandera = 1
            cont = 2

            While (cont < numero And bandera = 1)
                modulo = numero Mod cont
                If modulo = 0 Then
                    bandera = 2
                End If
                cont += 1
            End While

            If bandera = 1 Then
                bandera2 = 2
            End If

        End While

        res = CStr(numero)
```

Figura 4- 4. Numero primo (Moya, 2015)

### 4.1.1 Estándares

Para los estándares de la programación se tomó en cuenta el nombre de las funciones, la declaración de las variables.

- **Funciones para el algoritmo encriptación.**
  - Nombre de la función en minúsculas
  - Variables en minúsculas

- **Funciones para el algoritmo desencriptación**.
  - Nombre de la función en minúsculas
  - Variables en minúsculas

- **Estándares de diseño en el desarrollo web**
  - Para el inicio de la página se utilizara un fondo verde
  - Al lado izquierdo adjuntaremos el logo de nuestro proyecto JAMENCRYPT
  - Para las letras de nuestro proyecto se utilizara (heading 1, 2, 3) según corresponda.
  - Para mensajes de error se utilizara colores fuertes, para que se vea el mensaje
  - Cuando se encuentre dentro del aplicativo, se tendrá una página master, que será utilizada para todas las funciones internas. Con lo cual mantendremos el diseño del aplicativo web.
- **Documentación , comentarios y mensajes**
  - La parte fundamental del proyecto se encontrara en CLsUtilitarios.vb, se puede documentar el algoritmo, las funciones y conexiones a la base de datos.
  - En el caso de mensajes como se indica en la parte de estándares de diseño, el mensaje será presentado con colores fuertes para que llamen la atención del usuario y corrija su error.
  - En esta ocasión no se utilizara comentarios, solo en caso de ser necesarios en la presentación final.
- **Sintaxis**
  - Para mantener la equidad en el aplicativo se utilizara minúsculas para las funciones y variables, en el caso de los métodos o conversiones se utilizara lo la primera letra mayúscula.
  - En el caso de mensajes se utilizara el método título.
  - Para el aplicativo Web, se utilizara letras modo título.
- **Librerías y versiones**
  - ASP.NET 4.5.1

## 4.3 Pruebas de la aplicación

- Pantalla de ingreso

En la Figura 4-5. Pantalla de ingreso se puede observar el diseño y la primera prueba que se realiza que es el ingreso al aplicativo desde la dirección URL de internet, http://encriptacion.azurewebsites.net/index.aspx.


Figura 4- 5. Pantalla de Ingreso (Johanna Moya A. E., 2015)

Se muestra el error, en caso de ingresar mal el usuario o contraseña el usuario podrá observar un mensaje de error el cual le indica que el usuario no se encuentra registrado o el otro caso es que le diga contraseña incorrecta


Figura 4- 6. Usuario no registrado (Johanna Moya A. E., 2015)

Como se puede observar en la Figura 4-6. Usuario no registrado otra de las probabilidades que no se pueda acceder es que el gestor de base de datos no se encuentre operativo, y por esta razón no se conecte al aplicativo y no se pueda ingresar.

- Cuando se valida con el gestor de base de datos, se ingresa a la pantalla siguiente.

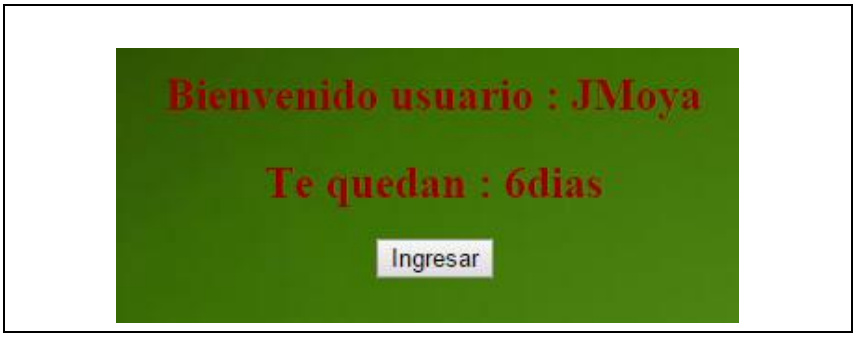



Figura 4- 7. Ingresando al aplicativo (Johanna Moya A. E., 2015)

Se puede observar en la Figura 4-7. Ingresando al aplicativo, que cuando se valida con el gestor de datos, ingresamos al aplicativo y nos muestra el nombre del usuario y el tiempo que me queda para realizar el cambio de la contraseña y obtener otro número primo, que este es cada 30 veces que ingrese al aplicativo me pedirá cambiar la clave, caso contrario no podrá utilizar el aplicativo para enviar mensajes.

- Se ingresa al aplicativo después de aceptar y verificar cuantas sesiones todavía nos quedan para utilizar dicho número primo para encriptar.

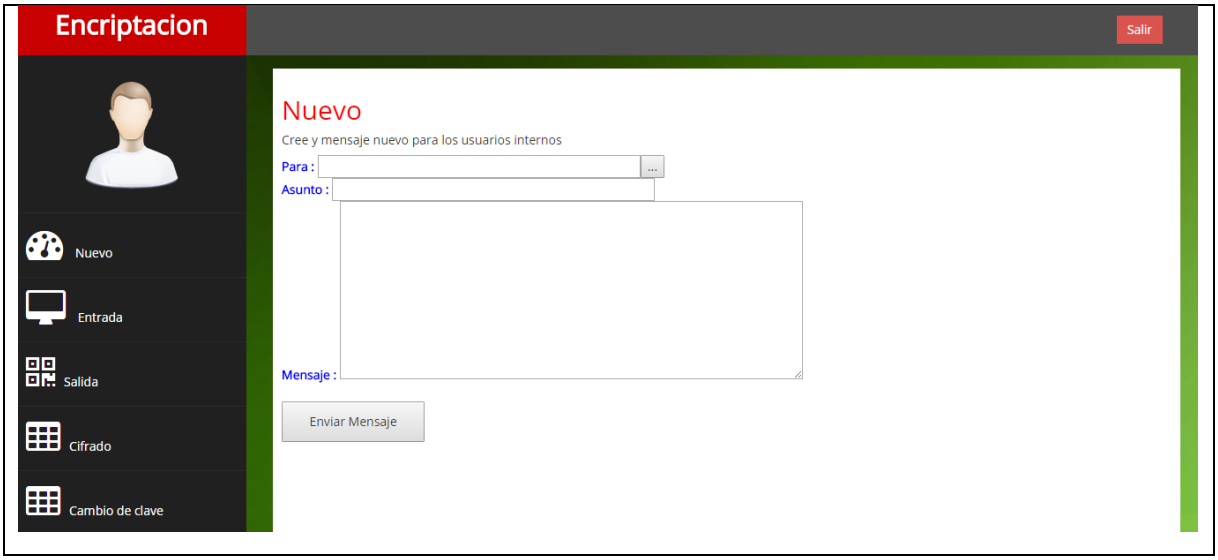



Figura 4- 8. Aplicativo de mensajería (Johanna Moya A. E., 2015)

Para la Figura 4-8. Aplicativo de mensajería, en la verificación o prueba de esta página, primero revisamos la web master funcione para todas las funciones, se puede observar también que al iniciar la sesión y pasar dos filtros la primera funcionalidad es escribir el mensaje.

- Para escribir un mensaje, se coloca la inicial del nombre y se da clic en buscar. Observaremos que se desplegara los usuarios que se tiene en la base de datos.
- En caso de colocar un usuario, que no se encuentra en la base de datos, saldrá el siguiente error.
- Después de ingresar el usuario al que va a ser enviado el mensaje, se coloca el asunto y cuerpo del correo.

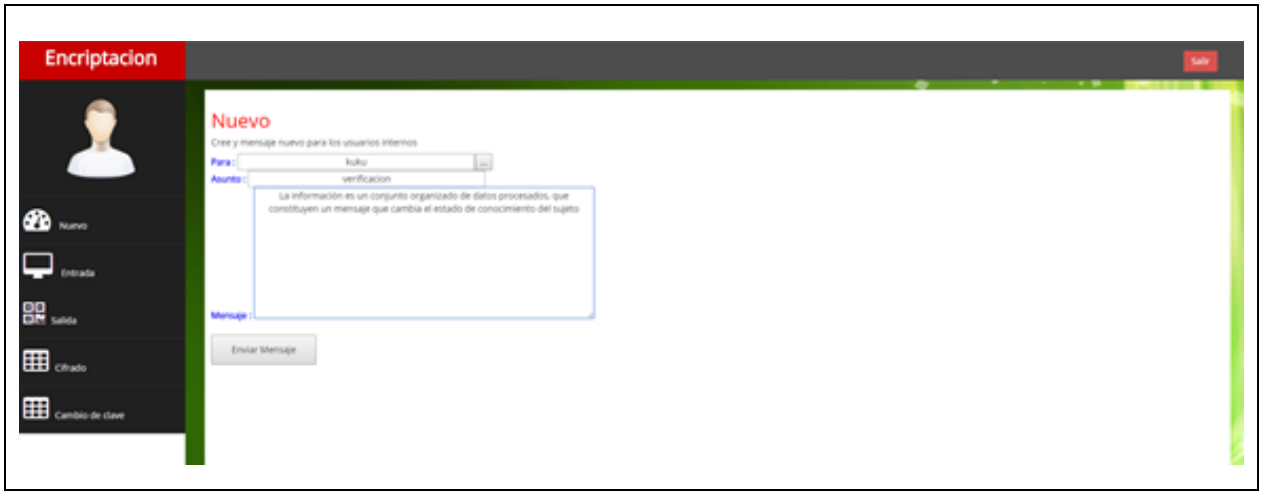



Figura 4- 9. Enviar mensaje (Johanna Moya A. E., 2015)

- Se da clic en enviar mensaje, y se observara un mensaje el cual nos indica que el mensaje se ha guardado con éxito.
- Para el caso de bandeja de entrada, se encontrara una lista de correo que han sido enviados por otros usuarios a nuestra cuenta.


Figura 4- 10. Bandeja de entrada (Johanna Moya A. E., 2015)

- Seleccionar, el mensaje que se desea leer y se desplegara la información de quien nos envía el mensaje, el asunto, la fecha y el mensaje encriptado.
- Para poder leer el mensaje, se da clic en ver codificación y se activara el botón leer.
- En este cuadro que se habilita, se debe ingresar el numero primo de la persona que va a ser enviado el mensaje, en caso de ser incorrecto se desplegara el mensaje notificando que es "Código erróneo"

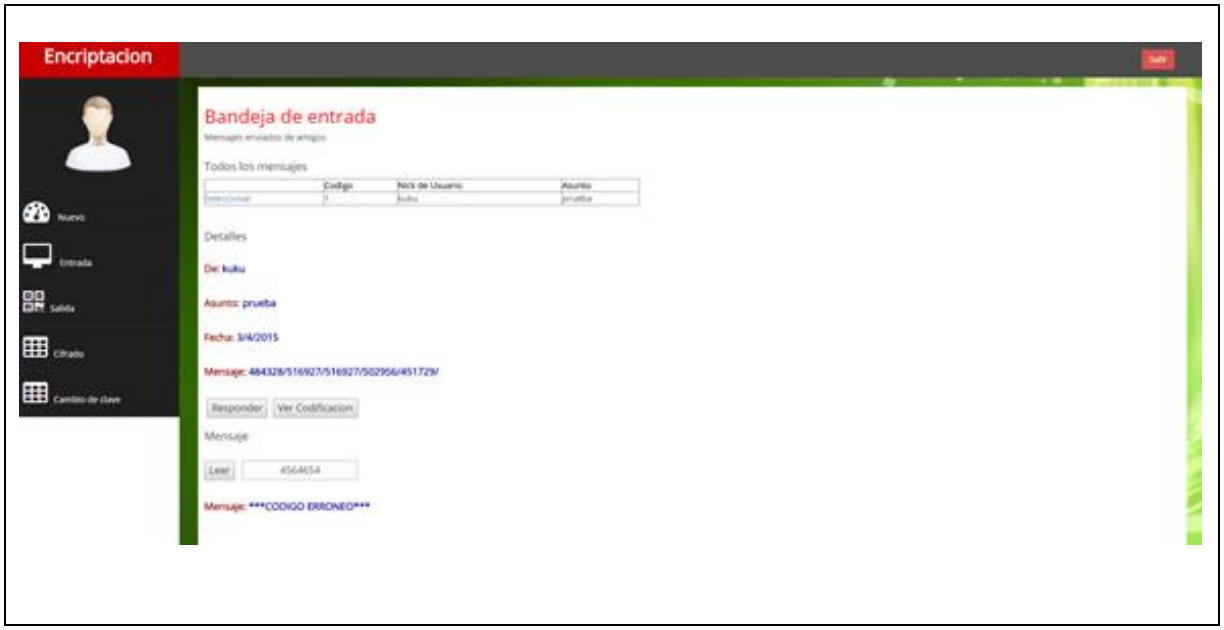



Figura 4- 11. Bandeja de entrada del receptor (Johanna Moya A. E., 2015)

Como se puede observar en la Figura 4-11. Bandeja de entrada del receptor, se ha visualizado que todas las pruebas anteriores ya han sido pasada con éxito, por lo cual pasamos a realizar las pruebas del usuario receptor, en donde se verificara que el mensaje está siendo enviado y que para poder desencriptar se necesita el numero primo, caso contrario será como estar viendo un texto en bloque.

- Para ver los mensajes de salida, se tendrá una lista de los mensajes que han sido enviados para diferentes usuarios
- Además se tendrá en la pestaña de cifrado un ejemplo de encriptación
- Para el cambio de clave, se debe dar clic en generar clave, después se ingresa la clave antigua y se coloca la clave nueva, se auto genera un numero primo que se debe entregar a los usuarios cuando le manden un mensaje.
- Cuando se cambia la clave, se genera el siguiente mensaje "Se ha guardado con éxito"

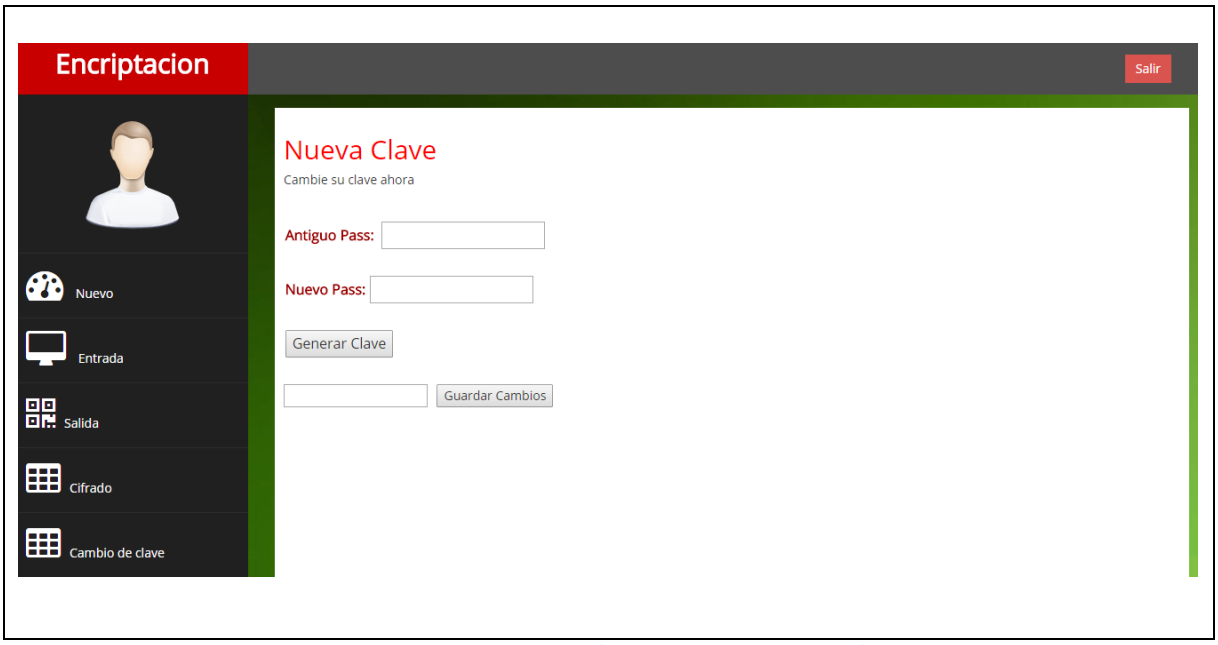



Figura 4- 12. Cambio de clave (Johanna Moya A. E., 2015)

- Se debe salir para verificar el cambio de clave y se debe ingresar con la nueva clave.


Figura 4- 13. Cambio de clave inicio (Johanna Moya A. E., 2015)

Como se puede observar en la Figura 4-12. Cambio de clave, es la última prueba que se puede aplicar a la página web dando todas las funcionalidades respuestas satisfactorias, las mismas que fueron aplicadas tanto al emisor como al receptor con todas sus funcionalidades podemos decir que el aplicativo lo podemos poner en marcha y que finalmente fue concebida con ASP.NET y gestor de datos SQL SERVER 2008.

## 4.4 Implementación de la solución

Toda la implementación fue realizada a base del análisis que se realizó al principio de este desarrollo de la disertación de grado, aplicando la correcta metodología ya que dicha metodología nos permitió realizar cambio los cuales no fueron muy complicados porque se hacía una revisión a cada final de iteración, y es así como pudimos cambiar de lenguajes, incluso de gestores de base de datos, con las técnicas ya mencionadas en el capítulo 1, los algoritmos de encriptación claros y siguiendo el procedimiento correcto para encriptar se puede decir que se pudo tener este aplicativo, que se encuentra 100% funcional.

Podemos acceder vía WEB a **http://encriptacion.azurewebsites.net/index.aspx,** o desde nuestra computadora madre en donde realizamos el desarrollo.

## 4.5 Documentación técnica del caso de estudio

La documentación técnica que tenemos en el desarrollo de este aplicativo, es el documento impreso donde se demuestra que realizamos el respectivo análisis, utilizamos una metodología correcta y a la hora de implementar utilizamos varios lenguajes de pruebas hasta obtener el resultado esperado.

Además cabe recalcar que en los anexos se adjuntara videos, entrevistas, libros,

links de páginas de internet, códigos encontrados para la implementación, revistas, presentaciones en power point, tutoriales.

### 4.5.1 Manual de usuario

El manual de usuario, consta del uso del aplicativo paso a paso, encontrara desde como registrarse en la aplicación, que sucede cuando no está registrada, que debe hacer cuando está dentro de la aplicación, que requerimientos básicos necesita para poder manejar esta aplicación, cuanto tiempo tiene disponible para usar, además encontrara teléfonos de soporte que le podremos estar ayudando las 24 horas del día los 365 días del año. Para el conocimiento del manejo del cifrado usted encontrara una pestaña que se llama cifrado a la cual usted después de darle clic, le dará un pequeño ejemplo de como la información se encripta.

### 4.5.2 Manual técnico

El manual técnico encontraremos, casos de uso, diagramas, diseño, diseño de la base de datos, que se refiere cada tabla como están conectadas cada una de ellas, que lenguaje de programación se utilizó, el nombre del gestor de base de datos, entidades relación. Explicación como el aplicativo se conecta a la base de datos, que servidor se contrató para que dicho aplicativo funcione en la web, cuanto tiempo estará disponible entre otras cosas.

# 5. CAPÍTULO V - CONCLUSIONES Y RECOMENDACIONES

En este capítulo final de nuestra disertación de grado se encontrara conclusiones y recomendaciones, que se pudo investigar durante todo el desarrollo y aplicación del aplicativo, y de nuestra vida institucional, ya que sin los conocimientos aprendidos durante este periodo universitario sería imposible haberlo logrado, incluiremos conclusiones técnicas, personales, de la carrera.

## 5.1 Conclusiones

1. La formación académica recibida en la PUCE, ha permitido cumplir con el desarrollo del proyecto de disertación de grado, que comprende la investigación del tema, el análisis de requerimientos, el diseño lógico y físico y por último el desarrollo de la aplicación.
   .
2. Gracias al avance tecnológico a nivel mundial, se cuenta con herramientas que permite desarrollar aplicativos de una manera ágil y confiable, como en este caso el encriptamiento de información.
3. Existen una variedad grande de funciones de encriptamiento, que se debe analizar y realizar las respectivas pruebas para la utilización individual o combinada según los requerimientos.
4. La selecciona adecuada de la metodología y las herramientas, permitió que el desarrollo del aplicativo maneje un código privado para que se pueda mantener la privacidad emisor-receptor, asegurando la confidencialidad evitando ser intersectada por terceros.
5. La metodología scrum, ayudó mucho ya que el trabajo en equipo funciona bastante bien, y se trabaja por igual, de la misma manera al realizar cambios siempre estamos al tanto de todo al final de una iteración y en caso de cambiar alguna parte fundamental se recibe como retroalimentación y además se realiza el cambio.
6. En la creación de cualquier aplicativo, es importante cumplir con el ciclo de desarrollo de software, al margen de la metodología que se aplique, hasta las pruebas integrales finales son muy importantes.
7. La base conceptual es importante para comprender el funcionamiento de los algoritmos, sin embargo es necesario realizar pruebas prácticas para validar y evaluar los resultados e ir realizando los ajustes indispensables.

## 5.2 Recomendaciones

- Para la creación de una nueva versión de nuestro aplicativo el número primo o código que se genera en el aplicativo para encriptar la información, podría ser desarrollado o profundizado de tal manera que no tenga un tiempo de caducidad, sino que se autogenere en cada mensaje.
- Se sugiere para la base de datos, utilizar programación para encriptar los datos de registro, así el gestor de base de datos aunque sea manejada por un técnico, no sabría lo que contiene dicha base de datos y se mantendría la confidencialidad.
- Se propone para el aplicativo desarrollado para la versión 2, insertar archivos para cuando se envía el mensaje.

# BIBLIOGRAFÍA

## A. Libros, Sitios Web, Artículos y folletos

Adiccion, D. (7 de octubre de 2012). *Seguridad Informatica*. Recuperado el junio de 2014, de http://seguridadinformaticaequipo7.blogspot.com/

Aguilera, P. (s.f.). Seguridad Informatica.

Alegsa. (2008). *Diccionario Web*. Recuperado el 30 de Agosto de 2014, de http://www.alegsa.com.ar/Dic/encriptacion.php

Assange. (2012). *Organización Autónoma sin Fines de Lucro "TV-Novosti"*. Recuperado el julio de 2014, de http://assange.rt.com/es/episodio-8--assange-y-los-criptopunks/full-translation-text/#page-1

Barbero, M. J. (2007). *manual de supervivencia del administrador de MySQL.*

Blogspot. (2009). *Criptologia y detalles*. Recuperado el 25 de Agosto de 2014, de http://encripdedatos.blogspot.com/

Bosselae. (2009). *Funciones Criptográficas de Hash*. Recuperado el 2 de julio de 2014, de http://www.segu-info.com.ar/proyectos/p1_hash.html

Cepheus. (21 de Octubre de 2006). *File usage on Commons*. Recuperado el mayo de 2014, de http://commons.wikimedia.org/wiki/File:Caesar3.svg

*Departamento de electrónica*. (marzo de 2012). Recuperado el junio de 2014, de http://profesores.elo.utfsm.cl/~agv/elo323/2s06/projects/NaborMoral/web3.htm

Echeverrria, G. (4, 37, 125,128,129,179,144). procedimientos y medidas de seguridad informatica conceptos básicos de seguridad de redes.

EPIC, C. d. (2010). *International Survey of Encryptio Policy*. Recuperado el 5 de Septiembre de 2014, de http://www2.epic.org/reports/crypto2000/

Escobar, A. (Abril de 2015). Quito, Pichincha, Ecuador: PUCE.

Escobar, A. (Abril de 2015). Quito, Pichincha, Ecuador: PUCE.

Escobar, A. (2015). Scrum. *Desarrollo de una aplicacion para encriptar informacion en la transmision de datos en un aplicativo web*. Quito, Pichincha, Ecuador: PUCE.

facebook. (2013). *facebook*. Recuperado el junio de 2014, de https://www.facebook.com/help/112061095610075

Fercufer. (20 de diciembre de 2011). *File usage on Commons*. Recuperado el junio de 2014, de http://commons.wikimedia.org/wiki/File:Hash_function2-es.svg

Fisher, R. P. (s.f.). *Seguridad en los sistemas informaticos.* Madrid: LAVEL, los llanos,nave 6, Humanos.

García, F. E. (s.f.). *Proyectos salón Hogar*. Recuperado el mayo de 2014, de http://www.proyectosalonhogar.com/Tatuaje/Historia.htm

Garibaldi, G. (s.f.). Comercio electronico: conceptos y reflexiones básicas. Buenos Aires, Argentina: BID-INTAL.

Gutiérrez, p. (enero de 2013). *GENBETA*. Recuperado el junio de 2014, de http://www.genbetadev.com/seguridad-informatica/manual-de-gpg-cifra-y-envia-datos-de-forma-segura

*Hardcode xploit*. (mayo de 2012). Recuperado el julio de 2014, de http://hardc0dexploit.blogspot.com/2014/06/criptografia-simetrica-asimetrica-e.html

Hermes, t. (2011). *Firma Digital*. Recuperado el julio de 2014, de http://www.firma-digital.cr/como%20funciona/

Ianturton. (6 de febrero de 2014). *Diario Turing*. Recuperado el mayo de 2014, de http://www.eldiario.es/turing/criptografia/alan-turing-enigma-codigo_0_226078042.html

Jessy. (8 de agosto de 2012). *Firma electrónica*. Recuperado el julio de 2014, de http://firmasdigielec.blogspot.com/2012/08/firma-electronica.html

Johanna Moya, A. E. (Abril de 2015). DESARROLLO DE UNA APLICACIÓN PARA ENCRIPTAR INFORMACIÓN EN LA TRANSMISIÓN DE DATOS EN UN APLICATIVO DE MENSAJERIA WEB. Quito, Pichincha, Ecuador: PUCE.

Johanna Moya, P. (julio de 2014). Codigo ASCII. *Codigo ASCII*. Quito, Pichincha, Ecuador: PUCE, Disertacion de grado Desarrollo de una aplicaion para encriptar informacion en la transmision de datos en un aplicativo web.

Kuleuven. (2012). *Ripemd*. Recuperado el 22 de agosto de 2014, de www.esat.kuleuven.ac.be/~bosselae/ripemd160.html

Laudon, K. C. (2004). Sistemas de información gerencial. En J. P. laudon. Mexico: Pearson Educacion.

López, J. P. (1998). Criptografía digital, fundamentos y aplicaciones. Zaragoza: Prensas Universitarias.

Luringen. (20 de Febrero de 2007). *Wikimedia Commons*. Recuperado el mayo de 2014, de http://commons.wikimedia.org/wiki/File:Skytale.png

México, U. n. (2012). *Redes y seguridad*. Recuperado el junio de 2014, de http://redyseguridad.fi-p.unam.mx/proyectos/criptografia/criptografia/index.php/2-tecnicas-clasicas-de-cifrado/24-formas-de-procesamiento-de-datos/243-cifrador-por-bloques

México, U. N. (2012). *Redes y Seguridad*. Recuperado el julio de 2014, de http://redyseguridad.fi-p.unam.mx/proyectos/criptografia/criptografia/index.php/2-tecnicas-clasicas-de-cifrado/23-numero-de-claves/233-sistemas-de-dos-claves-cifradores-asimetricos

Moya, J. (julio de 2014). *Desarrollo de una aplicacion para encriptar informacion en la transmision de datos en un aplicativo web*. Quito, Pichincha, Ecuador: PUCE.

Moya, J. (abril de 2015). Quito, Pichincha, Ecuador: PUCE.

Moya, J. (Abril de 2015). Quito, Pichincha, Ecuador: PUCE.

Moya, J. (Abril de 2015). Quito, Pichincha, Ecuador: PUCE.

Moya, J. (Abril de 2015). Ciclo de desarrollo Scrum. *Desarrollo de una aplicacion para encriptar informacion en la transmision de datos en un aplicativo web*. Quito, Pichincha, Ecuador: PUCE.

Moya, J. (Abril de 2015). DESARROLLO DE UNA APLICACIÓN PARA ENCRIPTAR INFORMACIÓN EN LA TRANSMISIÓN DE DATOS EN UN APLICATIVO DE MENSAJERIA WEB. Quito, Pichincha, Ecuador: PUCE.

Moya, J. (Abril de 2015). DESARROLLO DE UNA APLICACIÓN PARA ENCRIPTAR INFORMACIÓN EN LA TRANSMISIÓN DE DATOS EN UN APLICATIVO DE MENSAJERIA WEB. Quito, Pichincha, Ecuador: PUCE.

Moya, J. (2015). ECB Cifrado. *Desarrollo de una aplicacion para encriptar informacion en la transmision de datos en un aplicativo web*. Quito, Pichincha, Ecuador: PUCE.

Moya, J. (abril de 2015). Factorial. *Desarrollo de una aplicacion para encriptar informacion en la transmision de datos en un aplicativo web*. Quito, Pichincha, Ecuador: PUCE.

Moya, J. (Abril de 2015). HTML. *Desarrollo de una aplicacion para encriptar informacion en la transmision de datos en un aplicativo web*. Quito, Pichincha, Ecuador: PUCE.

Moya, J. (2015). PHP. *Desarrollo de una aplicacion para encriptar informacion en la transmision de datos en un aplicativo web*. Quito, Pichincha, Ecuador: PUCE.

Moya, J. (abril de 2015). SCRUM. *Desarrollo de una aplicacion para encriptar informacion en la*

*transmision de datos en un aplicativo web*. Quito, Pichincha, Ecuador: PUCE.

Oxford. (2002). Diccionario de internet. España: Complutense S.A.

Privaterra, O. (2011). *Encriptación: Preguntas y Respuestas*. Recuperado el 2 de Septiembre de 2014, de http://www.frontlinedefenders.org/book/export/html/4584

Public, M. (Julio de 3 de 2013). *The Current*. Recuperado el mayo de 2014, de http://www.thecurrent.org/feature/2013/07/03/a-to-z-weekend

Softeng. (julio de 2010). *The internet development company*. Recuperado el julio de 2014, de http://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum/proceso-roles-de-scrum.html

Solano, D. (12 de Octubre de 2012). *Seguridad informatica*. Recuperado el junio de 2014, de http://seguridadinformatica3c.blogspot.com/2012/10/encriptacion.html

Telefonica. (2012). El debate sobre la privacidad y seguridad en la red: regulación y mercados. madris (España): Ariel.

UNAD, U. N. (s.f.). *Datateca*. Recuperado el junio de 2014, de http://datateca.unad.edu.co/contenidos/233011/233011Exe/leccin_1_criptografa_de_clave_simtrica.html

Valderrama, J. O. (2002). Información Tecnologica. En c. J. Rojas. Chile: La Serena.

Zerboydakis, A. (2014). *Informarica de hoy*. Recuperado el 13 de Septiembre de 2014, de httpwww.informatica-hoy.com.ar/seguridad-informatica/Seguridad-Nuestros-datos-encriptados.php

# GLOSARIO

## A

Autenticación
es el proceso de detectar y comprobar la identidad de una entidad de seguridad examinando las credenciales del usuario y validando esas credenciales contra alguna autoridad 15, 20, 31, 32, 33, 34

## B

BLOB
es un objeto binario que se puede tratar una cantidad de datos variables. 50

## Ch

Checksum
Se basa en la suma de chequeo de internet
se suman todas las palabras de 16 bits que conforman el mensaje y se transmite, junto con el mensaj, el resultado de la suma es el algoritmo 10

## C

Confidencial
solo a individuos autorizados se les permite un acceso logico y una custodia fisica. 1
Confidencialidad
Se entiende como la protección de datos y de información intercambiada entre un emisor y uno o más destinatarios frente a terceros 1, 7, 15, 19, 20, 33, 76, 77
Criptograma
Es un fragmento de mensaje cifrado 4, 38
Criptopunks
Los criptopunks son los activistas que entienden que por medio de la criptografía, es decir, por medio de un lenguaje informático cifrado de uso ciudadano y abierto –y no solamente para fines militares o de guerra- es posible conseguir un cambio social y político radical en la sociedad 2

## D

DNS
Es una base de datos distribuida uasada por aplicaciones TCP/IP para mapear entre nombres de hosts(que vienen dados por una cadena ascii) y direcciones IP(en forma binaria), tambien provee a los correos electronicos información de ruteo 19

## E

Esteganografía
Del grupo steganos(oculto) y graphos(escritura) se puede definir como la ocultación de información en un canal encubierto con el prpósito de prevenir la deteccion de un mensaje oculto VIII, 6

## I



## J



## R



## S



## T

## W

WWW

Sigla de la expresion inglesa World Wide Web, red informatica mundial, sistema lógico de acceso y búsqueda de la información disponible en Internet, cuyas unidades informativas son las páginas web. 19

## X

XOR

El operador lógico Disyunción exclusiva tambien llamado o exclusivo, simbolizado como XOR, EOR< EXOR es un tipo de disyuncion lógica de dos operandos que es verdad si solo un operando es verdad pero no ambos 14, 36

Publius Censor-Resistent Publishing Protocol: http://cs1.cs.nyu.edu/waldman/publius/

# ANEXOS

Anexo A: Criptografía de clave secreta, Flujo_1.pdf

Anexo B: Cipher Modes of Operation and Stream Ciphers, Modes-Operations-RC4

Anexo C: Test de Encriptación, Test de encriptación. JPG

Anexo D: Cifrado asimétrico, Semana 14 SE14501 SHA, HASH.PDF

Anexo E: Metodología SRUM, scrum2.pdf

Anexo F: Gestión de proyectos informáticos Scrum, metodología scrum.pdf

Anexo G: Cuaderno de notas observatorio Esteganografía, Esteganografía.pdf

Anexo H: Tipos de cifrado, tiposdecifrado-100517120808.pdf

Anexo I: El cifrado de datos como medida de seguridad, TutorialPGP.pdf

Anexo J: Cifrado de la información, guía_cifrado_corporativo_2014.pdf

Anexo K: Criptología, leyescifrardatos.pdf

Anexo L: Una introducción matemática a la criptografía, encriptación_algebraI.pdf

Anexo M: Criptografía, Matematica.pdf

Anexo N: Criptografía en el aula de matemática, Encriptación matemática.pdf

Anexo O: Método de encriptación basado en el algoritmo R.S.A, CriptografiaAlgRSA.pdf

Anexo P: Certificados digitales y estándar PKCS, 17.ppt

Anexo Q: Protocolos Criptográficos, Protocols.pdf

Anexo R: Criptografía simétrica y asimétrica, criptograf_a_sim_trica_y_asim_trica.pdf

Anexo S: Códigos de encriptación, pequenos_codigos_para_ser_implemenado.rar

Anexo T: Testing de encriptación, TestingEncryptOrDecryptStringInVB.NET.rar

Anexo U: Código PHP, PHP.php

Anexo V: Videos, https://www.youtube.com/watch?v=H20OPj7FBaw

Anexo W: Entrevista, http://assange.rt.com/es/episodio-9--assange-y-los-criptopunks/full-

translation-text/#page-1

Anexo X: Video, http://assange.rt.com/es/episodio-3--entrevista-al-nuevo-presidente-de-la-repblica-de-tnez-moncef-marzouki/full-translation-text/#page-1

Anexo Y: Video, http://assange.rt.com/es/episodio-3--entrevista-al-nuevo-presidente-de-la-repblica-de-tnez-moncef-marzouki/full-translation-text/#page-1

Anexo Z: Video, http://assange.rt.com/es/episodio-4-entrevista-con-el-escritor-y-activista-egipcio-alaa-abd-el-fattah-y-con-nabeel-rajab-director-del-centro-de-derechos-humanos-de-bahrin/full-translation-text/#page-1

Anexo AA: Video, http://assange.rt.com/es/julian-assange-el-mundo-del-manana/full-translation-text/#page-1

Anexo AB: Video, http://assange.rt.com/es/el-mundo-del-maana-episodio-5-assange-y-correa-la-esperada-entrevista-en-rt/full-translation-text/#page-1

Anexo AC: Video, http://assange.rt.com/es/episodio-7--ocupa-londres-2012/full-translation-text/#page-1

Anexo AD: Video, http://assange.rt.com/es/episodio-9--assange-y-los-criptopunks/full-translation-text/#page-1

Anexo AE: Video, http://assange.rt.com/es/episodio-10--assange-y-imran-khan-niazi/full-translation-text/#page-1

Anexo AF: Video, http://assange.rt.com/es/episodio-10--noam-chomsky-y-tariq-al/full-translation-text/#page-1

Anexo AG: Criptopunks. La libertad y el futuro de internet, Julian A.pdf

Anexo AH: Manual de usuario, Manual de usuario JohannaAndres.pdf

Anexo AI: Manual Técnico, Manual Técnico JohannaAndres.pdf

Anexo AJ: Casos de Usos, Casos de uso.pdf

Anexo AK: Diagrama General y análisis de requerimientos, RequerimientosDiagrama.pdf