ANALISIS Y EVALUACIÒN DE LOS MÈTODOS CRIPTOGRAFICOS BASADOS EN LA SEGURIDAD DE LA INFORMACIÒN PARA LA CORPORACION EDUCATIVA MAYOR DEL DESARROLLO SIMON BOLIVAR

ULFRAN DIAZ BOLIVAR
JOHANA OSORIO MENA

Director
Luisa Arrieta
Ingeniera de Sistemas

CORPORACIÓN EDUCATIVA MAYOR DEL DESARROLLO
SIMON BOLIVAR
FACULTAD DE INGENIERIA DE SISTEMAS
AREA DE INVESTIGACIÓN FORMATIVA
BARRANQUILLA
AÑO 2003

NOTA DE ACEPTACIÓN

_____

_____

_____

Presidente del Jurado

_____

Jurado

_____

Jurado

Barranquilla, 14 de Noviembre de 2003

CONTENIDO

PAG.

INTRODUCCION

Durante las primera décadas de su existencia, las redes de computadores fueron usadas principalmente por investigadores universitarios para el envío de correo electrónico, en ese tiempo la seguridad no recibió mucha atención debido a que no tenia un buen desarrollo tecnológico, no existían las herramientas necesarias para que pudiera ser ejecùtado.

En la actualidad, las redes de informática y mucho mas, en el campo de la seguridad es muy importante debido a que con esta nueva tecnología, como son los diseños del software se hace mas fácil la vida cotidiana del ser humano

Se pretende hacer un Analisis y Evaluación de los Métodos de seguridad de la Información y realizar un buen uso de sus Aplicaciones.

1. PLANTEAMIENTO DEL PROBLEMA

1.1 DESCRIPCION DEL PROBLEMA

Actualmente la Corporación Educativa Mayor del Desarrollo Simón Bolívar no cuenta con una herramienta de Seguridad de la Información que le brinde a la Comunidad Estudiantil un conocimiento Teórico- Práctico de los diferentes Métodos de Criptografia.

No se cuenta con la disponibilidad de Equipos con los cuales los Estudiantes puedan realizar prácticas sobre estos métodos.

1.2 FORMULACION DEL PROBLEMA

¿ Con el Analisis y Evaluaciòn de los Metodos criptograficos aplicados a la Seguridad de la Informaciòn se lograra facilitar el entendimiento y mejorar el interés por parte de la comunidad Estudiantil en este tema?

2. OBJETIVOS

2.1 OBJETIVO GENERAL

Analizar y evaluar los diferentes métodos de criptografía mediante pruebas constantes obteniendo conclusiones eficientes que beneficien a la comunidad Estudiantil de la Corporación Educativa Mayor del Desarrollo Universidad Simón Bolívar.

2.2 OBJETIVOS ESPECÍFICOS

- *Describir los principales métodos de cifrado usando algoritmos de criptografía.*
- *Identificar las diferentes características de cada uno de los métodos de criptografía :(Cifrados Tradicionales: Sustitución, Transposición. Simétricos: cifrado Sencillo,Blowfish. Asimetricos: Des, Cifrado Publico). Etc.*
- *Analizar las ventajas y desventajas de cada uno de los métodos de cifrado para determinar el tipo de aplicaciones en la que puede ser utilizado eficientemente.*
- *Implementar los diferentes métodos de criptografía sometiéndolos a pruebas para evaluar su comportamiento ante diferentes condiciones.*

3. JUSTIFICACION

La Corporación Educativa Mayor del Desarrollo Simón Bolívar se beneficiara con este proyecto ya que este software pueden utilizarlo como una herramienta que les brindará mayor conocimiento a la Comunidad Estudiantil.

Este es unsoftware que es necesario para la universidad por que con el se profundiran los conceptos con respecto a los metodos criptograficos y los estudiantes podran hacer pruebas con los diferentes metodos y asi llegar a lo que nesecitamos saber cual es el metodo mas eficiente y mas seguro.

Los Estudiantes se beneficiarán ya que tendrán acceso a un software que les Mostrará de una forma sencilla el funcionamiento de cada uno de los Métodos Criptográficos y sus Aplicaciones.

4. MARCO DE REFERENCIA

4.0. MARCO HISTORICO

Ya antes del empleo masivo de computadores, e incluso mucho antes de su invención, se habían desarrollado sencillos algoritmos de encriptación. Solía ocurrir con ellos que, conociendo el método de encriptación se podía llegar a descifrar el mensaje sin conocer la clave.

La llegada de los computadores comienza a complicar las cosas. Asimismo, la Segunda Guerra Mundial hace que los países en contienda se interesen en la criptografía, desarrollando máquinas de cifrado entre las que podrían destacarse las máquinas Enigma del ejército alemán. Los algoritmos empiezan a utilizar claves aleatorias, y gracias a diversos avances en la investigación y el incremento tan rápido de las potencias de cálculo se ha llegado a algoritmos por ahora indestructibles como el IDEA, diseñado en el instituto ETH de Zurich en 1990 por James L. Massey y Xuejia Lai.

Paralelamente, surge otro tipo de criptografía, la criptografía de clave pública, cuyos principios se explicarán con detalle más adelante. Su historia comienza en 1976, cuando Whitfield Diffie y Martin Hellman desarrollan el algoritmo DH.

Al año siguiente, 1977, Ron Rivest, Adi Shamir y Leonard Adleman, del Massachussets Institute of Technology (MIT) diseñan un nuevo algoritmo muy potente, el RSA. Y tan potente era que la Agencia de Seguridad Nacional (NSA) del Gobierno americano les sugiere no publicarlo. Haciendo caso omiso, y sobre todo por temor a que más adelante no se les sugiriese, sino que les prohibiese, publican el algoritmo en la revista Scientific American. El algoritmo es patentado a nombre de la compañía RSA Data Security Inc., siendo válida esta

patente en Estados Unidos y Canadá (esto como veremos provocará problemas).

Unos años después Ron Rivest diseña un algoritmo para producir extractos de mensajes (en inglés, message digests), el MD5, siendo utilizado para comprobar que los mensajes no han sido alterados.

En 1991 empieza a extenderse el rumor en Estados Unidos de que el Gobierno quiere prohibir el empleo de la criptografía en líneas de comunicación. Por ello, el programador Phillip Zimmermann , combinando el mejor algoritmo existente de clave única, el IDEA, con el mejor de clave pública, el RSA, y añadiendo el MD5 para las firmas digitales, crea el programa PGP y lo distribuye como freeware por decenas de BBSs. Su intención, como él mismo declaró más adelante, era conseguir que una tecnología tan poderosa llegara a la gente y que no se quedara en manos de los gobiernos, dejando indefensa la intimidad del ciudadano. Consigue además que el Gobierno americano no siga adelante con su prohibición, ya que el PGP se había convertido en el standard de facto para encriptar mensajes de correo electrónico en Internet, y era ya demasiado tarde para detener su expansión.

Un tiempo más tarde surgiría el conflicto. Alguien envió en junio de 1991 a varios grupos de USENET una copia de PGP, y así empezó a distribuirse y utilizarse fuera de Estados Unidos, incumpliendo la legislación ITAR del Departamento de Estado americano sobre exportación de armas. La criptografía está presente en esta lista de productos prohibidos, en el mismo nivel que las ametralladoras de gran calibre, los tanques, las armas químicas... (¡literalmente !). Por otro lado, RSA Data reclamaba a Zimmermann el copyright del algoritmo RSA que usa en el PGP, sufriendo así una doble demanda.

Se llegó a una solución doble. Se crearon dos versiones paralelas de PGP : las

de uso exclusivo para Estados Unidos y Canadá, que emplean una biblioteca de funciones, la RSAREF, de RSA Data y que no son exportables a otros países, encargándose el MIT de la distribución gratuita del PGP ; y las de uso internacional (identificadas añadiendo al número de versión una i), que emplean la biblioteca MPILIB de Zimmermann y que desde la primera versión salida de Estados Unidos han sido desarrolladas en Europa para ahorrarle más quebraderos de cabeza a Zimmermann.

Sin embargo, Zimmermann fue denunciado por violar la normativa ITAR. Esta causa terminó su curso el 11 de enero de 1996 al recibir el abogado de Zimmermann la siguiente carta :
"La Oficina del Fiscal del Distrito Norte de California ha decidido que su cliente, Philip Zimmermann, no será juzgado por el envío a USENET en junio de 1991 del programa de encriptación Pretty Good Privacy. El caso queda cerrado".

También se envió este comunicado a la prensa :
"Michael J. Yamaguchi, Fiscal del Distrito Norte de California, ha anunciado que su Oficina ha decidido abandonar la causa contra cualquier individuo supuestamente implicado en el envío a USENET en junio de 1991 del programa de encriptación Pretty Good Privacy. El caso queda cerrado, y la Oficina no hará más declaraciones".

La demanda de copyright de RSA Data también siguió adelante, celebrándose el juicio en 1996, y siendo Zimmermann absuelto. Zimmermann creó su propia compañía, Pretty Good Privacy Inc., que desarrolló versiones comerciales de PGP y otros programas... sólo para Estados Unidos, a la vez que dentro de Estados Unidos y fuera se siguieron desarrollando nuevas versiones freeware de PGP hoy en día para casi todas las plataformas existentes (PC, Mac, Unix, VAX...).

En 1997, el mundo de PGP sufrió alguna "agitación". En primer lugar, surgió la primera versión para Windows, la 5, con un cómodo empleo de menús y ventanas. Asimismo, PGP Inc. fue adquirida por Network Associates, y Zimmermann quedó relegado a un puesto de asesor. Desde entonces, ha ido surgiendo un desbarajuste de versiones distintas, que han incorporado DH, que han abandonado RSA (aunque las versiones internacionales resisten, y lo mantienen), que han incorporado nuevas funcionalidades. Mientras tanto, la humilde versión 2.6.3, de línea de comandos, la última antes de este "salto", ha seguido al pie del cañón y se sigue utilizando. Su empleo de la línea de comandos y su limitación al RSA-IDEA-MD5 (problema que puede suplirse si se usa como programa "complementario" gnuPG, que utiliza DH) son sus desventajas, pero su rapidez, el poco espacio que ocupa, (y por qué no, ese ligero romanticismo de manejar la versión "legendaria", con la que PGP se expandió en sus "tiempos difíciles") le auguran aún algo de vida.

4.1. MARCO TEÓRICO

Históricamente, cuatro grupos de personas han usado y contribuido el arte del cifrado: los militares, el cuerpo diplomático, los diaristas y los amantes. De éstos, los militares han tenido el papel más importante y han allanado el camino. Dentro de las organizaciones militares, los mensajes a cifrar tradicionalmente se han entregado a empleados mal pagados para su codificación y transmisión. El inmenso volumen de los mensajes ha impedido asignar esta labor a unos cuantos especialistas de elite.

Hasta la llegada de las computadoras, una de las restricciones principales del cifrado había sido la capacidad del empleado encargado de la codificación para realizar las transformaciones necesarias, frecuentemente en un campo de batalla con poco equipo. Una restricción adicional ha sido la dificultad de cambiar rápidamente de un método de cifrado a otro, puesto que esto significa el reentrenamiento de una gran cantidad de gente. Sin embargo, el peligro de que un empleado fuera capturado por el enemigo ha hecho indispensable la capacidad de cambiar el método de cifrado al instante, de ser necesario. De estos requisitos en conflicto se deriva el modelo de la ***figura 1****:*

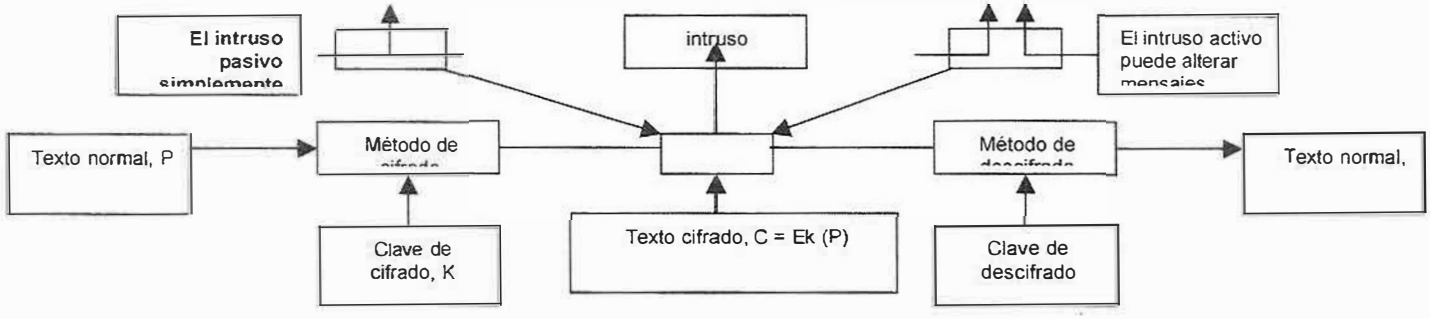


Figura 1.

Los mensajes a cifrar, conocidos como texto normal, se transforman mediante una función parametrizada por una clave. La salida del proceso de cifrado, conocida como texto cifrado, se transmite después, muchas veces mediante mensajero o radio. Suponemos que el enemigo, o intruso, escucha y copia con

exactitud el texto cifrado completo. Sin embargo, a diferencia del destinatario original, el intruso no conoce la clave de cifrado y no puede descifrar fácilmente el texto cifrado. A veces el intruso no sólo puede escuchar el canal de comunicación (intruso pasivo) sino que también puede registrar mensajes y reproducción después, inyectar sus propios mensajes y modificar los mensajes legítimos antes de que lleguen al destinatario (intruso activo). El arte de descifrar se llama criptoanálisis. El arte de diseñar cifradores (criptografía) y de descifrarlos (criptoanálisis) se conoce colectivamente como criptología.

A menudo resulta útil tener una notación para relacionar el texto normal, el texto cifrado y las claves. Usaremos $C = E_k(P)$ par indicar que el cifrado del texto normal P usando la clave K da el texto cifrado C. Del mismo modo, $P = D_k(C)$ representa el descifrado de C para obtener el texto normal nuevamente, por tanto:

$$D_k(E_k(P))=P$$

Esta notación sugiere que E y D son sólo funciones matemáticas, lo cual es cierto. El único truco es que ambas son funciones de dos parámetros, y hemos escrito uno de los parámetros (la clave) como subíndice, en lugar de cómo argumento, para distinguirlo del mensaje.

Una regla fundamental de la criptografía es que se debe suponer que el criptoanalista conoce el método general de cifrado usado. En otras palabras, el criptoanalista sabe cómo funciona el método de cifrado. La cantidad de esfuerzo necesario para inventar, probar e instalar un método nuevo cada vez que el viejo está en peligro, o se piensa que lo está, siempre ha hecho impráctico mantenerlo en secreto, y el pensar que es secreto cuando no lo es hace más daño que bien.

Aquí es donde entra la clave. La clave consiste en una cadena corta (relativamente) que selecciona uno de muchos cifrados potenciales. En contraste con el método general, que tal vez se cambie cada cierto número de años, la clave puede cambiarse con la frecuencia requerida. Por tanto, el modelo básico es un método general estable y conocido públicamente pero parametrizado por una clave secreta y fácilmente cambiable.

La naturaleza no secreta del algoritmo debe subrayarse. Al hacer público el algoritmo, el criptógrafo recibe asesoría gratuita de una gran cantidad de criptólogos académicos ansioso por descifrar el código del sistema para que puedan publicar trabajos demostrando su inteligencia. Si muchos expertos han tratado de descifrar el algoritmo durante cinco años después de su publicación y nadie lo ha logrado, probablemente es bastante sólido.

El secreto real es la clave, y su longitud es un aspecto importante del diseño. Considere una cerradura de combinación. El principio general es que se introducen dígitos en secuencia. Todo el mundo lo sabe, pero la clave es secreta. Una longitud de dos dígitos significa que hay 100 posibilidades. Una clave de tres dígitos de longitud significa 1000 posibilidades, y una clave de seis dígitos de longitud significa un millón. Cuanto más grande es la clave, mayor será el factor trabajo que tendrá que enfrentar el criptoanalista. El factor de trabajo para descifrar el sistema mediante una búsqueda exhaustiva del espacio de claves crece exponencialmente con la longitud de la clave. El secreto radica en tener un algoritmo robusto, pero público, y una clave larga. Para evitar que su hermano menor lea su correo electrónico las claves de 64 bits son suficientes. Para mantener a raya gobiernos poderosos, se requieren claves hasta de 256 bits.

Cifrados por Sustitución

En un Cifrado por sustitución, cada letra o grupo de letras se reemplaza por otra letra o grupo de letras para disfrazarla. Uno de los cifrados más viejos conocidos es el Cifrado de César, atribuido a Julio César. En este método, a se vuelve D, b se vuelve E, c se vuelve F,..., y z se vuelve C. Por ejemplo, ataque se vuelve DWDTXH. En los ejemplos, el texto normal se presentará en minúscula y el texto cifrado en mayúsculas.

Una pequeña generalización del cifrado de César permite que el alfabeto de texto cifrado se desplace K letras, en lugar de siempre 3. En este caso, K se convierte en una clave del método general de alfabetos desplazados circularmente. El cifrado de cesar posiblemente engañó a los cartagineses, pero no ha engañado a nadie desde entonces.

La siguiente es hacer que cada uno de los símbolos del texto normal, digamos las 26 letras del abecedario (no incluida la ñ) inglés, tengan una correspondencia con alguna otra letra, por ejemplo:

***Texto normal**: a b c d e f g h i j k l m n o p q r s t u v w x y z*
***Texto cifrado**: Q W E R T Y U I O P A S D F G H J K L Z X C V B N M*

Este sistema general se llama sustitución monoalfabetica, siendo la clave la cadena de 26 letras correspondiente al alfabeto completo. Para la clave anterior, el texto normal ataque se transforma en el texto cifrado QZQJXT.

A primera vista, esto podría parecer un sistema seguro, porque, aunque el criptoanalista conoce el sistema general (sustitución letra por letra), no sabe cual de las 26! = 4 x 10^{26} claves posibles se está usando. En contraste con el

cifrado de César. Intentarlas todas no es un enfoque muy prometedor. A una tasa de 1 μseg por solución, una computadora tardaría 10^{13} años en intentar todas las claves.

No obstante, si se cuenta con una cantidad aún pequeña de texto cifrado, el cifrado puede descifrase fácilmente. El ataque básico aprovecha las propiedades estadísticas de los lenguajes naturales. Un criptoanalista que intenta descifrar una codificación monoalfabetica comenzaría por contar la frecuencia relativa de todas las letras del texto cifrado.

Cifrados por Transposición

Los cifrados por sustitución conservan el orden de los símbolos de texto normal, pero los disfrazan. Los cifrados por transposición, en contraste, reordenan las letras pero no las disfrazan.
La clave del cifrado es una palabra o frase que no contiene letras repetidas. En este ejemplo, la clave es MEGABUCK. El propósito de la clave es numerar las columnas, estando la columna 1 bajo la letra clave más cercana al inicio del alfabeto, y así sucesivamente. El texto normal se escribe horizontalmente, en filas. El texto cifrado se lee por columnas, comenzando por la columna cuya letra clave es la más baja.

Para descifrar un cifrado por transposición, el criptoanalista debe primero estar consciente de que está tratando con un cifrado de transposición. El siguiente paso es adivinar la cantidad de columnas. En muchos casos, puede adivinarse una palabra o frase probable por el contexto del mensaje. El paso restante es ordenar las columnas. Cuando la cantidad de columnas, k, es pequeña, puede examinarse cada uno de los pares de columnas k(k-1) para ver si la frecuencia de sus diagramas es igual a la del texto normal. El par con la mejor concordancia se supone correctamente ubicado. Ahora cada columna restante se prueba tentativamente como el sucesor de este par. La columna cuyas

frecuencias de diagramas y trigramas produce la mejor concordancia se toma tentativamente como correcta. La columna antecesora se encuentra de la misma manera. El proceso completo se repita hasta encontrar un orden potencial. Es probable que el texto normal sea reconocible en este punto.

Algunos cifrados de transposición aceptan un bloque de longitud fija como entrada y producen un bloque de longitud fija como salida. Estos cifrados pueden describirse por completo con sólo dar una lista que indique el orden en el que deben salir los caracteres.

Dos principios criptográficos fundamentales:

El primer principio es que todos los mensajes cifrados deben contener redundancia, es decir, información no necesaria para entender el mensaje. Un ejemplo puede dejar en claro la necesidad de esto. Considere una compañía de compras por teléfono, el Perezoso (EP), que tiene 60.000 productos. Pensando que son muy eficientes, los programadores de EP deciden que los mensajes de pedidos deben consistir en un nombre de cliente de 16 bytes seguido de un campo de datos de 3 bytes (1 byte para la cantidad y 2 para el número del producto). Los últimos 3 bytes deben cifrase usando una clave muy grande conocida sólo por el cliente y EP.

Inicialmente, esto podría parecer seguro, y en cierto sentido lo es, puesto que los intrusos pasivos no pueden descifrar los mensajes. Desafortunadamente, el sistema también tiene una falla mortal que lo vuelve inútil. Supóngase que un empleado recientemente despedido quiere vengarse de EP por cesarlo. Antes de irse, se lleva con él (parte de) la lista de clientes; entonces trabaja toda la noche escribiendo un programa para generar pedidos ficticios usando nombres reales de los clientes. Dado que no tiene la lista de claves, simplemente pone números aleatorios en los últimos 3 bytes y envía cientos de pedidos a EP.

Al llegar estos mensajes, la computadora de EP usa el nombre del cliente para localizar la clave y descifrar el mensaje. Para mala suerte de EP, cada mensaje de 3 bytes es válido, por lo que la computadora comienza a imprimir instrucciones de embarque. Aunque podría parecer extraño que un cliente ordene 137 juegos de columpios para niño, o 240 cajas de arena, en lo que a la computadora concierne el cliente bien podría estar planeando abrir una cadena de lotes con juegos infantiles. De esta manera, un intruso activo (el ex empleado) puede causar muchísimos problemas, aún cuando no pueda entender los mensajes que genera su computadora.

Este problema puede resolverse agregando redundancia a todos los mensajes. Por ejemplo si se extienden los mensajes de pedido a 12 bytes, de los cuales los primeros 9 deben ser ceros, entonces este ataque ya no funciona, porque el ex empleado ya no puede generar una cadena grande de mensajes válidos. La moraleja de esta historia es que todos los mensajes deben contener una cantidad considerable de redundancia para que los intrusos activos no puedan enviar basura al azar y lograr que se interprete como mensajes válidos.

Sin embargo, la adición de redundancia también simplifica a los criptoanalistas el descifrado de los mensajes. Supóngase que el negocio de pedidos por correos es altamente competido, y que la competencia principal de EL Perezoso, el Bolsón, estaría encantado de conocer la cantidad de cajas de arena que EP vende. Para ello, ha intervenido la línea telefónica de EP. En el esquema original con mensajes de 3 bytes, el criptoanalisis era prácticamente imposible puesto que, tras adivinar una clave, el criptoanalista no tenía manera de saber si había adivinado correctamente.

Por tanto, el principio criptográfico número uno es que todos los mensajes deben contener redundancia para evitar que los intrusos activos engañen al receptor y lo hagan actuar ante un mensaje falso. Sin embargo, esta misma

redundancia simplifica mucho la violación del sistema por parte de los intrusos pasivos, por lo que aquí tenemos un poco tensión. Es más, la redundancia nunca debe tener la forma de n ceros al inicio o al fin de mensaje, ya que el análisis de tales mensajes con algunos algoritmos criptográficos da resultado más predecibles, simplificando la tarea del criptoanalista. Una cadena aleatoria de palabras sería mejor para incluir la redundancia.

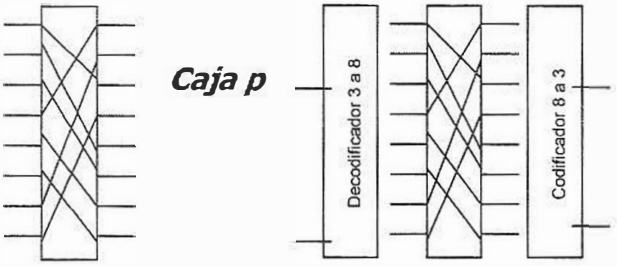
El segundo principio criptográfico es que deben tomarse algunas medidas para evitar que los intrusos activos reproduzcan mensajes viejos. Si no se toman tales medidas, nuestro ex empleado podría conectarse a la línea telefónica de EP y simplemente continuar repitiendo mensajes válidos enviados previamente. Una de tales medidas es la inclusión en cada mensaje de una marca de tiempo válida durante, digamos, 5 minutos. El receptor puede entonces guardar los mensajes unos 5 minutos, para compararlos con los mensajes nuevos que lleguen y filtrar los duplicados. Los mensajes con mayor antigüedad que 5 minutos pueden descartarse, dado que todas las repeticiones enviadas más de 5 minutos después también se rechazarán como demasiado viejas.

Algoritmos de clave secreta

La criptografía moderna usa las mismas ideas básicas que la criptografía tradicional, la transposición y la sustitución, pero su orientación es distinta. Tradicionalmente, los criptógrafos han usado algoritmos sencillos y se han apoyado en claves muy largas para la seguridad. Hoy día es cierto lo inverso: el objetivo es hacer el algoritmo de cifrado tan complicado y rebuscado que inclusive si el criptoanalista obtiene cantidades enormes de texto cifrado a su gusto, no será capaz de entender nada.

Las transposiciones y las sustituciones pueden implantarse mediante circuitos sencillos. En la figura se muestra un dispositivo, conocido como caja P (la P

significa permutación), que se usa para efectuar una transposición de entrada de 8 bits. Si se designan los 8 bits de arriba hacia abajo como 01234567, la salida de esta caja P en particular es 36071245. Mediante el alambrado interno adecuado, puede hacerse que una caja P efectúe cualquier transposición, y lo puede hacer prácticamente a la velocidad de la luz.



Las sustituciones se llevan a cabo mediante cajas S, como se muestra en la ***figura***

2. Data Encryption Estándar (DES).

El DES es un sistema de cifrado de bloques. En este caso, el algoritmo toma la información en bloques de 64 bits produciendo un bloque de texto cifrado también de 64 bits. Las claves utilizadas por este sistema son de 56 bits, aunque se suelen distribuir en forma de un número de 64 bits, donde cada octavo bit (el lsb o less-significant bit) de cada uno de los ocho bytes de la clave es un bit de paridad.

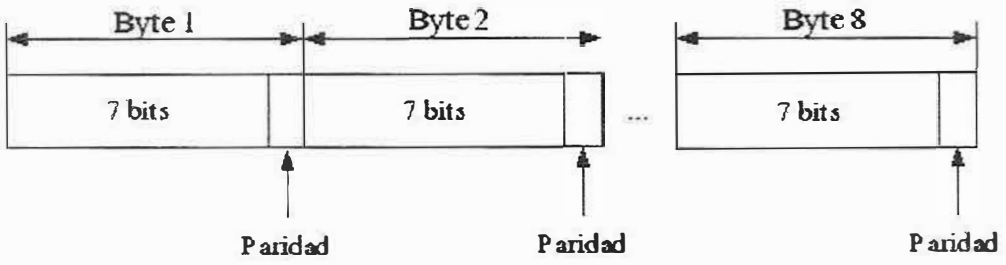


Figura 3 - Esquema clave DES

Algoritmo de cifrado.

Centrémonos ahora en el algoritmo de cifrado. ***Figura 4***

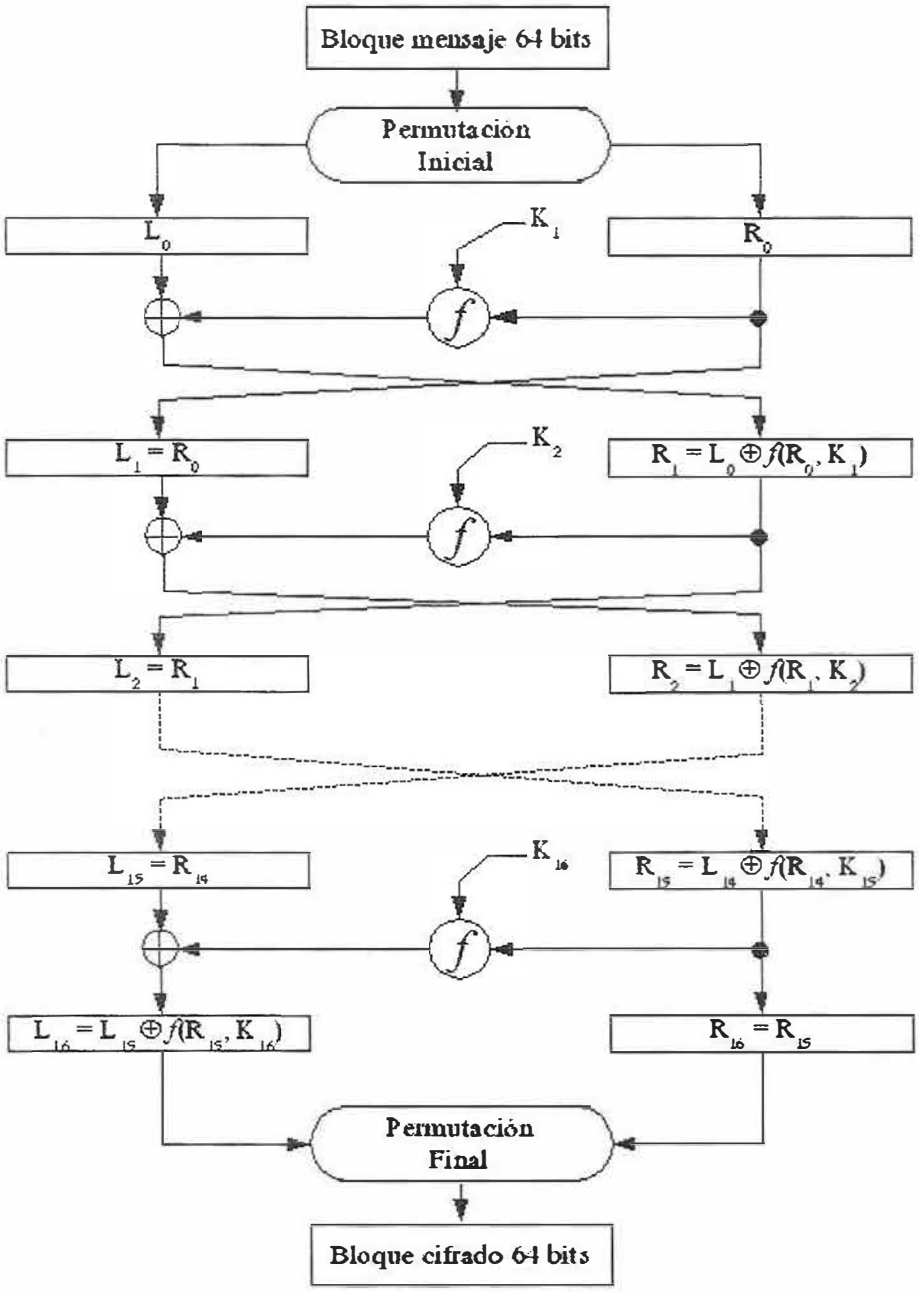


Figura 4- Esquema del DES

Lo primero que se realiza es una permutación de los 64 bits del bloque de entrada. Realmente, esta permutación no añade seguridad alguna y su principal función es la de facilitar la carga de los bits de información en bloques de 8 bits

a un chip especializado en DES (debemos recordar que el desarrollo del DES es anterior a los procesadores de 16 o 32 bits).

Al final del algoritmo hay otra permutación que es la inversa de esta, lo que vuelve a dejar a los bits en su posición inicial.

58	50	42	34	26	18	10	02	60	52	44	36	28	20	12	04
57	49	41	33	25	17	09	01	59	51	43	35	27	19	11	03
57	49	41	33	25	17	09	01	59	51	43	35	27	19	11	03
61	53	45	37	29	21	13	05	63	55	47	39	31	23	15	07

Tabla 1 – Permutación inicial

Esta tabla debe leerse de izquierda a derecha y de arriba a bajo. Esto quiere decir, que el primer bit a la salida de la permutación es el que en la entrada ocupa la posición 58; el segundo, el que ocupaba la 50; y así sucesivamente.

Después de esta permutación inicial, los 64 bits resultantes se dividen en dos partes de 32 bits cada una. Lo que viene a continuación se repite dieciséis veces:

1. *La mitad de la derecha (R_i) se introduce en una función (f) donde es combinada con la clave.*
2. *Se realiza una XOR con el resultado de la función y con la mitad de la izquierda (L_i).*
3. *La parte de la derecha de esta ronda (R_i) pasará a ser la parte izquierda de la siguiente (L_{i+1}) y el resultado de la XOR anterior pasará a ser la parte de la derecha de la ronda siguiente (R_{i+1}).*

$L_{i+1} = R_i$

$R_{i+1} = L_i \text{ XOR } f(R_i, K)$

La única excepción a este proceso está en la última ronda, en la que el entrecruce de las dos partes no se produce.

El esquema de esta función f es el siguiente ***Figura 5****:*

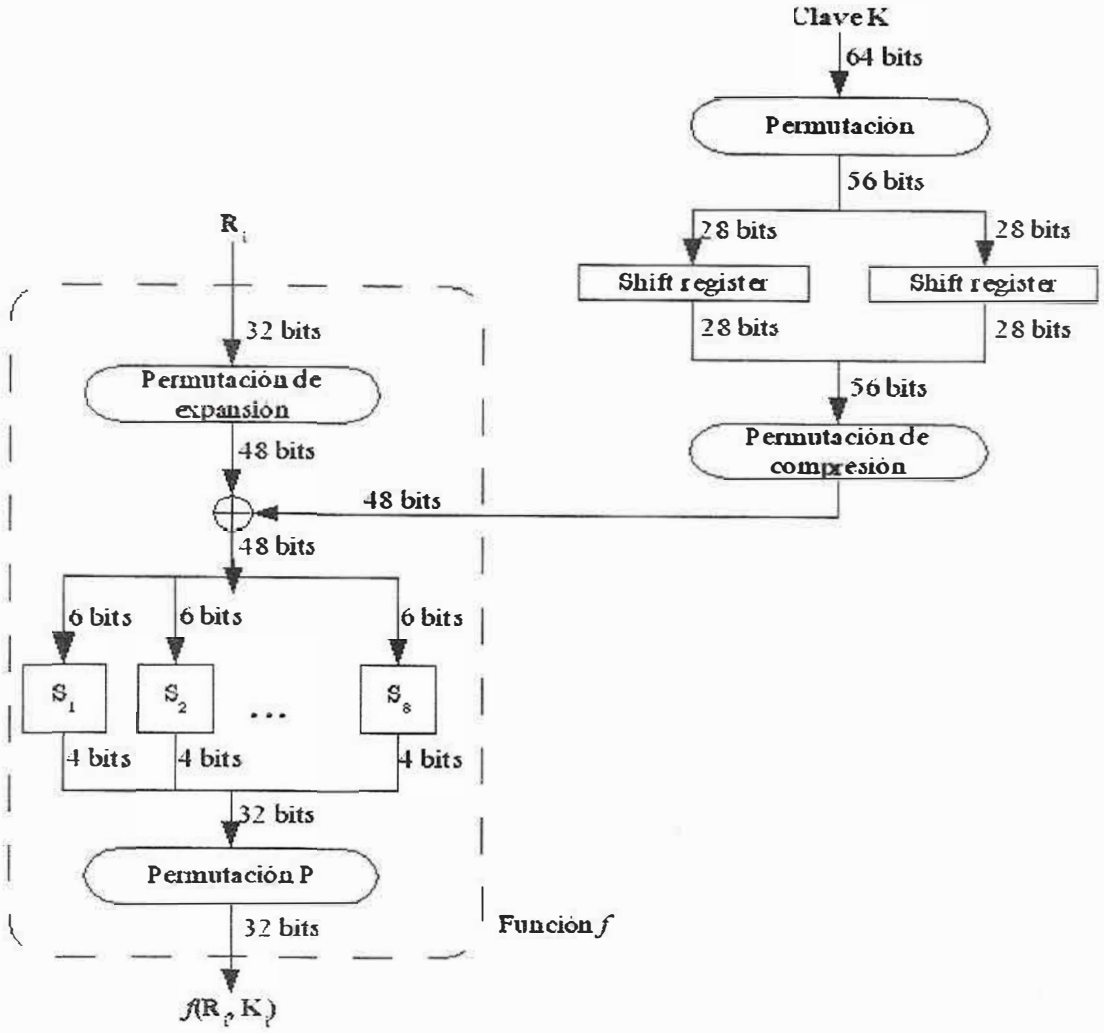


Figura 5 – Esquema de la función f

En cada ronda, se genera una subclave K_i distinta de la siguiente manera: como ya habíamos dicho, de los 64 bit de la clave, se descartan los 8 bits de paridad quedando una clave de 56 bits. Esto se realiza mediante la siguiente permutación:

57	49	41	33	25	17	09	01	58	50	32	34	26	18
10	02	59	51	43	35	27	19	11	03	60	52	44	36
63	55	47	39	31	23	15	07	62	54	46	38	30	22
14	06	61	53	44	37	29	21	13	05	28	20	12	04

Tabla 2 - Permutación de la clave

Como se puede observar, no aparecen los bits 8, 16, 24, 32, 40, 48, 56 ni 64, que son los que hacen la función de bits de paridad.

La clave resultante se divide en dos mitades de 28 bits que alimentan unos registros que rotan hacia la izquierda cada una de las mitades un número de bits que viene determinado por la ronda en la que nos encontramos (recordar que en el DES existen 16 rondas)

Ronda

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16

Desplazamiento

1	1	2	2	2	2	2	2	1	2	2	2	1	2	2	1

Tabla 3 - Desplazamientos según la ronda para el cifrado

De los 56 bits resultantes se seleccionan 48 modificando su orden. Esta operación es denominada permutación de compresión.

14	17	11	24	01	05	03	28	15	06	21	10
23	19	12	04	26	08	16	07	27	20	13	02
41	52	31	37	47	55	30	40	51	45	33	48
44	49	39	56	34	53	46	42	50	36	29	32

Tabla 4 - Permutación de compresión

En la salida de la permutación de compresión tendríamos la subclave K_i de 48 bits correspondiente a la ronda i-ésima del algoritmo. Una vez ya hemos explicado como se obtienen las subclaves de las rondas, centrémonos en las modificaciones que sufre la parte de la derecha R_i dentro de la función. Primero, mediante la permutación de expansión, se expanden los 32 bits de R_i a 48 bits (repitiendo alguno de los bits) cambiando, además, su posición. Esto se hace para que este operando tenga el mismo tamaño que la subclave de la ronda, y además hace que los bits del criptograma dependan todavía con más fuerza de los bits de la entrada. Esto hecho se conoce como el efecto avalancha: un pequeño cambio en el mensaje de entrada provoca grandes modificaciones en la secuencia de bits de salida.

32	01	02	03	04	05	04	05	06	07	08	09
08	09	10	11	12	13	12	13	14	15	16	17
16	17	18	19	20	21	20	21	22	23	24	25
24	25	26	27	28	29	28	29	30	31	32	01

Tabla 5 - Permutación de expansión

Los 48 bits resultantes de la XOR entre la parte derecha R_i expandida y la subclave K_i se introducen de seis en seis en ocho bloques que son conocidos como S-Box, de tal manera que el primer bloque de 6 bits se introduce en la S-Box 1, el segundo en la S-Box 2, y así sucesivamente. Como cada S-Box toma 6 bits como entrada y produce 4 bits como salida, al final obtenemos 32 bits (8 S-Box · 4 bits/S-Box = 32 bits)

Cada S-Box es una tabla de cuatro filas y dieciséis columnas, donde cada posición de la tabla es un número de 4 bits. De los 6 bits de la entrada b1 b2 b3 b4 b5 b6, los bits b1 y b6 indican la fila, mientras que los otros cuatro indican la columna. Es decir, si la entrada de una S-Box es 35, esto en binario es 100011, lo que significa que hemos de tomar el valor de la fila 3 (11) y columna 1 (0001). El contenido de cada una de las S-Box es distinta del de las demás.

Contenido de las S-Box

14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

Tabla 6 - S-Box 1

15	01	08	14	06	11	03	04	09	07	02	13	12	00	05	10
03	13	04	07	15	02	08	14	12	00	01	10	06	09	11	05
00	14	07	11	10	04	13	01	04	08	12	06	09	03	02	15
13	08	10	01	03	15	04	02	11	06	07	12	00	05	14	09

Tabla 7 - S-Box 2

10	00	09	014	06	03	15	05	01	13	12	07	11	04	02	08
13	07	00	09	03	04	06	10	02	08	05	14	12	11	15	01
13	06	04	09	08	15	03	00	11	01	02	12	05	10	14	07

01	10	13	00	06	09	08	07	04	15	14	03	11	05	02	12

Tabla 8 - S-Box 3

07	13	14	03	00	06	09	10	01	02	08	05	11	12	04	15
13	08	11	05	06	15	00	03	04	07	02	12	01	10	14	09
10	06	09	00	12	11	07	13	15	01	03	14	05	02	08	04
03	15	00	06	10	01	13	08	09	04	05	11	12	07	02	14

Tabla 9 - S-Box 4

02	12	04	01	07	10	11	06	08	05	03	15	13	00	14	09
14	11	02	12	04	07	13	01	05	00	15	10	03	09	08	06
04	02	01	11	10	13	07	08	15	09	12	05	06	03	00	14
11	08	12	07	01	14	02	13	06	15	00	09	10	04	05	03

Tabla 10 - S-Box 5

12	01	10	15	09	02	06	08	00	13	03	04	14	07	05	11
10	15	04	02	07	12	09	05	06	01	13	14	00	11	03	08
09	14	15	05	02	08	12	03	07	00	04	10	01	13	11	06
04	03	02	12	09	05	15	10	11	14	01	07	06	00	08	13

Tabla 11 - S-Box 6

04	11	02	14	15	00	08	13	03	12	09	07	05	10	06	01
13	00	11	07	04	09	01	10	14	03	05	12	02	15	08	06
01	04	11	13	12	03	07	14	10	15	06	08	00	05	09	02
06	11	13	08	01	04	10	07	09	05	00	15	14	02	03	12

Tabla 12 - S-Box 7

13	02	08	04	06	15	11	01	10	09	03	14	05	00	12	07
01	15	13	08	10	03	07	04	12	05	06	11	00	14	09	02
07	11	04	01	09	12	14	02	00	06	10	13	15	03	05	08
02	01	14	07	04	10	08	13	15	12	09	00	03	05	06	11

Tabla 13 - S-Box 8

Los 32 bits resultantes de la anterior operación entran en la permutación P, en la que los bits son permutados otra vez pero esta vez no se eliminan ni añaden bits como se había hecho en las permutaciones de compresión y expansión.

16	07	20	21	29	12	28	17	01	15	23	26	05	18	31	10

02	08	24	14	32	27	03	09	19	13	30	06	22	11	04	25

Tabla 14 - Permutación P

Para finalizar la ronda, la salida de la permutación P alimenta una XOR en la que el otro operando es la parte izquierda L_i. Si no estamos en la ronda 16, el resultado de la XOR anterior se convierte en la parte derecha de la siguiente ronda R_{i+1}, y la parte derecha de la ronda en la que estamos R_i pasará a ser la parte izquierda de la siguiente L_{i+1}. Y así sucesivamente hasta completar las 16 rondas.

Finalmente, la concatenación de L16 y R16 se introduce en una permutación final, que ya habíamos comentado al inicio de la explicación del algoritmo. Esta permutación es la inversa de la permutación inicial quedando de la siguiente manera:

40	08	48	16	56	24	64	31	39	07	47	15	55	23	63	31
38	06	46	14	54	22	62	30	37	05	45	13	53	21	61	29
36	04	44	12	52	20	60	28	35	03	43	11	51	19	59	27
34	02	42	10	50	18	58	26	33	01	41	09	49	17	57	25

Tabla 15 - Permutación final

Se podría haber hecho que en la ronda final se intercambiaran las partes derecha e izquierda, y adaptar la permutación final para que tuviera en cuenta este intercambio de más. Pero no se hizo porque de esta manera se puede utilizar el mismo algoritmo tanto para el cifrado como para el descifrado.

Algoritmo de Descifrado.

El algoritmo de descifrado es el mismo que el de cifrado, con la única matización de que las subclaves deben utilizarse en el orden inverso. Es decir, si en el algoritmo de cifrado las subclaves de cada ronda eran K1, K2, ..., K16, en el descifrado las subclaves son K16, K15, ..., K1, entendiendo que la primera subclave (K1 en el cifrado y K16 en el descifrado) es la que se utiliza en la primera ronda; la segunda subclave (K2 en el cifrado y K15 en el descifrado), en la segunda ronda; etc.
Además, el algoritmo mediante el cual se generan las subclaves es circular. Así que para generar las subclaves necesarias para el descifrado, los registros de rotación en lugar de desplazar hacia la

Ronda

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16

Desplazamiento

0	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Tabla 16 - Desplazamientos según la ronda para el descifrado

Consideraciones sobre las claves.

Debido al modo en el que se generan las subclaves en las diferentes rondas, existen una serie de claves que no son aptas para ser utilizadas. Las primeras de estas claves son conocidas como claves débiles (o weak keys). La debilidad de estas claves es que todos los bits de cada una de las dos mitades que entran en los registros de rotación son todo unos o todo ceros. Entonces, por mucho que rotemos estas palabras de bits, siempre obtendremos la misma subclave.

Claves Débiles	***Subclave***
101010101010101	*0000000 0000000*
1F1F1F1F0E0E0E0E	*0000000 FFFFFFF*
E0E0E0E0F1F1F1F1	*FFFFFFF 0000000*
FEFEFEFEFEFEFEFE	*FFFFFFF FFFFFFF*

También es posible encontrar pares de claves de tal manera que el criptograma generado por una de las claves del par puede ser descifrado por la otra clave. O lo que es lo mismo, las dos claves del par generan el mismo criptograma a partir del mismo texto en claro. Esto se debe a que esas claves, en lugar de generar 16 subclaves distintas, sólo generan dos subclaves, y cada una es usada ocho veces. A estas claves se le llaman claves semidébiles (semiweak keys) y existen 12 claves que cumplen esta condición.

Para finalizar, existen otras claves que producen únicamente cuatro subclaves distintas, que son usadas cuatro veces cada una. Estas son conocidas como claves posiblemente débiles (possibly weak keys). De este tipo de claves, existen 48.

Pero a pesar de la existencia de estas 64 claves débiles (4 + 12 + 48), hay que decir que son una minúscula porción de las 2^{56} = 72.057.594.037.927.936 claves que puede utilizar el DES.

ALGORITMOS DE CLAVE ÚNICA

PGPfone dispone de tres algoritmos de clave única para la comunicación: CAST, DES Triple y Blowfish. CAST

CAST es un algoritmo diseñado por Carlisle Adams y Stafford Tavares. El algoritmo básico emplea bloques de 64 bits y una clave de 64 bits. La versión que implementa PGPfone usa una clave de 128 bits. También se utilizan seis S-boxes con una entrada de 8 bits y una salida de 32 bits.

S-box es la abreviatura de "caja de sustitución", y no es más que una matriz cargada con determinados valores. Los bits de la entrada, combinados de determinada manera, determinan la fila y la columna del valor que se da como salida. La clave es que las S-boxes no son lineales, y por tanto, difíciles de analizar, estando en ellas la complejidad de los algoritmos de clave única que las utilizan.

Para encriptar, CAST divide el texto (o archivo) a encriptar en bloques de 64 bits, que a su vez son divididos en dos mitades, izquierda y derecha. En cada ronda de encriptación (se usan ocho rondas) se combina la mitad derecha con 16 bits de la clave (subclave) empleando una función f, para después hacer el XOR del resultado de f y la mitad izquierda. La mitad derecha original (anterior a la ronda de encriptación) se convierte en la nueva mitad izquierda para la siguiente ronda. Al terminar las ocho rondas, se concatenan las dos mitades, que formarán el texto (o archivo) encriptado.

La función consiste en:

1. *Dividir los 32 bits de entrada en cuatro partes de 8 bits: a, b, c y d.*
2. *Dividir los 16 bits de la parte de clave usada en la ronda en dos mitades: e y f.*
3. *Utilizar a como entrada para la primera S-box, b para la segunda, c para la tercera, d para la cuarta, e para la quinta y f para la sexta.*

4. *Hacer el XOR de las salidas de las S-boxes para obtener los 32 bits de salida.*

Otra forma de llevar a cabo el algoritmo consiste en hacer el XOR de los 32 bits de entrada y 32 bits de la clave, dividirlos en cuatro partes de 8 bits, pasarlos por las S-boxes, y hacer el XOR de todos. Al parecer, n rondas del algoritmo según este sistema serían tan seguras como n+2 rondas del primero.

Las subclaves de 16 bits se calculan así. Si k_1, k_2,... k8 son los ocho bytes de la clave, las subclaves son:

Ronda 1: k_1, k_2	*Ronda 3: k_5, k_6*	*Ronda 5: k_4, k_3*	*Ronda 7: k_8, k_7*
Ronda 2: k_3, k_4	*Ronda 4: k_7, k_8*	*Ronda 6: k_2, k_1*	*Ronda 8: k_6, k_5*

La fuerza del algoritmo, como comentamos, está en las S-boxes. No se dan unas S-boxes fijas; se crean nuevas para cada aplicación. Hay diversos criterios para hacer unas S-boxes fuertes, por lo que crearlas aleatoriamente no es una buena idea.

CAST, al menos por ahora, ha resistido al criptoanálisis, tanto lineal como diferencial, con lo que el único ataque posible contra CAST sería el de fuerza bruta, teniendo en cuenta que las claves posibles para la implementación básica de CAST son 2^{64}, y que para la implementación usada por PGP son 2^{128}. Nadie puede asegurar que no pueda alcanzarse la potencia computacional suficiente para probar todas esas claves en un tiempo razonable, pero al menos en los próximos años no va a poderse.

DES TRIPLE

DES es quizá uno de los algoritmos de clave única más conocidos. Durante años se ha trabajado sobre él ha sido analizado de muchas maneras, consiguiendo así en sus últimas versiones un diseño muy fuerte y resistente. No hablamos aquí de este diseño, ya que se extendería demasiado. Si alguien está interesado, puede consultar el magnífico libro de Bruce Schneier, Applied Cryptography, editado por John Wiley and Sons, o consultar documentos y

código fuente disponibles a través de Internet. Tan sólo comentar que sigue los mismos principios de diseño que CAST (bueno, en realidad habría que decir que CAST y todos los demás algoritmos de clave única que utilizan este tipo de diseño, siguen los principios de diseño de DES, que es el "padre" de todos): se utilizan rondas, en las que los datos a encriptar se dividen en dos, se combinan con parte de la clave, sufren alguna transformación y pasan por varias S-boxes.

Sin embargo, tiene un talón de Aquiles que lo debilita: se utiliza una clave demasiado pequeña, de 56 bits. Durante años se conjeturó con la posibilidad de construir una máquina que con miles de chips procesando en paralelo, pudiera, en un tiempo razonable, probar todas las claves posibles de DES para desencriptar unos datos dados. También durante años se especuló con que sólo grandes agencias de seguridad podrían pagar algo así... Pero el 12 de julio de 1998, la Electronic Frontier Foundation presentó su DES Cracker , una máquina de procesamiento paralelo que costó 250.000 dólares, algo al alcance de muchos gobiernos. Probando 88 mil millones de claves por segundo, se encontró la clave correcta en 56 horas.

Podemos así ahora afirmar con rotundidad lo que ya se suponía desde hace años: DES no es seguro. Sin embargo, un diseño tan bueno como el de DES no podía desperdiciarse. La solución fue crear nuevas variantes de DES más fuertes, y entre ellas está DES Triple.

Para encriptar con DES Triple, se generan tres claves de 56 bits. Se realiza una encriptación DES de los datos con la primera clave, después se desencriptan con la segunda, y se vuelven a encriptar con la tercera. Para desencriptar se sigue el proceso inverso. Tenemos así un algoritmo de excelente diseño con un total de 2^{112} claves para probar en un ataque de fuerza bruta (sí, deberían ser 2^{168}, pero determinados ataques hacen que el número efectivo de claves sea $2^{112)}$.

BLOWFISH

Blowfish es un algoritmo diseñado por Bruce Schneier, y está diseñado para ejecutarse preferentemente en microprocesadores y satisfacer los siguientes criterios:

1. *Rapidez. Blowfish encripta datos en procesadores de 32 bits a razón de 26 ciclos de reloj por byte.*
2. *Escaso consumo de recursos. Blowfish necesita como mínimo sólo 5 Kb de memoria.*
3. *Simplicidad. Blowfish usa operaciones simples: suma, XOR, y manejo de tablas con operandos de 32 bits. Su diseño es fácil de analizar, y por tanto pueden descubrirse fácilmente errores de implementación.*
4. *Seguridad configurable. La longitud de la clave de Blowfish es variable, y puede llegar hasta 448 bits. En concreto, la implementación de PGPfone utiliza una clave de 192 bits.*

Blowfish maneja bloques de datos de 64 bits. El algoritmo tiene dos partes: expansión de la clave y encriptación de los datos. Se utiliza un array P, formado por 18 subclaves de 32 bits, y 4 S-boxes de 32 bits con 256 elementos cada una. La expansión de la clave convierte una clave de 448 bits en varios arrays de subclaves que totalizan 4168 bytes. Explicaremos más adelante el proceso de cálculo de las subclaves.

El proceso de encriptación está formado por 16 rondas, siendo la entrada un bloque de datos de 64 bits, x. El algoritmo consiste en:

Dividir x en dos mitades de 32 bits: x_L, x_R
Para i = 1 hasta 16
$x_L = x_L$ XOR P_i
$x_R = F(x_L)$ XOR x_R
Pasar x_L a la "derecha" y x_R a la izquierda
Pasar x_L a la "derecha" y x_R a la izquierda (deshace el intercambio de posición de la última ronda)

$x_R = x_R \text{ XOR } P_{17}$

$x_L = x_L \text{ XOR } P_{18}$

Unir x_L y x_R

Para la función F se divide x_L en cuatro partes de 8 bits, a, b, c y d, y se calcula:

$$F(x_L) = ((S_{1,a} + S_{2,b} \bmod 2^{32}) \text{ XOR } S_{3,c}) + S_{4,d} \bmod 2^{32}$$

El proceso de desencriptación sigue el mismo algoritmo, salvo que P_1, P_2,... P_{18} se usan en orden inverso.

El cálculo de las subclaves se realiza utilizando el algoritmo Blowfish:

1. Inicializar el array P y las cuatro S-boxes con una cadena fija, formada por dígitos hexadecimales del número pi.
2. Hacer el XOR de P_1 con los primeros 32 bits de la clave, el XOR de P2 con los siguientes 32 bits... y así con toda la clave, hasta llegar a P_{18}. Si se terminan los bits de la clave antes de haber completado el array P, se vuelve a empezar con el primer bloque de 32 bits.
3. Encriptar una cadena formada por ceros según el algoritmo Blowfish, utilizando las subclaves obtenidas en los pasos 1 y 2.
4. Sustituir P_1 y P_2 por la salida del paso 3.
5. Encriptar la salida del paso 3 usando el algoritmo Blowfish con las nuevas subclaves.
6. Sustituir P_3 y P_4 por la salida del paso 5.
7. Continuar con el proceso, sustituyendo todos los elementos del array P, y después todos los elementos de las S-boxes.

Hace falta un total de 521 iteraciones para obtener todas las subclaves.

Serge Vaudenay ha atacado Blowfish utilizando S-boxes conocidas y r rondas; un ataque con criptoanálisis diferencial podría obtener el array P con 2^{8r+1} textos no encriptados seleccionados (lo que en la bibliografía en inglés se conoce como "chosen plaintext attack", es decir, un ataque en el que se utilizan

para el análisis parejas de textos no encriptados y el correspondiente texto encriptado, seleccionando aquellas parejas que podrían revelar más contenido de la clave que otros). Con algunas claves débiles que generan S-boxes de peor calidad, el mismo ataque necesita 2^{Ar+1} textos no encriptados seleccionados. La posibilidad de dar con una de estas claves es de 1 entre 2^{14}. Si se realiza el ataque sin conocer las S-boxes, se puede detectar si se está usando una clave débil, pero no si se trata de una S-box o del array P. El ataque sólo funciona con versiones de Blowfish que usaran pocas rondas, siendo completamente inútil contra la versión con 16 rondas.

Así pues, no existe por ahora ningún ataque contra Blowfish basado en criptoanálisis que sea efectivo. Quedaría entonces el ataque de fuerza bruta, que en el caso de PGPfone, ha de hacer frente a 2^{192} posibles clave

ALGORITMO DIFFIE-HELLMAN DE INTERCAMBIO DE CLAVES

Dado que el PGPfone del emisor y del receptor utilizan alguno de los algoritmos de clave única mencionados anteriormente para encriptar la conversación, han de ponerse de acuerdo en una clave, y han de hacerlo de forma segura. Si la clave fuera generada por uno de los comunicantes y enviada directamente al otro por el canal inseguro que utilizan, ni que decir tiene que podrían ahorrarse el usar PGPfone:-). Por ello, se utiliza un algoritmo de intercambio de claves, que permite que ambas partes se pongan de acuerdo con una clave sin revelarla.

El algoritmo utilizado es Diffie-Hellman, basado en un sistema de clave pública. Dado que ElGamal es una implementación específica de Diffie-Hellman utilizada para encriptar, su seguridad se basa en los mismos principios, por lo que podéis echar un vistazo a la descripción de ElGamal incluida en las páginas dedicadas a PGP 5 si estáis interesados en el tema.

Para utilizar el algoritmo, en primer lugar ambas partes han de ponerse de acuerdo en un número primo grande, n, y un número g que sea primitivo mod n. Es decir, dado n, número primo, y g, un número menor que n, entonces g es primitivo mod n si para cada valor b de 1 a n-1, existe un valor a tal que $g^a = b$ mod n. En concreto, para agilizar los cálculos, y para evitar los problemas de seguridad que pueden surgir con determinados primos que son más débiles que otros, la implementación de PGPfone utiliza g = 2, y selecciona n de una lista de números primos que incluye el programa. La lista está formada por varias sublistas de números primos de distintos tamaños: 1024 bits, 2048... El proceso de selección es simple: cada una de las partes, A y B, envía una lista de hashes de números primos de la lista (calculados utilizando SHA), colocándose en orden de preferencia (el orden se basa en el tamaño del número primo, y puede modificarse en las opciones de configuración del programa). Se coge el primero de la lista de cada una de las dos partes, y se elige el más grande. El realizar este proceso en un canal inseguro no supone ningún riesgo.

Una vez decidido el número primo n:

1. *A elige un entero aleatorio grande, x, y envía a B X, tal que $X = g^x$ mod n*
2. *B elige también un entero aleatorio grande, y, y envía a A Y, tal que $Y = g^y$ mod n*
3. *A calcula $k = Y^x$ mod n*
4. *B calcula $k' = X^y$ mod n*

Se cumple que $k = k' = g^{xy}$ mod n. Nadie que esté pinchando el canal puede calcular dicho valor, ya que sólo conocen n, g, X e Y, por las razones expuestas en la descripción de ElGamal al hablar del problema del logaritmo discreto.

LOS ALGORITMOS DE PGP

El programa PGP emplea tres algoritmos de encriptado: el RSA, de clave pública, el IDEA, de clave única, y el MD5 para producir digests(recopilación),

combinados de una forma que se explicará más adelante para conseguir la máxima seguridad y la mayor rapidez y comodidad.

EL ALGORITMO RSA

RSA, diseñado por Ron Rivest, Adi Shamir y Leonard Adleman, basa su seguridad en la pertenencia al grupo de los problemas difíciles de la clase NP del problema de la factorización de números grandes. La clave pública y la privada están en función de un par de números primos grandes (de 100 o 200 dígitos). Es decir, hallar la clave privada supondría encontrar por factorización los dos números primos que la forman, un cálculo hoy en día todavía imposible para claves de al menos 1024 bits.

*Para generar las dos claves, se toman aleatoriamente dos números primos grandes, p y q. Para una mayor seguridad, p y q son de la misma longitud. Se calcula n = p*q.*

A continuación, se elige aleatoriamente la clave de encriptación, e, de tal manera que e y (p-1)(q-1) son primos relativos (es decir, de manera que el máximo común divisor de e y (p-1)*(q-1) sea 1; o dicho en otras palabras, que el único factor que compartan e y (p-1)*(q-1) sea 1).*

Por último, se calcula la clave de descifrado, d, tal que:

$d = e^{-1} \bmod ((p-1)*(q-1))$

d y n también son primos relativos. Los números e y n son la clave pública; d es la clave privada. Los dos números primos, p y q, dejan de ser necesarios al terminar el proceso. Pueden ser descartados, pero nunca han de ser revelados.

Para encriptar un mensaje m en primer lugar se divide en bloques numéricos m_i menores que n (por ejemplo, en el caso de datos binarios, se toma la mayor potencia de 2 menor que n), obteniéndose un mensaje encriptado, c formado por varios bloques c_i. La fórmula de encriptación es:

$c_i = m_i^e \bmod n$

Para desencriptar, se toma cada bloque c_i y se calcula:

$m_i = c_i^d \bmod n$

Igualmente, se podría haber encriptado el mensaje con d y descifrarlo con e.

SEGURIDAD DE RSA

Se supone (no es demostrable matemáticamente) que la seguridad de RSA se basa totalmente en el problema de factorización de números grandes, es decir, la factorización de n, hoy en día virtualmente imposible, como se ha dicho, con claves de al menos 1024 bits (conviene ir siempre por delante, y usar ya las de 2048). También sería posible atacar RSA descubriendo el valor (p-1)(q-1), pero este ataque equivale a una factorización de n.*

Otra posibilidad es un ataque de fuerza bruta probando cualquier d posible hasta dar con el valor correcto, pero es un ataque aún menos eficiente.

Han surgido diversos estudios sobre nuevos posibles ataques, pero hasta la fecha no ha surgido ninguno sencillo.

Otra cuestión podría surgir en cuanto a p y q. La mayoría de los algoritmos empleados para calcular primos grandes son probabilísticos; por tanto, ¿qué ocurriría si no son primos?. El problema se resuelve haciendo disminuir esta posibilidad al mínimo, (y así ocurre en PGP, que emplea algunos algoritmos de detección de primalidad) y aunque se diera el caso, el proceso de cifrado y descifrado daría errores con estos números, por lo que el problema sería rápidamente descubierto. Hay unos pocos números, los números de Carmichael, que no son detectados por algunos algoritmos de primalidad y son inseguros. Pero si alguien da aleatoriamente al generar las claves con uno de estos números, entonces es al día siguiente le caerá un rayo justo después de enterarse de que ha acertado la quiniela y la primitiva. :-)

Existen asimismo varios ataques, no contra el algoritmo en sí, sino contra la forma de implementarlo. Explicarlos aquí haría que nos extendiéramos

demasiado, y que se sepa hasta la fecha, la implementación de PGP no es vulnerable a ellos.

EL ALGORITMO IDEA

El algoritmo de clave única IDEA utiliza texto en bloques de 64 bits y una clave de 128 bits. Ha sido diseñado de tal manera que el proceso de encriptado consiste en ocho pasos de encriptación que son idénticos excepto en los sub-bloques de la clave utilizados, terminando con una transformación de la salida. En cada paso se utilizan tres operaciones: suma modular con módulo 2^{16}, multiplicación modular con módulo $2^{16}+1$ y OR exclusivo.

Hay un total de ocho rondas de encriptación. En cada ronda se divide el bloque de 64 bits en cuatro bloques de 16 bits: X_1, X_2, X_3 y X_4, siendo combinados con las operaciones indicadas entre sí y con seis sub-bloques de 16 bits de la clave (subclaves). Entre ronda y ronda, se cambia de posición a los bloques 2 y 3. Finalmente, se combinan los cuatro sub-bloques con cuatro subclaves.

Los pasos de cada ronda son los siguientes:

1. *Multiplicar X_1 y la primera subclave.*
2. *Sumar X_2 y la segunda subclave.*
3. *Sumar X_3 y la tercera subclave.*
4. *Multiplicar X_4 y cuarta subclave.*
5. *Calcular el XOR de los pasos 1 y 3.*
6. *Calcular el XOR de los pasos 2 y 4.*
7. *Multiplicar los resultados del paso 5 y la quinta subclave.*
8. *Sumar los resultados de los pasos 6 y 7.*
9. *Multiplicar los resultados del paso 8 y la sexta subclave.*
10. *Sumar los resultados los pasos 7 y 9.*
11. *Calcular el XOR de los pasos 1 y 9.*

12. Calcular el XOR de los pasos 3 y 9.

13. Calcular el XOR de los pasos 2 y 10.

14. Calcular el XOR de los pasos 4 y 10.

La salida producida por la ronda son los cuatro sub-bloques resultado de los pasos 11, 12, 13 y 14. Se cambia el bloque 2 por el 3 (excepto en la última ronda), y ésta será la entrada de la siguiente ronda.

Tras la octava ronda, hay una transformación final de la salida:

1. Multiplicar X_1 y la primera subclave.
2. Sumar X_2 y la segunda subclave.
3. Sumar X_3 y la tercera subclave.
4. Multiplicar X_4 y cuarta subclave.

El algoritmo emplea 52 subclaves, que son creadas de la siguiente manera: se divide la clave de 128 bits en ocho subclaves de 16 bits. Éstas serán las ocho primeras subclaves del algoritmo (seis para la primera ronda, y dos para la segunda). Después, se rotan 25 bits de la clave hacia la izquierda y de nuevo se divide en ocho subclaves. Las cuatro primeras son para la segunda ronda; las otras cuatro para la tercera. Se realiza otra rotación de 25 bits a la izquierda, se vuelve a dividir en ocho subclaves... hasta el final del algoritmo.

El proceso de descifrado es prácticamente el mismo que el de encriptación, con la diferencia de que las 52 subclaves son las inversas de las empleadas en la encriptación respecto de la operación en la que fueron usados, además de utilizarse en el orden inverso.

SEGURIDAD DE IDEA

La longitud de la clave de IDEA es de 128 bits. Suponiendo que un ataque de fuerza bruta fuera el más eficiente, haría falta calcular 2^{128} (10^{38}) encriptaciones para encontrar la clave. Si se lograra diseñar un chip que probara mil millones de claves por segundo y se fabricaran mil millones de ellos, aún harían falta

10^{13} años - más que la edad del Universo. Con 10^{24} chips como éste se podría encontrar la clave en un día, pero no hay suficientes átomos de silicio en el Universo para construirlos.

En cuanto a los ataques basados en criptoanálisis, por ahora, sólo unos pocos estudios, han logrado ataques más eficientes que el de fuerza bruta (empleando 2^{42} operaciones), pero contra una implementación de IDEA con dos rondas. El IDEA normal, con ocho rondas, es por ahora seguro.

Por último, algunas claves débiles de IDEA, pero la posibilidad de generar aleatoriamente alguna de estas claves es de una entre 2^{96}. Y si incluso queremos eliminar esta posibilidad, basta con calcular el XOR de cada subclave y 0x0 da antes de emplearla.

EL ALGORITMO MD5

El algoritmo MD5 de message digests (Recopilación).

Se comienza suponiendo que se tiene un mensaje de b bits de longitud, escritos m0, m1,...m(b-1). El algoritmo tiene cinco pasos :

Paso 1: Adición de bits de relleno.

El mensaje es rellenado con n bits, de tal manera que le falte a su longitud 64 bits para ser un múltiplo de 512. De esos n bits, el primero es 1 y el resto son 0.

Paso 2 : Adición de la longitud.

La nueva longitud tras añadir los bits de relleno es almacenada en una representación de 64 bits y añadida al final del mensaje en forma de dos palabras de 32 bits, yendo en primer lugar la que contiene los bits menos significativos. Si la longitud del mensaje fuera mayor que 2^{64}, solamente se usan los 64 bits menos significativos.

De esta manera, la longitud del mensaje es ahora múltiplo de 512. $M_0...M_{n-1}$ denotan las palabras de 32 bits del mensaje.

Paso 3 : Inicialización de los bufferes.

Se usan cuatro bufferes, A,B,C y D, que son registros de 32 bits. Son inicializados a los siguientes valores :

A: 01 23 45 67.

B: 89 ab cd ef

C: fe dc ba 98

D: 76 54 32 10

Paso 4: Procesado del mensaje en bloques de 16 bits.

En primer lugar se definen cuatro funciones auxiliares que tienen como entrada tres palabras de 32 bits y como salida una palabra de 32 bits.

F(X,Y,Z) = (X AND Y) OR ((NOT(X)) AND Z)

G(X,Y,Z) = (X AND Z) OR (Y AND (NOT(Z))

H(X,Y,Z) = X XOR Y XOR Z

I(X,Y,Z) = Y XOR (X OR (NOT(Z)))

*En este paso se usa una tabla de 64 elementos T[1... 64] construida con la función seno, siendo Ti la parte entera de 4294967296 * abs(sen(i)) (i en radianes).*

El proceso es el siguiente:

```
/*Procesar cada bloque de 16 bits. */
For i = 0 to N/16-1 do
/*Copiar bloque i en X. */
For j = 0 to 15 do
Set X[j] to M[i*16+j].
end /* of loop on j */
/* Grabar A como AA, B como BB, C como CC, y D como DD. */
AA = A
BB = B
CC = C
DD = D
/* Primera etapa. */
```

/ [abcd k s i] denota la operación a = b + ((a + F(b,c,d) + X[k] + T[i]) << s). */*

/ Realizar estas 16 operaciones. */*

[ABCD 0 7 1] [DABC 1 12 2] [CDAB 2 17 3] [BCDA 3 22 4] [ABCD 4 7 5] [DABC 5 12 6] [CDAB 6 17 7] [BCDA 7 22 8] [ABCD 8 7 9] [DABC 9 12 10] [CDAB 10 17 11] [BCDA 11 22 12] [ABCD 12 7 13] [DABC 13 12 14] [CDAB 14 17 15] [BCDA 15 22 16]

*/*Segunda etapa. */*

/ [abcd k s i] denota la operación a = b + ((a + G(b,c,d) + X[k] + T[i]) << s). */*

/ Realizar estas 16 operaciones. */*

[ABCD 1 5 17] [DABC 6 9 18] [CDAB 11 14 19] [BCDA 0 20 20] [ABCD 5 5 21] [DABC 10 9 22] [CDAB 15 14 23] [BCDA 4 20 24] [ABCD 9 5 25] [DABC 14 9 26] [CDAB 3 14 27] [BCDA 8 20 28] [ABCD 13 5 29] [DABC 2 9 30] [CDAB 7 14 31] [BCDA 12 20 32]

*/*Tercera etapa. */*

/ [abcd k s t] denota la operación a = b + ((a + H(b,c,d) + X[k] + T[i]) << s). */*

/ Realizar estas 16 operaciones. */*

[ABCD 5 4 33] [DABC 8 11 34] [CDAB 11 16 35] [BCDA 14 23 36] [ABCD 1 4 37] [DABC 4 11 38] [CDAB 7 16 39] [BCDA 10 23 40] [ABCD 13 4 41] [DABC 0 11 42] [CDAB 3 16 43] [BCDA 6 23 44] [ABCD 9 4 45] [DABC 12 11 46] [CDAB 15 16 47] [BCDA 2 23 48]

*/*Cuarta etapa. */*

/ [abcd k s t] denota la operación a = b + ((a + I(b,c,d) + X[k] + T[i]) << s). */*

/ Realizar estas 16 operaciones. */*

[ABCD 0 6 49] [DABC 7 10 50] [CDAB 14 15 51] [BCDA 5 21 52] [ABCD 12 6 53] [DABC 3 10 54] [CDAB 10 15 55] [BCDA 1 21 56]

[ABCD 8 6 57] [DABC 15 10 58] [CDAB 6 15 59] [BCDA 13 21 60] [ABCD 4 6 61] [DABC 11 10 62] [CDAB 2 15 63] [BCDA 9 21 64]

/ Incrementar los registros*/*

A = A + AA

B = B + BB

C = C + CC

D = D + DD

end / del bucle i */*

Paso 5: Salida.

El message digest producido es A, B, C, D, empezando con los bits menos significativos de A y terminando con los más significativos de D. Independientemente de la longitud del mensaje, el digest será de 128 bits.

SEGURIDAD DE MD5

Berson trató de utilizar el criptoanálisis diferencial contra una sola ronda de MD5, pero su ataque no es efectivo contra las cuatro rondas. Hay un ataque más efectivo de den Boer y Bosselaers, en el que se producen colisiones en la función de compresión. En principio, esto no tendría un impacto práctico en la seguridad del hash, pero a fin de cuentas es una violación de los principios de diseño de MD5, por lo que convendría la cautela. Hay que tener en cuenta, por otro lado, que en el caso que nos ocupa, PGP, aunque se pudiera explotar esta debilidad, no afectaría a la seguridad de la encriptación, sino a la veracidad de las firmas digitales. En el difícil caso de que se lograra un ataque efectivo, bastaría recurrir a una versión superior de PGP o a gnuPG y emplear firmas DSA/SHA-1.

Con la aparición de Internet y la mayor importancia que se le va dando a la información día tras día, la tecnología que antes era utilizada sólo por la Militar o Gobiernos ha cobrado mayor importancia.

Seguridad de Protocolo Internet (IPSec)

IPSec, un protocolo que representa la tendencia a largo plazo hacia las redes seguras, es un conjunto de servicios basados en técnicas de cifrado y protocolos de seguridad. Como no requiere cambios en las aplicaciones o en los protocolos, IPSec se puede instalar fácilmente en las redes existentes.

IPSec proporciona autenticación en el nivel de equipo y cifrado de datos para conexiones VPN que utilicen el protocolo L2TP. IPSec negocia entre el equipo y el servidor de túnel remoto antes de establecer la conexión L2TP, lo que protege tanto las contraseñas como los datos.

L2TP utiliza protocolos de autenticación estándar basados en PPP, como EAP, MS-CHAP, SPAP y PAP con IPSec.

El cifrado está determinado por la asociación de seguridad IPSec o IPSec SA. Una asociación de seguridad es una combinación de una dirección de destino, un protocolo de seguridad y un valor de identificación único, denominado Índice de parámetros de seguridad (SPI). Los cifrados disponibles son:

- *Estándar de cifrado de datos (DES) con una clave de 56 bits, que se ha diseñado para uso internacional y cumple la legislación de cifrado para exportación de datos de EE.UU.*
- *Triple DES (3DES), que utiliza dos claves de 56 bits y se ha diseñado para entornos de alta seguridad de Norteamérica.*

Firmas Digitales ("Digital Signatures")

Una firma digital utiliza el mismo funcionamiento del "public key" o algoritmo asimétrico mencionado anteriormente. Como se mencionó, existe una "llave pública" y una "llave secreta", en el caso de firmas digitales la llave pública que es ampliamente conocida es capaz de identificar si la información proviene de una fuente fidedigna. En otras palabras, la llave pública será capaz de

reconocer si la información realmente proviene de la "llave secreta" en cuestión. Ejemplo:

El departamento de compras posee las llaves públicas de todos los empleados de la compañía, si llega un pedimento con la dirección de email del Director de Finanzas, Cómo puede asegurarse el departamento de compras que en realidad esta persona realizó el pedimento y no alguna otra que sobrepuso el email ?. La llave secreta del director de finanzas debe de encontrarse solo en su computadora, por lo tanto al enviar el email esta llave pública se añadió al email, y por lo tanto las llaves publicas determinarán si la llave secreta coincide con la del director.

Firmas Digitales en Internet

En el caso anterior de un Intranet, todas las llaves públicas provienen de una fuente fidedigna, esto es, las llaves "publicas" que posee el departamento de compras son autenticas ya que TODAS pertenecen sólo a empleados dentro de la compañía , la única posibilidad de fraude que existe, es si alguien trata de forjar la "llave secreta" de un empleado para hacerse pasar por otro.

Pero que sucede cuando este departamento de compras empiece a realizar transacciones en Internet?

Ellos anuncian su "llave publica" para todos los usuarios de Internet, y como solo ellos poseen la "llave secreta" sólo ellos podrán descifrar("desencriptar") la información. Pero ahora, surge la siguiente pregunta: Quién le garantiza a los usuarios de Internet que esta "llave publica" REALMENTE proviene de este departamento de compras ?

Para esto existen los certificados digitales que son emitidos por "agencias autorizadas" como Thawte o Verisign las cuales dan el VoBo("Visto Bueno") sobre la "llave publica".

Existen pocas compañías que realizan este servicio, pero debido a la naturaleza de las "llaves publicas" siempre debe de existir una agencia central que sea

capaz de decir "Si, esta llave publica proviene del departamento de compras" eso es todo su servicio, esto garantiza a los usuarios finales de "Internet" que la "llave publica" ha sido reconocida por una autoridad confiable Thawte o Verisign

La secuencia de eventos es la siguiente:

1. *Se adquiere un "Certificado Digital" de Thawte o Verisign (Costo aprox:$100-$350 Dlls U.S Anuales, variación depende de su uso)*
2. *Se coloca este certificado digital ("llave publica") en el servidor de paginas y se configura para que éste envíe información encriptada según sea necesario.*
3. *Cuando un usuario en Internet solicite información encriptada de nuestro sitio se envía esta "llave publica" para que pueda encriptar la información y enviarla de una manera segura al sitio.*
4. *Al recibir la "llave publica" el navegador ("Netscape" o "Explorer") del usuario final corrobora que en realidad esta "llave publica" proviene de quien dice, en este caso, la "llave publica" dice: "Soy la llave publica de osmosislatina.com y fui emitida por Verisign mi serie es:u7767DbXs4br342Dbnn6".*
5. *El navegador corrobora con Verisign (en este caso) y continua o avisa al usuario final el estado de la "llave publica".*
6. *NOTA: Si el navegador ("Netscape" o "Explorer") no corrobora la veracidad del "certificado digital" no implica que la información será enviada de manera insegura , la encriptación seguirá siendo valida. Lo que sucederá es que cuando sus visitantes entren a paginas que requieran encriptación (transacciones financieras) , el Navegador desplegara un mensaje a sus visitantes indicándoles que la fuente de encriptación ("llave publica") es insegura; esto se debe a que nadie puede avalar su "llave publica", de nuevo lo anterior no implica que la encriptación es invalida solo insegura.*

7. *En ocasiones lo anterior es suficiente para hacer desconfiar al usuario final o inclusive exponerse a que un tercero este generando esta "llave publica"*

Encriptación de 40-bits y 128-bits.

Existen varios niveles de encriptación, pero las combinaciones más comunes son 40-512 bits ("llave secreta--llave pública") y 128-1024 bits ("llave secreta--llave pública"). La versión 128-1024 bits es el tipo de encriptación más fuerte que existe en el mercado. Actualmente U.S.A prohibe la exportación de productos con este tipo de Tecnología, pero cabe mencionar que ya existen varios productos producidos en Europa con esta Tecnología que no poseen tales restricciones de exportación.

La gran mayoría de los sitios en Internet utilizan la encriptación 40-512 bits, la encriptación 128-1024 bits es utilizada generalmente en transacciones de alto riesgo, como las bancarias.

Es segura la encriptación que existe hoy en día?

Depende quien la intente observar!, Aunque la información sea enviada encriptada, cualquier persona en Internet con entrenamiento mínimo puede interceptar esta información encriptada, sin embargo, para observarla requiere de su "llave privada".

Y es aquí donde depende quien intente observar esta información, considere que una computadora personal (PC) puede realizar millones de operaciones por segundo, debido a esto, no es tan ilusorio generar una "llave privada" a partir de cierta información interceptada; las "llaves privadas" generalmente constan de 40-bits, en una PC es posible (aunque tardado) procesar estas 2^40 alternativas, ahora bien, si se tienen varios servidores en paralelo realizando

trillones de operaciones por segundo probablemente sea posible procesar estas 2^40 alternativas en cuestión de minutos.

Lo anterior es una de las razones por las que U.S.A cuida (cuidaba!) con tanto recelo la exportación de encriptación de 128-bits, la cual es 3 veces más poderosa (2^128 alternativas) que la de 40-bits.

Públicamente se conoce que en los servidores más poderosos del mercado es posible descubrir una "llave privada" en cuestión de días de procesamiento. Esto obviamente detiene aquellas personas ("hackers") con servidores "comunes" y en este caso hasta oficinas de seguridad gubernamentales en "desencriptar" información con este tipo de encriptación.

La importancia de los números primos

Una de las tareas que más tiempo ocupa a los grandes sistemas de ordenadores es el cálculo de números primos cada vez mayores. Su objetivo es poder obtener un número que sirva para cifrar mensajes y que luego sea muy complicado descifrarlos.

Vamos a ver cómo se podría cifrar un mensaje en función de un número primo. Cada letra en un mensaje tiene un número asociado que nunca varía. El número está establecido por el código denominado "American Standard Code for Information Interchange" (ASCII). El conjunto de caracteres ASCII define cada carácter con un número que va desde el 0 al 255. Por ejemplo, la letra "A" mayúscula tiene el código 65, la "z" minúscula tiene el código 122, etc. Cualquier texto escrito en un ordenador se puede trasladar a notación ASCII. Por ejemplo, en código ASCII la palabra "antivirus" es:

97 110 116 105 118 105 114 117 115

Así tenemos una cadena de números (que es como realmente se transmite la información digitalmente) que podríamos multiplicar por un número que sea la multiplicación de dos números primos. Si elegimos, por ejemplo, 14 (multiplicando 2 y 7), la cadena de números nos quedaría así:

1358 1540 1624 1470 1652 1470 1596 1638 1610

La persona que quiera leer lo que pone primero deberá averiguar cuál es el número que hemos utilizado para cifrar la información. Y para ello deberá adivinar cuáles son los dos factores que hemos utilizado para cifrar la información. Evidentemente, en este ejemplo es muy fácil, 14 es 7 por 2, no hace falta ninguna titulación en Matemáticas más allá de la obtenida cuando estábamos en primaria.

Sin embargo, si utilizamos números muy grandes, el problema se complica. Por ejemplo, si utilizamos el número 2.591.372.723, su descomposición en dos factores primos ya no es tan inmediata. A pesar de eso, en muy poco tiempo veríamos que es el producto de 97.453 y 26.591.

La longitud de estos números (lo que se llama el "tamaño de la clave") es primordial para que un cifrado sea más o menos efectivo. En el primer ejemplo, si pasamos a notación binaria el número 14 veríamos que se escribe 1110, un número de 4 bits. El segundo ejemplo, 2.591.372.723, se escribe en binario como 10011010011101010011010110110011, 32 bits. Y en los sistemas de cifrado actuales una clave de menos de 400 ó 500 bits se considera ridícula. Lo más normal es utilizar, como poco, ¡¡¡1.024 bits de longitud de clave!!!

Ventajas y problemas del cifrado

Mandar un correo electrónico cifrado aporta indudables ventajas tanto al emisor como al receptor del mensaje. La confidencialidad está prácticamente asegurada, nadie que no conozca las claves con las que se ha enviado el correo electrónico podrá enterarse de qué es lo que hay en el correo. Así podremos mandar todo tipo de información con la tranquilidad de que estará a salvo de teóricas intercesiones de la comunicación.

Pero... ¿qué es una intersección de la comunicación? En principio todos pensamos que es un hacker, o un espía de la competencia, o cualquier otro elemento de esa calaña que quiere inmiscuirse en las conversaciones ajenas. Entre todos los que quieren echar un vistazo hay uno que lo hace con muy buena fe y grandes dosis de ganas de defendernos... ¡el antivirus!

Un antivirus siempre va a intentar inmiscuirse en la comunicación por correo electrónico. Tiene que abrir el mensaje y mirar qué es lo que hay dentro, no con afán de espionaje sobre el contenido del mensaje, sino en busca de un virus. Si el mensaje lo hemos cifrado, la misión del antivirus puede fracasar.

En la estructura empresarial actual se tiende a instalar antivirus en los puntos de conexión de las redes a Internet: firewalls, proxys, etc. Es el sitio más lógico, ya que por allí van a entrar casi el 90% de los virus. Pero si el virus viene en un mensaje de correo electrónico cifrado, ese maravilloso antivirus que está en el firewall va a fracasar en su misión: al estar el contenido del mensaje cifrado, no va a poder detectar el virus que pueda contener el mensaje.

De otra manera, los usuarios que recibieran correos cifrados podrían verse envueltos, inconscientemente en una infección. Si el emisor ha cifrado el mensaje, también ha cifrado el virus. Nadie, ni siquiera el antivirus más potente podría descifrar el mensaje para su análisis. Simplemente buscaría un virus en un montón de caracteres y símbolos que al ser incomprensibles ocultarían al virus de una manera tan eficaz como ocultan el texto. Por ello, la protección en

las estaciones de trabajo debe ser tan fuerte como la perimetral, ya que es en la estación de trabajo donde el mensaje se descifrará y aparecerá el virus.

En definitiva, nadie duda que los sistemas de cifrado son una herramienta que aumenta la seguridad de las comunicaciones, pero tienen su reverso tenebroso: ocultan virus a los antivirus perimetrales que no estén preparados. La única solución para evitar que los virus cifrados entren en la empresa, debe ser una protección perimétrica efectiva que bloquee los elementos cifrados no autorizados antes de que puedan alcanzar los servidores y las estaciones de trabajo de la empresa.

4.3. MARCO CONCEPTUAL.

VPN: Estos tres modelos se incluyen en la amplia categoría de redes privadas virtuales (VPN, virtual Private Networking) aunque es verdad que cada modelo proporciona algún nivel de red privada, esta amplia definición es un poco confusa. Microsoft ha adoptado una definición más limitada del término y utiliza "VPN" para referirse a un servicio de seguridad a través de una infraestructura de red pública o sin confianza que incluye:

- *Acceso remoto seguro de cliente a puerta de enlace, bien a través de conexiones de Internet o bien dentro de redes privadas o subcontratadas.*
- *Conexiones seguras de puerta de enlace a puerta de enlace a través de Internet, o bien a través de redes privadas o subcontratadas.*

PPTP: *Se diseñó para proporcionar comunicaciones autenticadas y cifradas entre un cliente y una puerta de enlace o entre dos puertas de enlace (sin necesitar una infraestructura de clave pública) utilizando un Id. De usuario u una contraseña. Apareció por primera vez en 1996, dos años antes de la disponibilidad de IPSec y L"TP. El objetivo del diseño era la simplicidad, la compatibilidad multiprotocolo y la capacidad de cruzar una amplia gama de redes IP. E protocolo de túnel punto a punto (PPTP, Point-ti-point Tunnelign Protocolo) utiliza una conexión TCP para el mantenimiento del túnel y tramas PPP encapsuladas mediante encapsulación de enrutamiento genérico (GRE, Generic Routing Encapsulation) para los datos del túnel. Las cargas (partes de datos útiles) de las tramas PPP encapsuladas se pueden cifrar o comprimir. El uso de PPP proporciona la capacidad de negociar los servicios de autenticación, cifrado y asignación de dirección IP.*

SSL: *Qué es el SSL? SSL (Secure Sockets Layer) fue diseñado y propuesto en*

1994 por Netscape Comunications Corporation unto con su primera versión del Navihgator como un protocolo para dotar de seguridad a las sesiones de navegación a través de Internet. Sin embargo, no fue hasta su tercera versión, conocida como SSL v3.0 que alcanzó su madurez, superando los problemas de seguridad y las limitaciones de sus predecesores. En su estado actual proporciona los siguientes servicios:

Cifrado de datos: la información transferida, aunque caiga en manos de un atacante, será indescifrable, garantizando así la confidencialidad.

Autenticación de servidores: el usuario puede asegurarse de la identidad del servidor al que se conecta y al que posiblemente envíe información personal confidencial.

Integridad de mensajes: se impide que modificaciones intencionadas o accidentales en la información mientras viaja por Internet pasen inadvertidas. Opcionalmente, autenticación de cliente: permite al servidor conocer la identidad del usuario, con el fin de decidir su puede acceder a ciertas áreas protegidas.

Rl rasgo que distingue a SSL de otros protocolos para comunicaciones seguras, como el hoy prácticamente extinto S-http, es que se ubica en la pila OSI entre los niveles de transporte (TCP/IP) y de aplicación (donde se encuentran los conocidos protocolos http para Web, FTP para transferencia de ficheros, SMTP para correo electrónico, Telnet para conexión a máquinas remotas, etc.). gracias a esta característica, SSL resulta muy flexible, ya que puede servir para securizar potencialmente otros servicios además de http para Web, sin más que hacer pequeñas modificaciones en el programa que utilice el protocolo de transporte de datos TCP.

SSL proporciona sus servicios de seguridad sirviéndose de dos tecnologías de cifrado distintas: criptografía de clave pública (asimétrica) y criptografía de clave secreta (simétrica). Para el intercambio de los datos entre el servidor y el cliente, utiliza algoritmos de cifrado simétrico, que pueden elegirse típicamente entre DES, triple- DES, RC2, RC4 o IDEA. Para la autenticación y para el cifrado de la clave de sesión utilizada por los algoritmos anteriores, usa un algoritmo de cifrado de clave pública, típicamente el RSA.

Cómo funciona SSL?

Cuando un navegador solicita una página a un servidor seguro, ambos intercambian una serie de mensajes para negociar las mejoras de seguridad. Este protocolo sigue las siguientes fases (de manera muy resumida):

1. *La fase Hola, usada para ponerse de acuerdo sobre el conjunto de algoritmos para garantizar la confidencialidad e integridad y para la autenticación mutua. El navegador le informa al servidor de los algoritmos que posee disponibles. Normalmente, se utilizarán los más fuertes que se puedan acordar entre las dos partes. En función de las posibilidades criptográficas del navegador, el servidor elegirá un conjunto u otro de algoritmos con una cierta longitud de claves.*
2. *La fase de autenticación, en la que el servidor envía al navegador su certificado x.509v3 que contiene su clave pública y solicita a su ves al cliente su certificado x.509v3 (sólo si la aplicación exige la autenticación de su cliente).*
3. *La fase de producción de la clave de sesión en la que el cliente envía al servidor una clave maestra a partir de la cual se generará la clave de sesión para cifrar los datos intercambiados posteriormente mediante el algoritmo de cifrado simétrico acordado en la fase 1. el navegador envía cifrada esta clave maestra usando la clave pública del servidor que*

extrajo de su certificado en la fase 2. más adelante, ambos generarán idénticas claves de sesión a partir de la clave maestra generada por el navegador.

4. *la fase Fin, en la que se verifica mutuamente la autenticidad de las partes implicadas y que el canal seguro ha sido correctamente establecido. Una vez finalizada, ya se puede comenzar la sesión segura.*

De ahí en adelante, durante la sesión segura abierta, SSL proporciona un canal de comunicaciones seguro entre los servidores Web y los clientes (los navegadores) a través del cual se intercambiará cifrada la siguiente información:

El URL del documento solicitado.
Los contenidos del documento solicitado.
Los contenidos de cualquier formulario enviado desde el navegador
Las cookies enviadas desde el navegador al servidor y viceversa.
Los contenidos de las cabeceras http.

Limitaciones y problemas:

Debido a la limitación de exportación del gobierno de los Estados Unidos sobre los productos criptográficos, las versiones de los navegadores distribuidas legalmente más allá de sus fronteras operan con nada más de 40 bits de longitud de clave, frente a los 128 ó 156 bits de las versiones fuertes. Claves tan cortas facilitan los ataques de fuerza bruta o búsqueda exhaustiva de claves, pudiéndose descifrar mensajes cifrados en estas condiciones tan desfavorables en cuestión de horas o días, dependiendo de los recursos informáticos disponibles. Este serio problema ganó notoriedad en los medios

de comunicación cuando en 1995 un estudiante francés, Damien Doligesz, fue capaz de descifrar un mensaje cifrado con SSL en pocos días utilizando la red de ordenadores de su Universidad. Aún hoy, sin importar lo sofisticado y seguro que sea SSL v3.0, mientras se mantengan las restricciones de exportación y los usuarios españoles adquieran navegadores de claves cortas, la falta de seguridad está garantizadas.

Es importante recalcar que SSL sólo garantiza la confidencialidad e integridad de los datos de tránsito, ni antes ni después. Por lo tanto, si se envían datos personales al servidor, entre ellos el número de tarjeta de crédito, número de la seguridad social, DNI, etc., SSL solamente segura que mientras viajan desde el navegador hasta el servidor no serán modificados ni espiados. Lo que el servidor haga con ellos, está ya más allá de la competencia de este protocolo. Los datos podrían ser manipulados irresponsablemente o caer en manos de un atacante que asaltara el servidor. Por otro lado, debe tenerse muy en cuenta que SSL no garantiza la identidad del servidor al que se conecta el usuario. muy bien podría suceder que el servidor seguro contase con un certificada perfectamente válido y que estuviera suplantando la identidad de algún otro servidor seguro bien conocido, como el de Amazon. Por consiguiente, es de extrema importancia que se compruebe siempre el certificado del sitio Web para cerciorarse de que no se está conectando a un Web falsificado.
El servidor identifica el navegador incluso aunque éste no se autentique mediante certificados. Cuando un usuario se conecta a un servidor, rutinariamente le comunica ciertos datos como su dirección IP, tipo y versión de navegador y sistema operativo, y otros más. Con esta información es posible, según los casos, llegar directamente hasta el individuo, o al menos compañía que se conectó al servidor. El mero hecho de utilizar un canal cifrado con SSL no elimina este traspaso de información desde el navegador al servidor. Por

último, actualmente SSL solamente se utiliza para comunicaciones seguras en WWW, por lo que otros servicios de Internet, como el correo electrónico, no irán cifrados a pesar de utilizar SSL para el envío de formularios o la recuperación de páginas Web. Recuerde, debe usar S/MIME, PGP o algún otro software criptográfico para correo, ya que SSL no ofrece protección para sus mensajes.

Ventajas del SSL

SSL v3.0 goza de gran popularidad y se encuentra ampliamente extendido en Internet, ya que viene soportado por los dos principales navegadores del mercado, Netscape Navigator 3.0 ó superior así como por Internet Explorer 3.0 ó superior.

SSL proporciona un canal de comunicaciones seguro entre los servidores Web y los clientes (los navegadores), pero su uso no se limita a la transmisión de páginas Web. Al encontrarse entre los niveles de transporte y de aplicación, potencialmente SSL puede servir para securizar otros servicios, como FTP, correo, Telnet, etc.

El usuario no necesita realizar ninguna acción especial para invocar el protocolo SSL, basta con seguir un enlace o abrir una página cuya dirección empieza por https://. El navegador se encarga del resto.

5. METODOLOGIA

En la realización de un proyecto es importante definir los elementos o componentes metodologicos que describen o definen de una forma más directa la metodología utilizada en el proyecto.

5.1. AREA DE CONOCIMIENTO

El área de conocimiento del proyecto es la Ingeniería

5.2. AREA TEMATICA

El área temática del presente proyecto es la Ingeniería aplicada a la Seguridad en Redes Informáticas.

5.3. TEMA

El tema del proyecto es la Encriptación o Criptografía.

5.4. TITULO

El título del proyecto es Analisis y Evaluación de los Diferentes Metodos de la Seguridad de la Información aplicados a la Corporación Educativa Mayor del Desarrollo Simón Bolívar.

5.5. AREA DE INVESTIGACIÓN

La investigación en estudio esta inscrita en el área de sistematización y desarrollo institucional.

5.6. LINEA DE INVESTIGACIÓN

La línea de investigación a que se dirige el estudio es modelo y estrategia para la optimización de la seguridad en las redes informáticas para el desarrollo

académico de la Universidad.

5.7 METODO DE ESTUDIO

El método de estudio que se utilizó en la investigación es Analítico - Deductivo.

5.8 TIPO DE ESTUDIO

El estudio es de tipo exploratorio aplicado.

5.9 TECNICAS E INSTRUMENTOS

Con el fin de recopilar información veraz y precisa de una forma oportuna, se utilizó una serie de herramientas útiles y necesarias para este fin, dentro de éstas, podemos identificar dos tipos que son las técnicas e instrumentos que siguen.

5.9.1 Técnicas

Dentro de los caminos explorados, se destaca una serie de procedimientos de que se ha servido la investigación para enriquecerse y que han aportado seriedad, objetividad y claridad a la misma.

5.9.1.1 Fuentes primarias

Los datos más certeros y apropiados para definir los criterios de análisis de una investigación son los aportados por las personas involucradas en la entidad objeto del estudio, obtenidos a través de diferentes técnicas de recolección de información primaria.

5.9.1.2 Fuentes Secundarias

Además de los hechos anotados producto del escrutinio directo hecho en la sede del proyecto, existen otras normales que pueden ser usados como puntos de referencia general y específico, que resultan ser coadyacentes ideales en la labor de la investigación y sitúan el proyecto en sus fases preliminares e iniciales en un plano de certidumbre muy adecuado.

5.9.1.2.1 Textos

Dentro de la bibliografía consultada para ampliar los conocimientos y bases teóricas necesarias para el logro de los objetivos propuestos, se destacan las áreas de investigación y desarrollo de proyectos, y diversos libros sobre seguridad en redes.

5.9.2 Instrumentos

Con el objetivo de crear una atmósfera de seguridad y confianza alrededor de la investigación preliminar, fueron utilizados unos instrumentos útiles a este fin, los cuales sirven para situar a los gestores y al proyecto en sus momentos e instancias específicas, y para estimar la duración y el curso a seguir en el desarrollo del mismo.

5.9.2.1 Planeación

El cuadro de planificación detalla las actividades principales que se llevaron a cabo en todo el desarrollo del proyecto a cabo en todo el desarrollo del proyecto. Ver anexo No. 1

5.9.2.2 Cronograma

Se diseñó un cronograma con un tiempo estimado del desarrollo del proyecto. Ver Anexo No. 2

5.9.2.3 Presupuesto

En la tabla de presupuesto se especifican los gastos generales de papelería, transporte y recursos en general en que se incurrió en el desarrollo del anteproyecto. Ver anexo No. 3

1. ANALISIS DE REQUISITOS

En el análisis de requisitos utilizmos los diferentes Metodos de Seguridad de la Información con el fin de que la comunidad estudiantil se instruyan más acerca de su funcionamiento y las aplicaciones en que pueden ser utilizado eficientemente.

Los temas de Seguridad en red son los siguientes:

- *Cifrado Tradicionales (Cifrado por Sustitución, Cifrado por Transposición).*
- *Asimetrico (Des, Cifrado Publico)*
- *Simetrico (Cifrado Sencillo, Blowfish, Idea).*
- *Firmas Digitales (FIRMAS DE CLAVE SECRETA Y CLAVE PÚBLICA, RSA, DSA).*
- *Message Digests (MD5, SHA).*

2. INGENIERIA DE LA INFORMACION

2.1 HISTORIA DE LA UNIVERSIDAD SIMON BOLIVAR

Desde su llegada a la Universidad del Atlántico, el profesor JOSE CONSUEGRA HIGGINS, pone en practica su concepto sobre la Universidad y el papel que debe jugar en nuestra sociedad; "es una especie de antena receptora, analista responsable y faro erradicador de estrategias ideológicas. En ella debe acometerse el estudio consciente de la realidad social para dotar a los teóricos e ideológicos de buena parte del material que esta sirviendo y habrá de servir a la formulación de los preceptos liberadores de nuestros países. Este concepto de la Universidad que va más allá del tradicional que la concibe como simple superestructura al servicio del sistema, hace que desde el primer día de posesionarse del cargo de rector de esta Alma Mater, se le presente una fuerte oposición a la labor que desea realizar.

Así desde bien temprano se inicia la tarea de convertir a la Universidad en tribuna de denuncia, de estudio de los problemas de la colectividad y preparadora intelectual d el pueblo explotado.

La masificación del Alma Mater comienza con una campaña encaminada a abrir las puertas de esta a los hijos de los campesinos, los obreros y los empleados que nunca antes pudieron ingresar.
La población estudiantil al asumir Consuegra la rectoría estaba conformada por tres mil estudiantes y cuatro meses mas tarde ascendía a seis mil, aumentándose los cupos en dicho periodo en un cien por ciento. Para lograr esto se restablecieron inscripciones gratuitas, se redujo el valor de las matrículas y se eligió e instaló un comité de admisiones autónomo constituido por estudiantes y profesores.

Esta política de democratización y masificación fue complementada responsablemente por una dinámica conducta encaminada a lograr nuevos recursos fiscales nacionales, departamentales, contratar profesores, traer conferencistas nacionales y extranjeros, enviar profesores a hacer cursos de especialización, publicación de libros, etc.

El día 25 de Agosto la represión llegó a su máximo extremo cuando el gobernador Abelló Roca en un acto sin precedentes en la historia de la Universidad, violando sus estatutos, despreciando su relativa autonomía y desconociendo la voluntad del Consejo Superior (que había elegido al rector para un período de tres años, del cual apenas había cumplido unos ocho (8) meses) destituyó al rector de la Universidad.

Las razones expuestas para justificar este insólito hecho y que salieron publicadas en varios periódicos del país fueron: "El rector CONSUEGRA HIGGINS no estaba funcionando. Los dineros destinados a inversiones que no eran indispensables. Lo gastaban en editar libros, traer conferencistas y enviar profesores a dictar conferencias a universidades de Centro América y países del Sur del Continente. Igualmente se concedían títulos honorarios a catedráticos.

Las actuaciones y declaraciones del Gobernador fueron repudiadas por toda la prensa del país sin distingo de colores políticos. Además los intelectuales del país y del extranjero, las universidades públicas y privadas, las agrupaciones culturales, los sindicatos, etc., dieron a

Conocer su respaldo a CONSUEGRA HIGGINS por la labor realizada en bien de la Universidad.

También es digno mencionar que el acto represivo de destitución fue

complementado con otro más reaccionario que consistió en nombrar como nuevo rector a GULLERMO RODRIGUEZ FIGUEROA, pero a este Sr. El día de Agosto de la juventud estudiosa consciente de nuestra Institución, respaldada de sus profesores y trabajadores le impidió la entrada al recinto Universitario.
Ante este rechazo al nuevo rector, al primer mandatario del departamento ordenó la invasión a la Universidad por parte de las fuerzas combinadas del ejército y la policía. Fue en esta forma como pudo ingresar el "Policía Figueroa" al Alma Mater. Igualmente, a partir de su ingreso se desató la más grande represión que se haya dado en la vida de EST Casa de Estudios contra profesores, estudiantes y trabajadores. Se clausuraron semestres, se expulso masivamente a profesores y estudiantes, se aumentaron las matriculas, se disolvió el comité de admisiones, etc., o sea que de un día para otro se acabo con la gran labor realizada por CONSUEGRA HIGGINS
En los ocho meses que estuvo al frente de la Universidad del Atlántico.
Iguai que el estudiantado del resto del país, durante todo el año de 1.971 y primer semestre de 1.972, el estudiantado de la Universidad del Atlántico había demostrado su gran capacidad de combate luchando por la solución a problemas internos. Pero el segundo semestre del presente año, la represión del gobierno ya llegaba al límite máximo hasta el punto de convertir a varias Universidades del país en verdaderos cuarteles de policía. Es así como el estudiantado y profesorado consecuente es expulsado de esta Institución y vetado su ingreso a otras universidades oficiales. Lo anterior conduce a un grupo de catedráticos y directivos reprimidos a fundar una
Verdadera CASA DE ESTUDIOS SUPERIORES a la cual ingresaran aquellos estudiantes y profesores ultrajados por RODRIGUEZ FIGUEROA.

Los objetivos de este grupo de Catedráticos e intelectuales son: "Formar una universidad Latinoamericana completamente diferente a la actual que sigue respondiendo a esquemas obsoletos, alejados de las exigencias actuales de nuestros pueblos. Más que simples abogados, economistas y sociólogos

aspiramos a formas profesionales con respaldo cultural e ideológico. Hombres en condiciones de responder a las exigencias del país y con capacidad para estudiar y comprender sus problemas.

En la misma entrevista y sobre estos objetivos al profesor CONSUEGRA HIGGINS comenta algo más: "La superación de la situación de atraso y dependencia es la meta anhelada de nuestro pueblo, nuestro propósito es facilitar las condiciones para que nuestros estudiantes se preparen de tal manera que puedan servir, en sus diferentes áreas a una situación de cambio en cualquier momento que se le exija o las circunstancias lo permitan. La metodología para ello es cambiar la enseñanza de manual y de cartillas por la investigación y el compromiso con la realidad nacional. Queremos un estudiante que participe activamente en clase, respaldado por la lectura intensa de los libros dados como bibliografía.

2.2 misión

LA CORPORACION MAYOR DEL DESARROLLO SIMON BOLIVAR es una Casa de Estudios Superiores del pueblo, para la investigación científica, la formación técnica y la promoción cultural e ideológica.

Sin animo de lucro, no oficial, dedicada al servicio de la profundización del proceso de formación personal y profesional con una concepción integral que permite el desarrollo de las facultades humanas, orientándolas al servicio de la cultura regional, nacional y latinoamericana y a la producción del conocimiento científico, teniendo como fundamento el ideario bolivariano de un ser humano autónomo, ético y culto, y una sociedad libre, justa y solidaria.

Para cumplir su función social de DOCENCIA, Investigación y Extensión. La CORPORACION EDUCATIVA MAYOR DEL DESARROLLO SIMON BOLIVAR se

caracteriza por la actualización y universalización de los saberes, fundamentada en los aportes que las Ciencias Sociales, Naturales y Exactas brindan para la comprensión total de la realidad. Realidad que ha sido fragmentada para aproximarnos a su compleja expresión; y la flexibilidad del curriculum que tendrá como norte la creación de una Teoría Social - Económica para el Desarrollo Latinoamericano en consonancia con el entorno y la gestión oportuna, eficaz y eficiente de los procesos administrativos y de los recursos para el logro de los propósitos institucionales, de tal manera que la comunidad educativa pueda cumplir su papel de constructora de la sociedad proyectada en esta misión.

La Corporación cultiva el ideario de EL LIBERTADOR en lo relacionado con la valoración l ancestro y la cultura propia y la defensa de la unidad regional, nacional y Latinoamérica.

2.3 Visión

La CORPORACION EDUCATIVA MAYOR DE DESARROLLO SIMON BOLIVAR es una comunidad universitaria científica que se empeña en crear, reproducir y difundir el conocimiento en favor de una sociedad desarrollada, autónoma, justa y solidaria.

La Corporación pretende incorporarse al futuro como una institución que forma líderes y dirigentes con conciencia nacional y latinoamericana, con responsabilidad ética, identificados con el compromiso histórico del enriquecimiento espiritual e intelectual de la sociedad y el fortalecimiento de la identidad regional, nacional y latinoamericana en la conquista del sueño bolivariano de una América unida y solidaria.

En tal sentido, la CORPORACION EDUCATIVA MAYOR DEL DESARROLLO

SIMON BOLIVAR se constituirá en Factor de Desarrollo Humano Local, Regional y Nacional mediante la construcción de un Modelo de Desarrollo Social que se fundamente en los Principios de la pluralidad de las Culturas y la Participación Democrática de sus Actores.

Con propósitos definidos en la utilización de sus recursos en la formación humanística de sus estudiantes, se esmera en crear y mantener Bibliotecas y Museos que sirven de símbolos de su responsabilidad ene l fomento de la cultura y la formación de profesionales capaces de responder a las exigencias del desarrollo.

2.4 Propósito

2.4.1 *Formular una teoría económica y social que pueda interpretar los fenómenos propios del subdesarrollo y ofrecer estrategias adecuadas para su superación.*

2.4.2 *Realizar investigaciones de carácter socioeconómico, político, jurídico y cultural de la localidad de nuestra sociedad y proponer soluciones y estrategias de desarrollo, que conduzcan al establecimiento de una senda armónica local, regional y nacional.*

2.4.3 *Facilitar experiencias de aprendizaje que le permitan a cada estudiante acceder reflexiva, critica y creativamente nuestra herencia cultural diversa y compleja que de paso habilite para la creación, desarrollo y transmisión de conocimientos que le capacite para cumplir son sus funciones profesionales, investigativas y de servicio social que requieren la región y el país.*

2.4.4 *Proporcionar condiciones democráticas que le facilite a la comunidad*

educativa desarrollar sus capacidades autónomas para emitir juicios respetables y respetuosos, ante las diferentes comunidades a las que pertenece, y frente al principio de autoridad como elemento rector de vida.

2.4.5 *Promover ambientes pedagógicos que favorezcan el desarrollo de la capacidad de comprensión, de discernimiento y de juicio en el educando.*

2.4.6 *Favorecer relaciones sociales éticas que permitan construir colectivamente los valores de la convivencia pacifica, promover la unidad, descentralización y actuar armónicamente entre sí y con las demás estructuras educativas.*

2.5 Himno

CORO

Simón Bolívar, ciencia y Libertad
Simón Bolívar, tu Universidad

I

La experiencia que cubre mis años,
Es un germen de aurora boreal;
Soy el surco feraz que germina
En las luchas que debo librar

II

Soy la llama procera que ofrece
En los claustros radiante el saber,
A este mundo colmado de bienes
Repleto de amor, y de paz y de fe

III

Soy la madre genero esperanza
Soy cultura, ciencia y libertad
Es mi afán extinguir la ignorancia
Soy el pueblo y traigo la paz

IV

Tras las metas gloriosas del arte,

Del deporte y la ciencia social;
Nuestras almas conducen la antorcha
Que despide su lumbre, ¡OH luz! inmortal.

2.6 Descripción De La Universidad

LA CORPORACION EDUCATIVA MAYOR DE DESARROLLO SIMON BOLIVAR esta situada en varias Sedes:

SEDE CRA 54: CRA 54 CALLE 59 ESQUINA

SEDE CRA 59: CRA 59 59-76

SEDE CRA 59: CRA 59 59-92

SEDE DE POTGRADOS: CRA 54 64-223

TELEFONO DEL PBX: 3 444 333

2.7 Programas

PSICOLOGIA

FISIOTERAPIA

TRABAJO SOCIAL

BASICA PRIMARIA

CIENCIAS SOCIALES

DERECHO

INGENIERIA DE SISTEMAS

INGENIERIA COMERCIAL

ECONOMIA

SOCIOLOGIA

ADMINISTRACION DE EMRPESAS

CONTADURIA PÚBLICA

2.8 Dependencias

RECTORIA

SINDICATURA

CONTABILIDAD

DECANATURAS

BIBLIOTECA

BIENESTAR

CREDITO Y COBRANZAS

ADMISIONES Y MATRICULAS

DIRECCION INFORMATICA

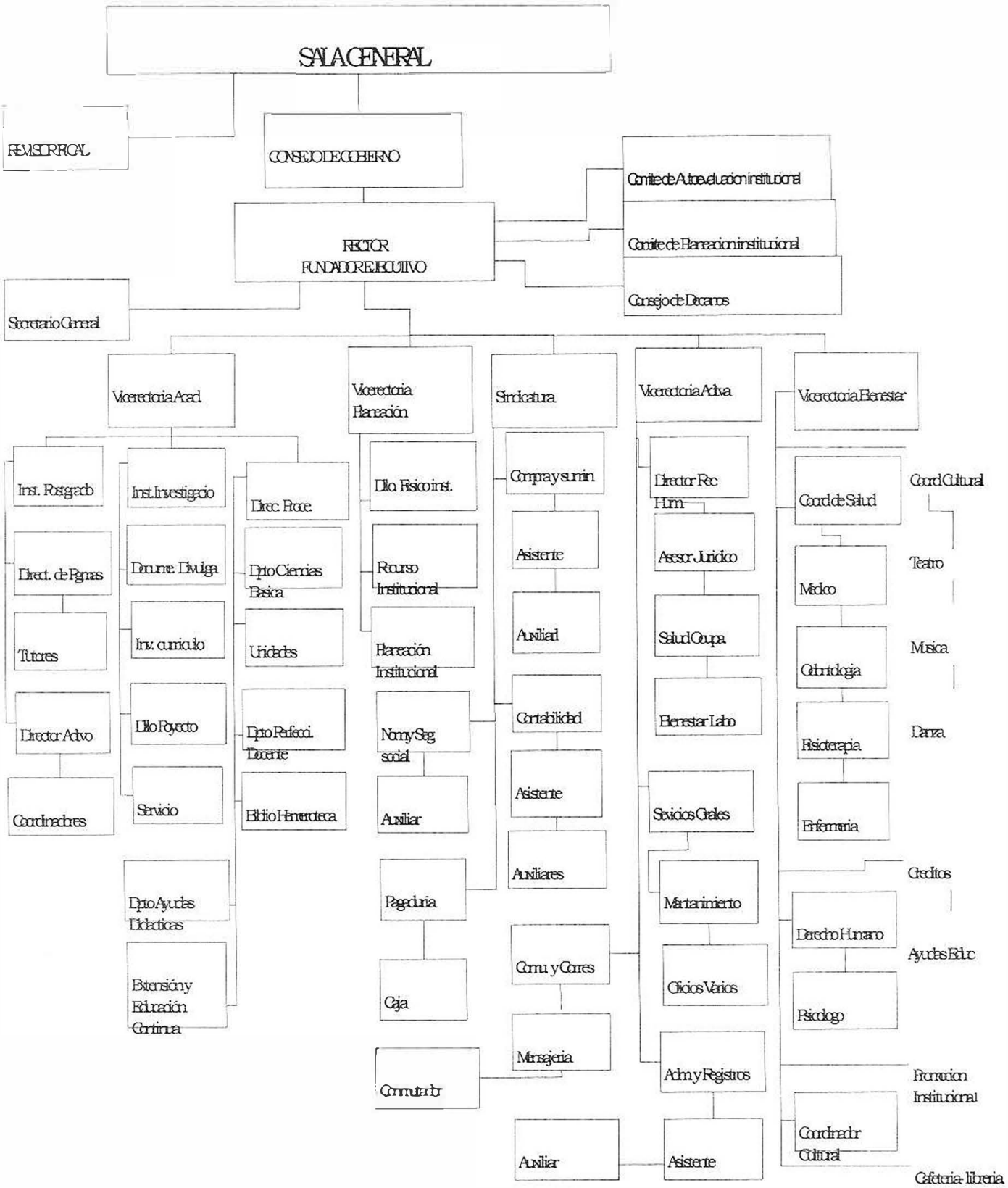
CENTRO DE COMPUTOS

2.9 Organigrama

La estructura de la Universidad Simón Bolívar, tiene gran diversidad de jerarquías, entre estas se encuentran la rama Administrativa, Académica, Ejecutiva, las cuales se muestran en la gráfica 3.

También, existen diversidad de departamentos los cuales poseen sus propias instalaciones y

Personal de trabajo. En el Anexo XIX se muestran detalladamente los organigramas utilizados en esta empresa


SALA GENERAL
REVISOR FISCAL
CONSEJO DE GOBIERNO
Comite de Autoevaluacion institucional
Comite de Planeacion institucional
RECTOR
FUNDADOR EJECUTIVO
Consejo de Decanos
Secretario General
Vicerrectoria Acad.
Vicerrectoria Planeación
Sindicatura
Vicerrectoria Adtva
Vicerrectoria Bienestar
Inst. Postgrado
Inst. Investigacio
Direc. Proye.
Dllo. Fisico inst.
Compra y suminis
Director Rec Hum
Coord de Salud
Coord Cultural
Direct. de Pgmas
Docume. Divulga
Dpto Ciencias Basica
Recurso Institucional
Asistente
Asesor Juridico
Medico
Teatro
Tutores
Inv. curriculo
Unidades
Planeación Institucional
Auxiliar
Salud Ocupa.
Odontologia
Musica
Director Adtvo
Dllo Proyecto
Dpto Perfecci. Docente
Norm y Stgs social
Contabilidad
Bienestar Labo
Fisioterapia
Danza
Coordinadores
Servicio
Biblio Hemeroteca
Auxiliar
Asistente
Sevicios Grales
Enfermeria
Dpto Ayudas Didacticas
Pagaduria
Auxiliares
Mantenimiento
Creditos
Extensión y Educación Continua
Caja
Comu y Cores
Oficios Varios
Derecho Humano
Ayudas Educ
Psicologo
Conmutador
Mensajeria
Adm y Registros
Promocion Institucional
Auxiliar
Asistente
Coordinador Cultural
Cafeteria-libreria

3.0 Organización administrativa

SALA GENERAL

ANA BOLIVAR DE CONSUEGRA PRESIDENTA

MANUEL FIGUEROA RUIZ VICEPRESIDENTE

JOSE CONSUEGRA HIGGINS
LEONELLO MARTHE ZAPATA
ALVARO CASTRO SOCARRAS
EUGENIO BOLIVAR ROMERO
JOSE IGNACIO CONSUEGRA MANZANO

RAFAEL BOLAÑO MOVILLA SECRETARIO

JOSE CONSUEGRA HIGGINS RECTOR FUNDADOR

JOSE CONSUEGRA BOLIVAR RECTOR EJECUTIVO

RAFAEL BOLAÑO MOVILLA SECRETARIA GENERAL

ANA EMILIA DE BAYUELO SINDICATURA

ISRAEL ARTETA ARTETA REVISOR FISCAL

4.0 Funciones

4.1 Créditos y cobranzas

4.1.1 Propósito

Asignar créditos a los estudiantes para su matrícula y llevar un control del crédito durante el semestre estableciendo cual es la situación financiera del estudiante, en relación con la Universidad.

4.1.2 Funciones

1. *Entregar los formularios de crédito a los estudiantes*
2. *Recibir los documentos de crédito para aprobarlos*
3. *Asignar crédito a los estudiantes*
4. *Enviar carta de cobro a los codeudores que respaldan el crédito*
5. *Listar los estudiantes morosos y enviarlos a los Decanaturas*
6. *Liquidar a los estudiantes que han realizado abonos de las cuotas de créditos internos.*
7. *Informar sobre el saldo del crédito a los estudiantes.*
8. *Imprimir los volantes de pago de estudiantes con créditos internos*
9. *Imprimir los volantes de las tres cuotas con sus respectivas fechas de pago.*

4.1.3 Asistente De Control y Registro De Matricula

4.1.3.1 Propósito

Llevar el archivo de los estudiantes activos.

4.1.3.2 Funciones

1. Archivar la documentación.

2. Llevar el archivo cronológico de cada estudiante.

3. Registrar la firma de cada estudiante.

4.1.4 Auxiliares De Control y Registro De Matriculas

4.1.4.1 Propósito

Atender a los estudiantes y recibir la documentación para su matrícula.

4.1.4.2 Funciones

1. Atención al público.

2. Recibir la documentación.

3. Registrar las inscripciones y matrículas.

4. Archivar.

5. Listado de estudiantes por Facultad.

4.1.5 Jefe De Control y Registro De Matriculas

4.1.5.1 Propósito

La misión del cargo es supervisar, coordinar y agilizar el funcionamiento de las matrículas.

4.1.5.2 Funciones

1. Matricular a los estudiantes que llenan los requisitos exigidos para tal fin.

2. *Recibir la documentación de los estudiantes nuevos.*
3. *Llevar el archivo de fólderes de los estudiantes matriculados en todas las unidades académicas.*
4. *Revisar la documentación de los estudiantes egresados necesaria para optar al título.*
5. *Llevar las listas de los estudiantes de cada facultad y remitirlas a los profesores.*
6. *Resolver los problemas relacionados con Registro y matrículas.*
7. *Las demás que le asignen los estatutos, reglamento y el Rector.*

4.1.6 Mensajero

4.1.6.1 Propósito

Efectuar las diligencias asignadas por su jefe inmediato y entregar la correspondencia entre las diferentes dependencias, de acuerdo con los procedimientos establecidos, a fin de contribuir a que el proceso de comunicación a nivel interno y externo sea ágil y oportuno.

4.1.6.2 Funciones

1. *Transportar y distribuir la correspondencia, y documentos a las diferentes dependencias de la universidad u otros lugares requeridos.*
2. *Recibir y consignar el dinero recaudado en la Universidad.*
3. *Entregar el volante de consignación al Cajero y llenar la correspondiente planilla.*
4. *Realizar las diligencias en bancos, corporaciones y demás entidades financieras, de acuerdo con las instrucciones de su jefe inmediato, correspondientes a los movimientos financieros.*
5. *Enviar la correspondencia entregada por la Jefe de Archivo y Correspondencia.*
6. *Las demás que le asignen el reglamento y su Jefe Inmediato.*

4.1.7 Caja

4.1.7.1 Propósito

Prestarle un mejor servicio a la comunidad universitaria facilitándole el recaudo.

4.1.7.2 Funciones

1. *Recibo efectivos por los siguientes conceptos certificados, constancias, duplicados, formularios, duplicados de paz y salvo, duplicados de carné, duplicados de consignaciones, seminarios.*
2. *Recibir tarjetas de crédito por los siguientes conceptos matrículas, módulos, diplomados, derecho de grado, cursos vacacional.*
3. *Recibir órdenes de las diferentes cajas de compensación y cooperativas.*

4.1.8 Auxiliar Del Departamento De Nomina

4.1.8.1 Propósito

- *Elaboración de nominas y liquidación de ISS*
- *Registro de préstamos a empleados en los libros auxiliares.*

4.1.8.2 Funciones

1. *Registrar préstamos a empleados en los libros auxiliares.*
2. *Elaboración de nominas, correspondencia.*
3. *Registrar los cambios del ISS- de cada empleado*
4. *Registrar los cambios de las diferentes EPS. y fondos de pensiones.*
5. *Impresión y entrega de volantes sobre salarios devengados mensual y quincenal a cada empleado.*

6. *Archivo de nominas en sus respectivos fólderes.*
7. *Elaboración de certificados de retención en la Fuente*
8. *Atender al público y tomar informaciones para luego comunicar las decisiones*
9. *Recibir llamadas e informar de sobre los casos de inmediata solución.*

4.1.9 Contador

4.1.9.1 Propósito

La misión del área de contabilidad es llevar los registros contables de todas las transacciones económicas que realiza la corporación para cumplir fielmente lo establecido dentro de la ley y registrar una contabilidad transparente que permita la toma de decisiones de la Dirección.

4.1.9.2 Funciones

1. *Elaborar los estados financieros de la Corporación.*
2. *Revisar los ingresos de los estudiantes de postgrado.*
3. *Revisar libros mayores y auxiliares.*
4. *Realizar las conciliaciones bancarias.*
5. *Liquidar los aportes de la Caja de Compensación Familiar.*
6. *Mantener informada a la Sala General sobre todos los aspectos contables y presentar periódicamente los estados financieros.*
7. *Registros en los libros Diarios y Mayores Y Balances.*
8. *Elaboración de asientos contables.*
9. *Liquidación de Retención en la Fuente.*
10. *Informes para el Dane.*
11. *Facturación*
12. *Elaboración de recibos de cajas y consignaciones.*
13. *Registros de asientos contables en el sistema.*
14. *Asistir a reuniones programadas por Icetex, relacionadas con los créditos*

fondo Simón Bolívar - Icetex.

15. *Revisión de los libros. del Colegio de Bachillerato de Isabel López.*
16. *Las demás que se le asignen los estatutos, Sindicatura y el Rector.*

4.1.10 Asistente De Contabilidad

4.1.10.1 Propósito

Revisar que los registros contables de todas las transacciones económicas realizadas por la corporación estén registradas en las respectivas cuentas

4.1.10.2 Funciones

1. *Buscar informaciones sobre, pagos de facturas, pagos empleados, etc.*
2. *Revisión de información que sale del sistema si fue suministrada correctamente.*
3. *Efectuar la interfase de las nominas de empleados, para el programa contable*
4. *Registrar al sistema comprobantes internos*
5. *Archivar.*

4.1.11 Auxiliar De Contabilidad

4.1.11.1 Propósito

Registró contable de todas las transacciones económicas que efectúa la corporación.

4.1.11.2 Funciones

1. *Revisar las notas crédito y débito que nos envían diferentes bancos.*
2. *Registro las notas crédito y débito en su respectivo libro de banco.*
3. *Conciliar los diferentes extractos bancarios*
4. *Registro de las matricula manual (recibos amarillos).*
5. *Elaborar comprobante de ingreso por cada banco.*
6. *Archivar consignaciones por los diferentes bancos.*
7. *Asientos internos*
8. *Conciliación Bancaria*
9. *Contabilizar facturas*
10. *Elaboración comprobante de egreso y cheques.*

4.1.12 Jefa Del Departamento De Nomina y Seguridad Social

4.1.13 Propósito

Confección y liquidación en aspectos nominales, prestacionales e informes generales.

4.1.14 Funciones

1. *Supervisión y revisión de nóminas.*
2. *Efectuar los recibos ejecutados en la nómina*
3. *Atender las informaciones solicitadas por jefatura de personal*
4. *Darle curso y liquidar todo tipo de memorando sobre novedades en nomina*
5. *Realizar todas las liquidaciones prestaciones a que hubiere lugar.*
6. *Realizar visitas a entidades tales como juzgado, Comfamiliar, fondos de pensiones, I-S-S- etc.*

7. *Atender y revisar la documentación que el personal entrega para obtener liquidaciones parciales e informales sobre los detalles de los mismos.*
8. *Vaciar las informaciones emanadas de las distintas Decanaturas sobre las horas dictadas y deducir las faltas, incapacidades, permisos que se presten.*
9. *Elaborar los certificados de ingresos y retenciones del año inmediatamente anterior a todo el personal de la Universidad.*
10. *Liquidar intereses de cesantías anuales al personal de término indefinido.*
11. *Atender e informar al personal que lo requiera, sobre reclamos que se originen por liquidaciones, afiliaciones cambios, etc.*
12. *Informar sobre la apertura de cuentas de los nuevos empleados a las entidades crediticias.*

4.1.15 Sindico

4.1.16 Propósito

La misión del cargo del sindico asumir la dirección de los aspectos económicos y contables, teniendo bajo su responsabilidad las dependencias de Compra y Suministro, de Tesorería, Contabilidad; Pagaduría, Nomina y Seguridad social.

4.1.17 Funciones

1. *Atender lo relacionado con el cobro de cuotas y cuentas que se adeuden a la Corporación y recibir toda clase de bienes, valores que se deben ingresar al patrimonio de la misma, expedir los correspondientes recibos.*
2. *Dirigir la oficina de Contabilidad con la debida claridad y corrección de acuerdo con las leyes Colombianas, así como la ejecución del presupuesto*
3. *Presentar a la Sala General, semestralmente, o antes, parcialmente, si lo*

considera conveniente, el balance de las cuentas.

4. *Cubrir las cuentas autorizadas por la Sala General y el Rector, y revisadas por el Revisor Fiscal*
5. *El Síndico asistirá con derecho a voz pero sin voto, a las reuniones de la Sala General y del Consejo de Gobierno, cuando se le cite a este último.*
6. *Custodiar los bienes de la Corporación.*
7. *Elaborar, con el Rector, los Vice Rectores Administrativos y de - - Planeación y el Revisor Fiscal el proyecto de Presupuesto.*
8. *Firmar con el respaldo o visto bueno del Revisor fiscal los Cheques, giros, operaciones bancarias y demás gestiones financieras que realice la Corporación.*
9. *Las demás que se le asignen los estatutos, reglamentos y el Rector*

6.0. AMBITO ESPACIAL

6.1 POBLACION REFERENCIA

La investigación en estudio se referencia en el área circunscrita por la ciudad de Barranquilla

6.2 POBLACION AFECTADA

La población afectada está compuesta por las diferentes organizaciones dedicadas al manejo de Redes informáticas.

6.3 POBLACION OBJETIVO

La población objetivo la conforma la Corporación Educativa Mayor de Desarrollo Universidad Simón Bolívar.

7.0. DELIMITACIÓN DEL PROYECTO

7.1. DELIMITACIÓN FINANCIERA

Para la investigación preliminar y el subsecuente desarrollo del proyecto, se incurrió en gastos varios, que serán asumidos así:

Los gestores del proyecto asumirán los gastos relativos con la investigación preliminar, consistentes en transportes, papelería, y demás desembolsos que sean necesarios.

7.2. DELIMITACIÓN TEMPORAL

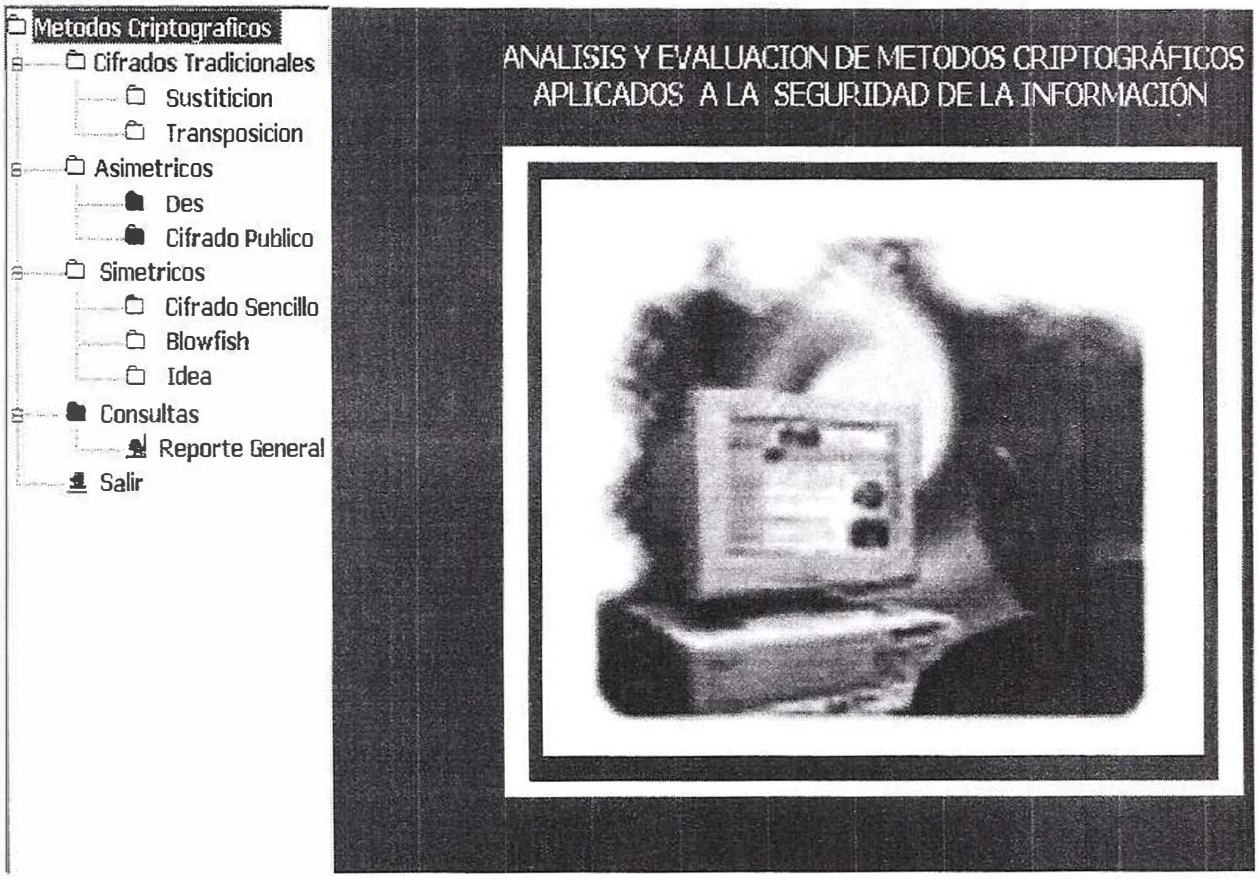
El tiempo de duración del proyecto abarca desde la primera semana de julio del año 2001 hasta la última semana de septiembre del año 2003, contando con la etapa de aprobación del proyecto.

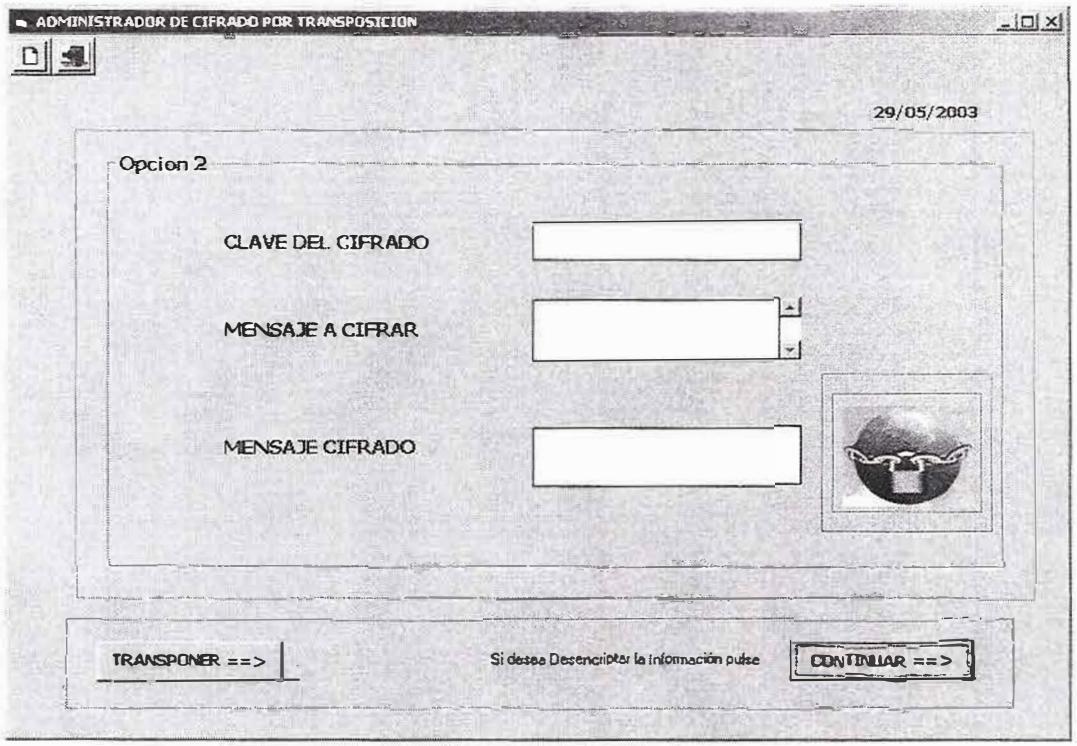
7.3. DELIMITACIÓN ESPACIAL

La investigación correspondiente al presente proyecto se realizará en la ciudad de Barranquilla, que se encuentra situada en el departamento del Atlántico, específicamente en el centro de cómputo de la Corporación Educativa Mayor de Desarrollo Universidad Simón Bolívar.

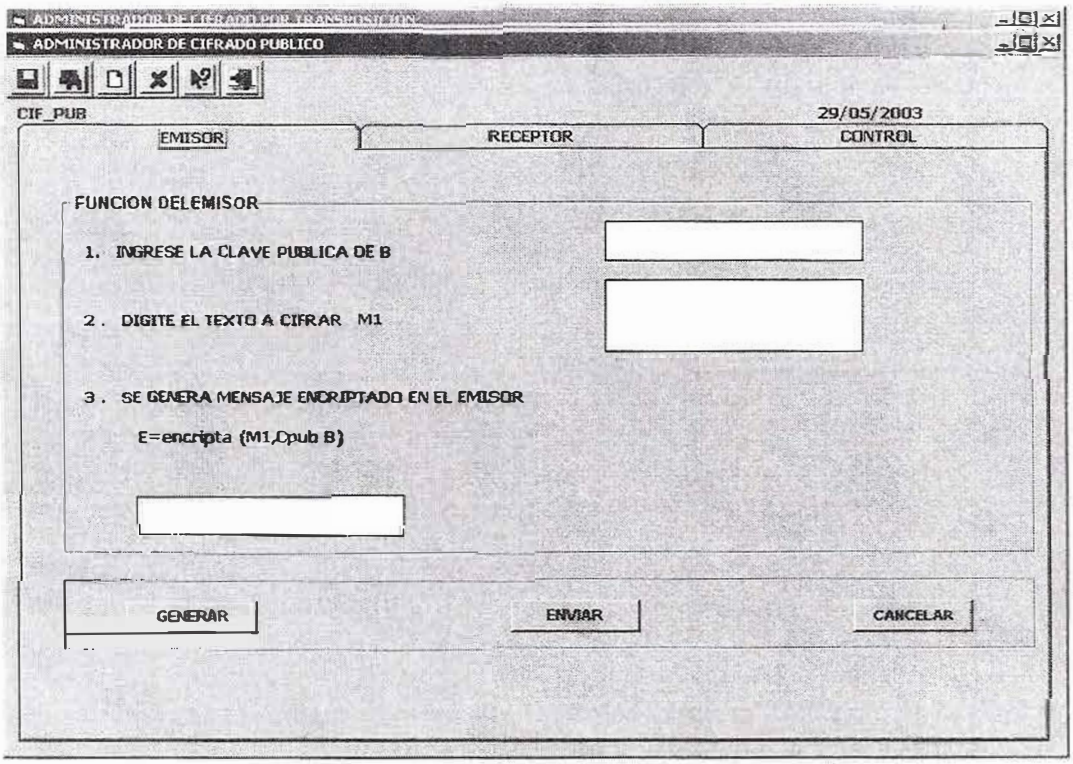
ANEXOS

CORPORACIÓN EDUCATIVA MAYOR DEL DESARROLLO
UNIVERSIDAD SIMÓN BOLÍVAR



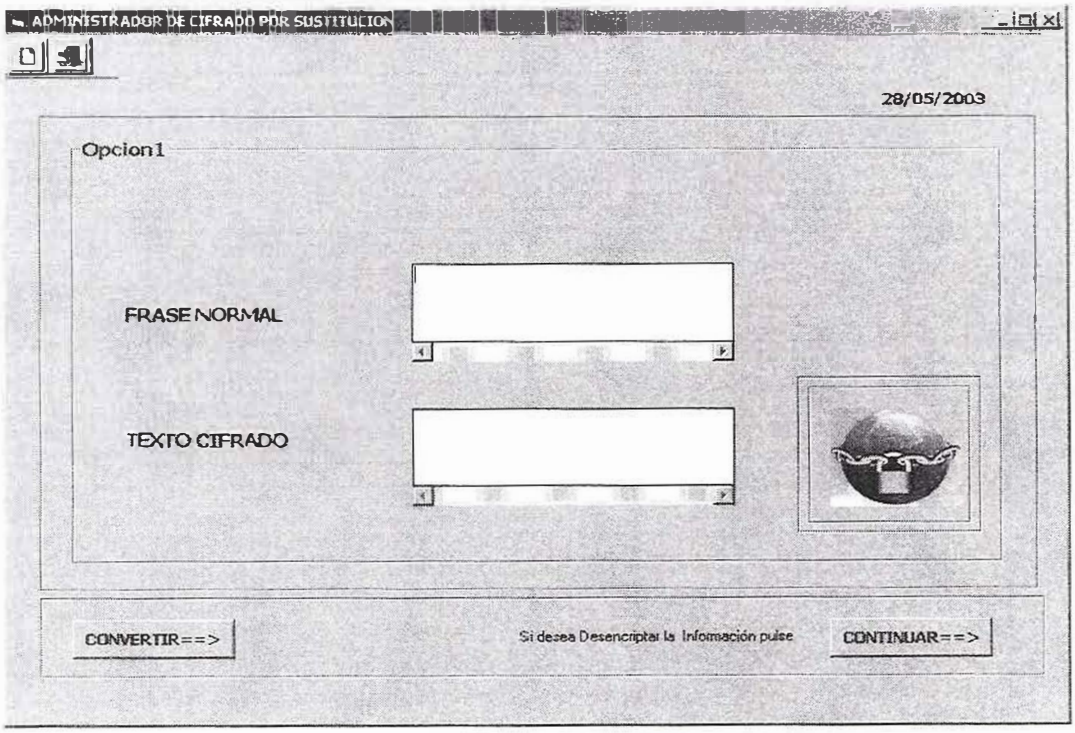


CONTROL DE ENCRIPCIÒN DE TRANSPOSICIÒN

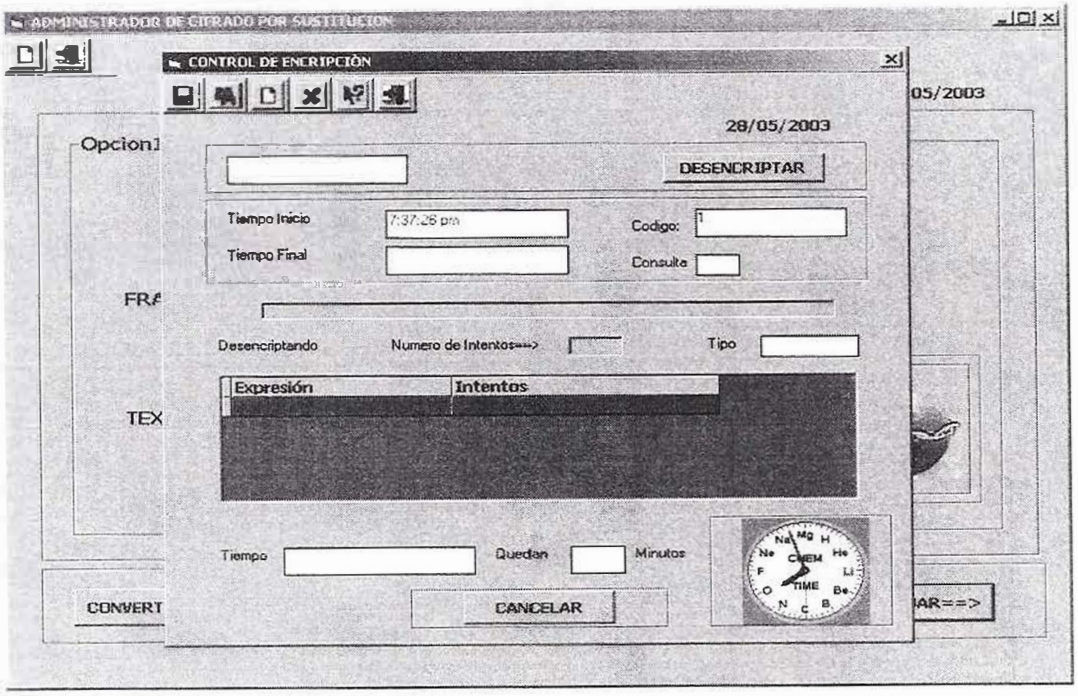


ADMINISTRADOR DE CIFRADO PUBLICO(EMISOR)

ADMINISTRADOR DE CIFRADO POR SUSTITUCIÒN

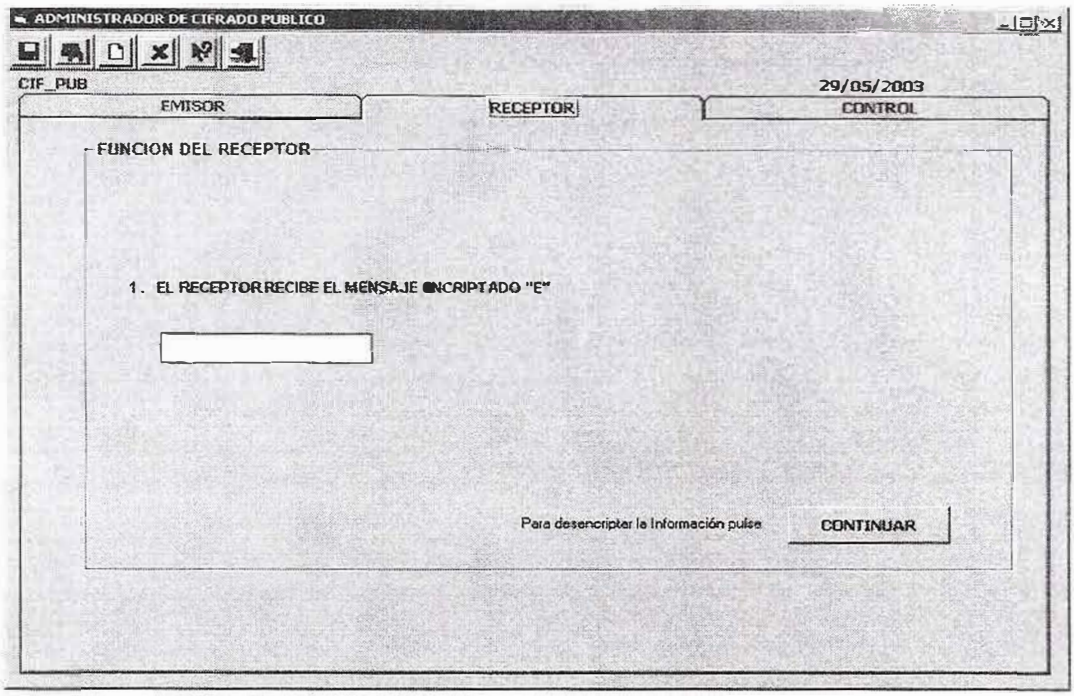


CONTROL DE ENCRIPCIÒN DE SUSTITUCIÒN

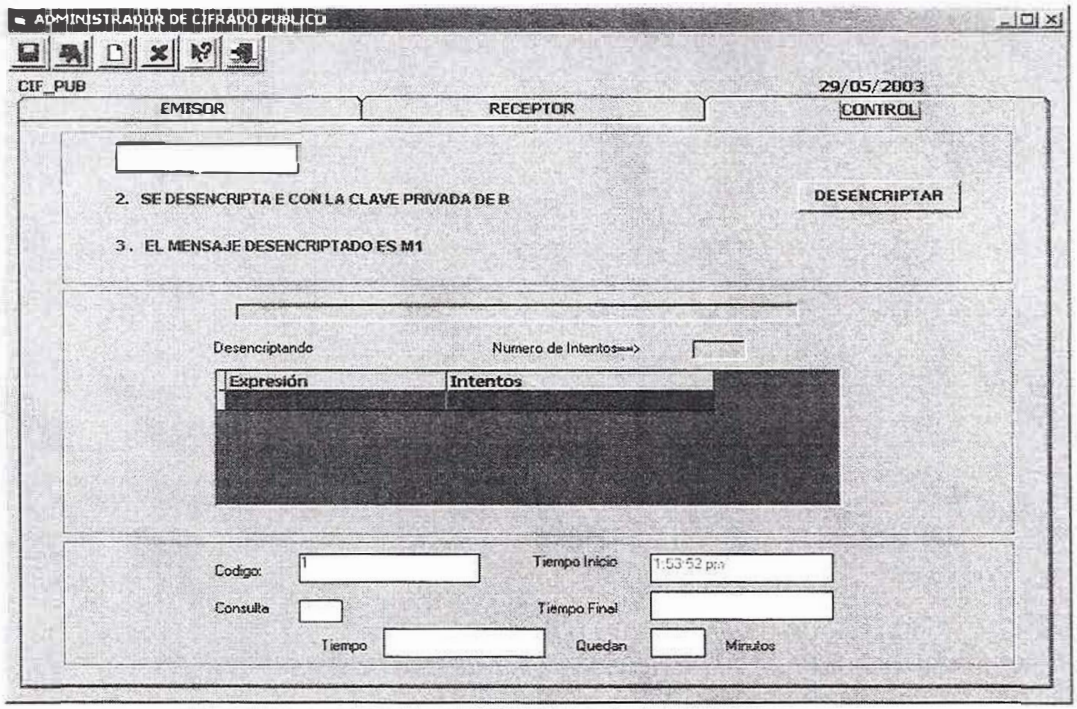


ADMINISTRADOR DE CIFRADO POR TRANSPOSICIÒN

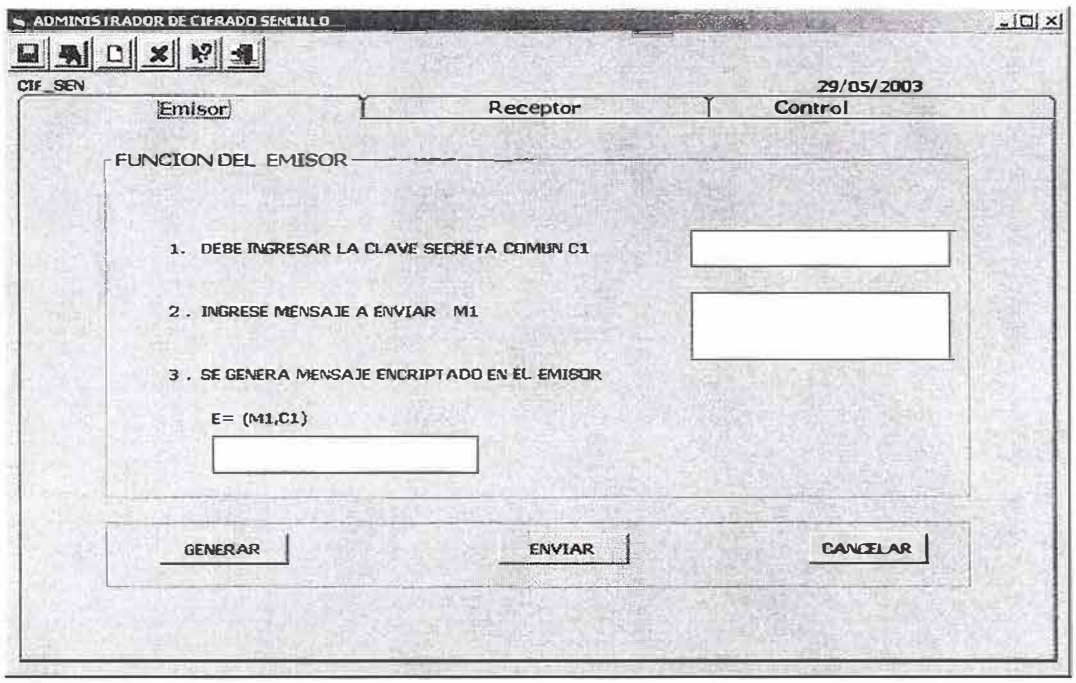
ADMINISTRADOR DE CIFRADO PUBLICO (RECEPTOR)



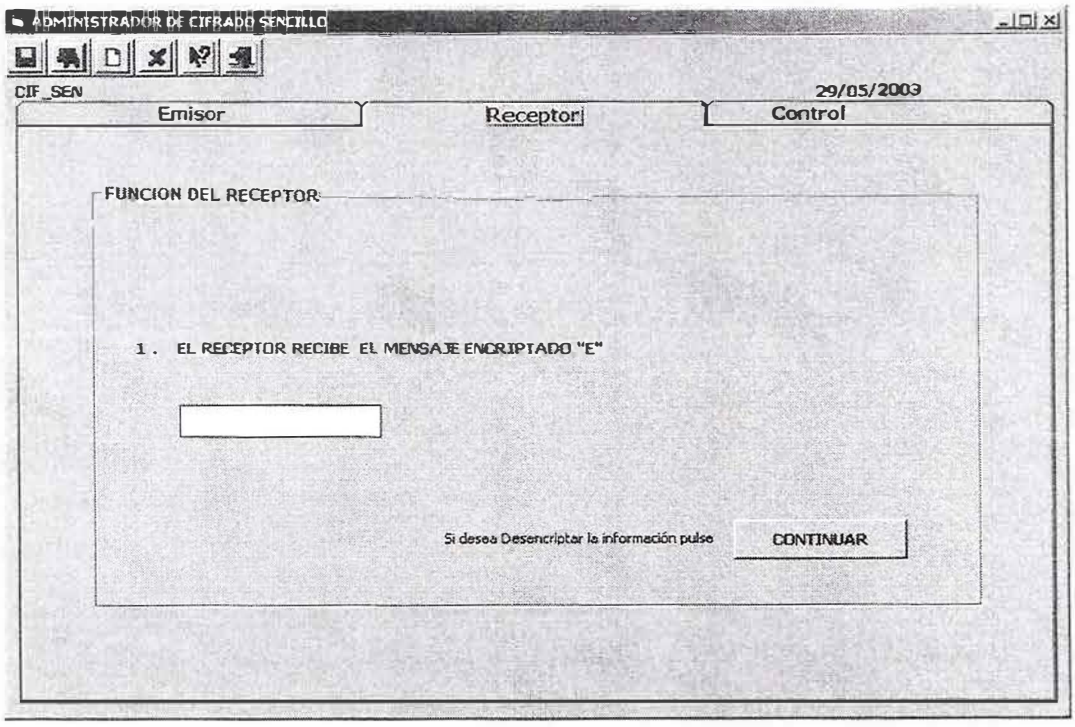
ADMINISTRADOR DE CIFRADO PUBLICO(CONTROL DE ENCRIPCIÒN)



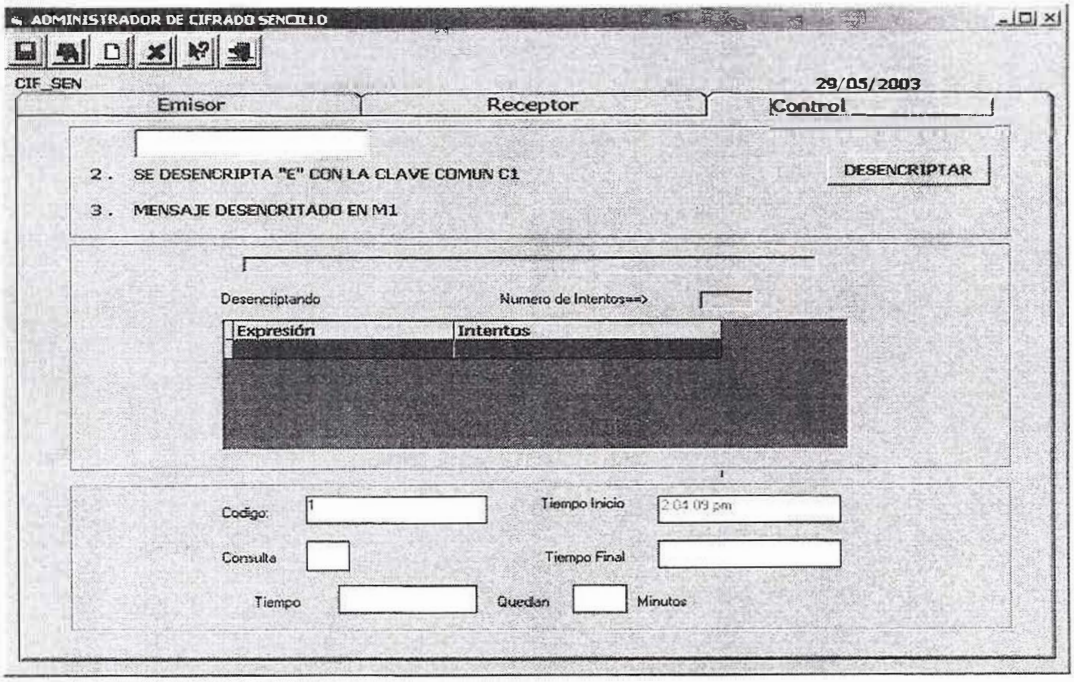
ADMINISTRADOR DE CIFRADO SENCILLO (EMISOR)



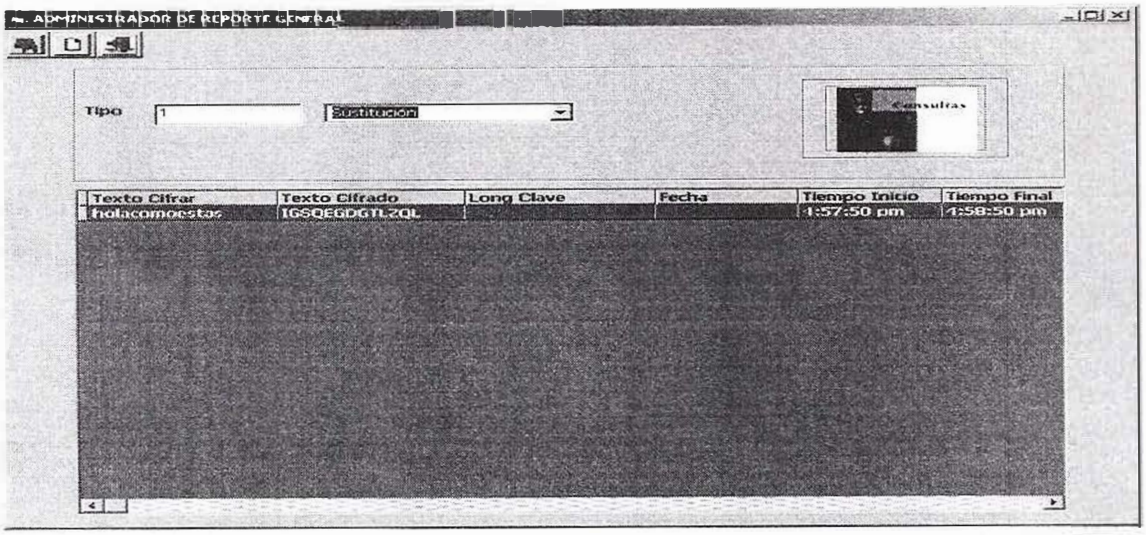
ADMINISTRADOR DEL CIFRADO SENCILLO (RECEPTOR)



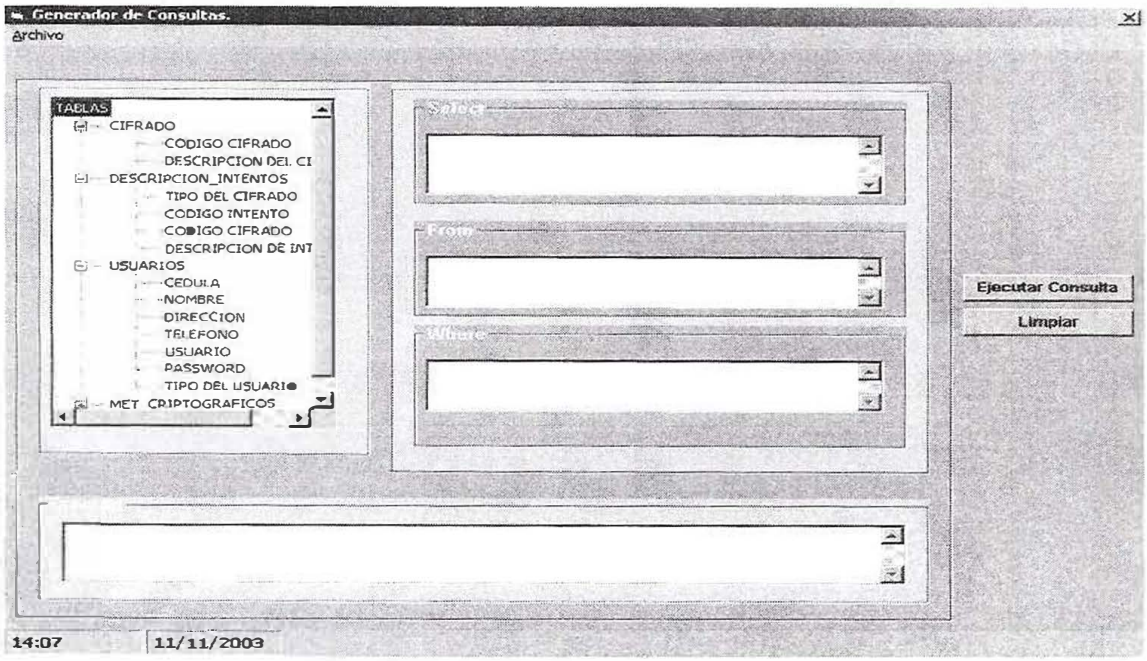
ADMINISTRADOR DE CIFRADO SENCILLO(CONTROL DE ENCRIPCIÒN)



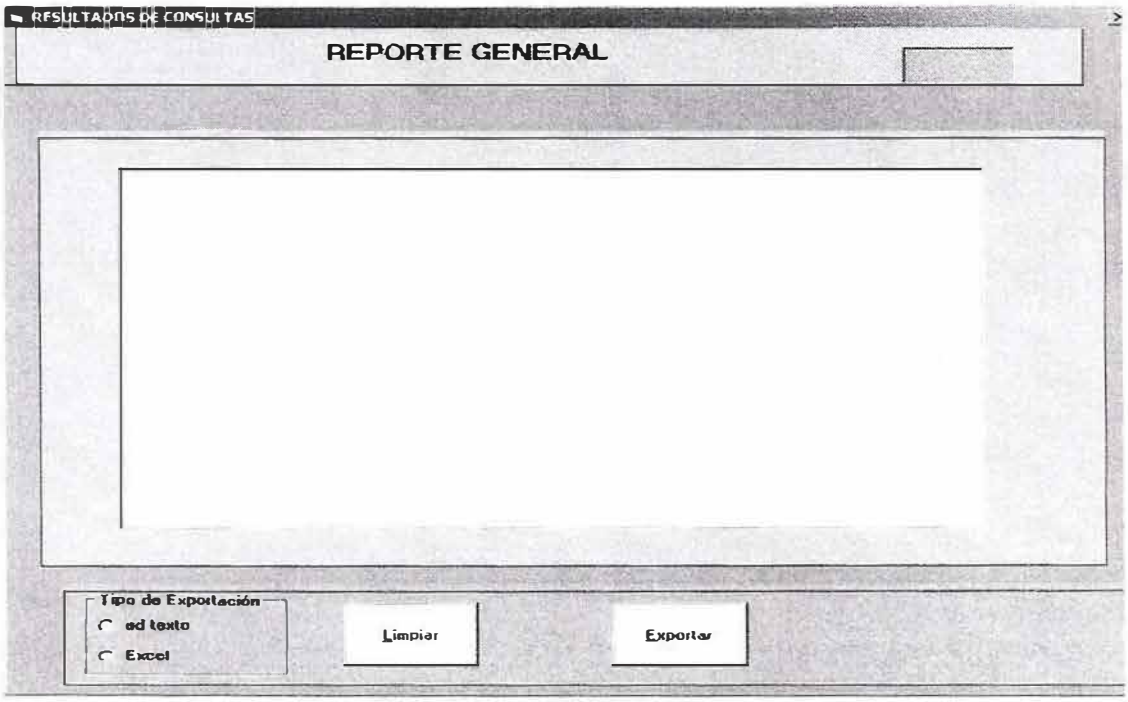
ADMINISTRADOR DE REPORTE GENERAL(CONSULTA TODA LA INFORMACIÒN ADICIONADA EN LA BASE DE DATOS)



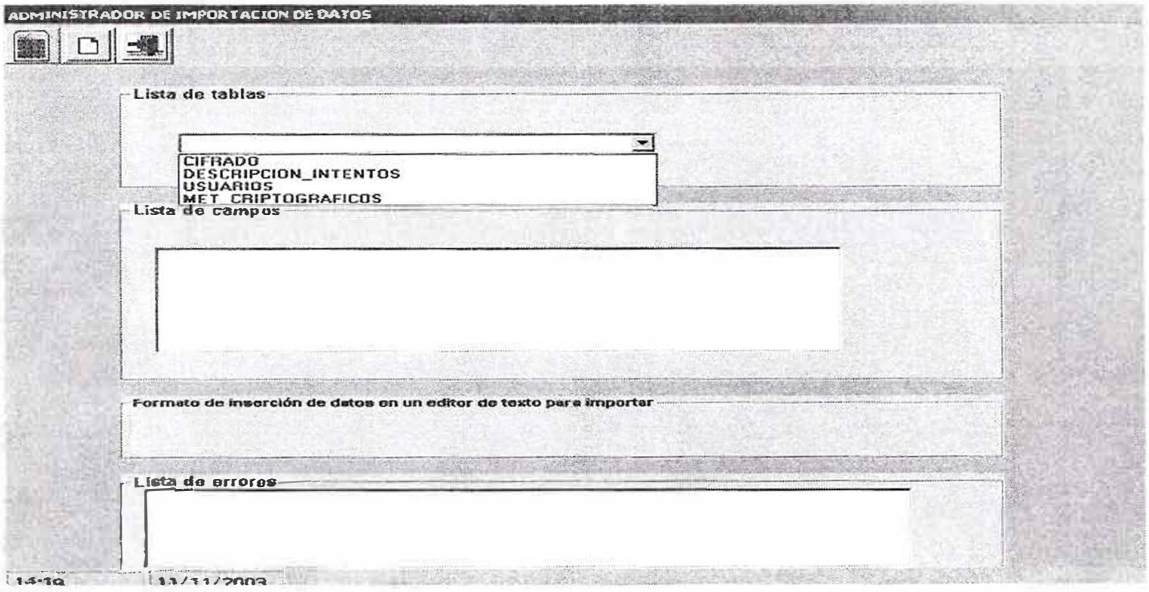
ADMINISTRADOR DEL GENERADOR DE REPORTE (GENERA TODAS LAS TABLAS Y ATRIBUTOS DE UNA FORMA DINAMICA, MOSTRANDO CADA UNA DE SUS RELACIONES)



ADMINISTRADOR DE EXPORTACIÒN X DE DATOS (EXPORTA LOS DATOS A UN EDITOR DE TEXTO O A EXCEL)



ADMINISTRADOR DE IMPORTACIÒN DE DATOS (IMPORTA LOS DATOS)



CORPORACIÓN EDUCATIVA MAYOR DEL DESARROLLO SIMÓN BOLÍVAR

MANUAL DEL SISTEMA

ANALISIS Y EVALUACIÒN DE LOS MÈTODOS CRIPTOGRAFICOS BASADOS EN LA SEGURIDAD DE LA INFORMACIÒN PARA LA CORPORACION EDUCATIVA MAYOR DEL DESARROLLO SIMON BOLIVAR

ULFRAN DIAZ BOLIVAR
JOHANA OSORIO MENA

Monografía para optar el título de
Ingenieros de Sistemas

Director
EMILIO AUQUE
Ingeniero de Sistemas

CORPORACIÓN EDUCATIVA MAYOR DEL DESARROLLO
SIMON BOLIVAR
FACULTAD DE INGENIERIA DE SISTEMAS
AREA DE INVESTIGACIÓN FORMATIVA
BARRANQUILLA
AÑO 2003

TABLA DE CONTENIDO

1. OBJETIVOS

1.1 OBJETIVO GENERAL

Analizar y evaluar los diferentes métodos de criptografía mediante pruebas constantes obteniendo conclusiones eficientes que benefician a la comunidad Estudiantil de la Corporación Educativa Mayor del Desarrollo Universidad Simón Bolívar.

1.2 OBJETIVOS ESPECÍFICOS

- ✓ *Describir los principales métodos de cifrado usando algoritmos de criptografía.*
- ✓ *Identificar las diferentes características de cada uno de los métodos de criptografía: (Cifrados tradicionales: Sustitución, Transposición. Simétricos: Cifrado Sencillo, Asimétricos: Des, Cifrado publico. Etc.*
- ✓ *Analizar las ventajas y desventajas de cada uno de los métodos de cifrado para determinar el tipo de aplicaciones en la que pueden ser utilizado eficientemente.*
- ✓ *Implementar los diferentes métodos de criptografía sometiéndolos a pruebas para evaluar su comportamiento ante diferentes condiciones.*

2. INSTALACIÓN

Para instalar el software Análisis y Evaluación de Métodos Criptográficos aplicados a la Seguridad de la Información debe cumplir con los siguientes requisitos.

- *Una unidad de CD – ROM*
- *Tener a su alcance el CD de instalación.*
- *Cerrar todos los programas que esté ejecutando antes de iniciar la instalación del software.*
- *El equipo maestro (SERVIDOR) debe tener conexión con la base de datos (oracle 8i).*
- *Los equipos subalternos deben estar conectados con el equipo maestro.*

3. REQUISITOS BÁSICOS

Para ejecutar Visual Basic, tiene que disponer de cierto Hardware Software instalado en su equipo. Entre los requisitos del sistema cabe citar los siguientes item:

- *Microsoft Windows 95 o posterior, o Microsoft Windows NT Workstation 4.0 o posterior (Se recomienda Service Pack 3).*

- *486 DX/66 MHz o modelo superior de procesador (se recomienda Pentium o superior) o cualquier procesador Alpha que ejecute Microsoft Windows NT Workstation.*

- *16 MB de RAM para Windows 95, 32 MB de RAM para Windows NT Workstation.*

- *Pantalla VGA o de mayor resolución, compatible con Microsoft Windows.*

4. DEFINICIÓN DE LA BASE DE DATOS DEL SISTEMA

4.1 LISTADO DE TABLAS

NOMBRE DE LAS TABLAS	DESCRIPCIÓN DE LAS TABLAS
CIFRADO	*En esta tabla se almacena el código y la descripción del código de la información ingresada*
MET_CRIPTOGRAFICO	*Aquí se guarda la información de todos los formularios con sus respectivos campos.*
DESCRIPCION_INTENTOS	*Esta tabla muestra la descripción de los intentos uno a uno.*
USUARIO	*Esta tabla guarda las información de la persona que ingresa al sistema.*

4.2 SCRIPT DE LA BASE DE DATOS

```
Create table MET_CRIPTOGRAFICO (
        COD_CIFRA               NOT NULL VARCHAR2(5)
        TIPO_CIFRA              NOT NULL VARCHAR2(5)
        TEXTO_CIFRAR            NOT NULL VARCHAR2(30)
        TEXTO_CIFRADO           NOT NULL VARCHAR2(30)
        TIEMPO_INICIO           NOT NULL VARCHAR2(20)
        TIEMPO_FINAL            NOT NULL VARCHAR2(20)
        TIEMPO_TOTAL            NOT NULL VARCHAR2(20)
        NUMEROS_INTENTOS        NOT NULL NUMBER(5)
        FECHA                   DATE
        LONG_CLAVE              NUMBER(10)
        RESULTADO               VARCHAR2(30)

Create table DESCRIPCION_INTENTOS (
        COD_CIFRA               NOT NULL VARCHAR2(5)
        TIPO_CIFRA              NOT NULL VARCHAR2(5)
        COD_INTENTO             NOT NULL VARCHAR2(5)
        DES_INTENTOS            NOT NULL VARCHAR2(100)

Create table USUARIOS (
        CEDULA                  NOT NULL NUMBER(20)
        NOMBRE                  NOT NULL VARCHAR2(20)
        DIRECCION               NOT NULL NUMBER(30)
        TELEFONO                NOT NULL NUMBER(20)
        E_MAIL                  NOT NULL VARCHAR2(30)
```

USUARIO *NOT NULL VARCHAR2(20)*
PASSWORD *NOT NULL VARCHAR2(20)*
TIPO_USUARIO *NOT NULL VARCHAR2(30)*

Create table CIFRADO(
COD_CIFRADO *NOT NULL VARCHAR2(2)*
DES_CIFRADO *NOT NULL VARCHAR2(30)*

4.3 DESCRIPCIÓN DE LA BASE DE DATOS

Nombre de la tabla: ***MET_CRIPTOGRAFICO***							*Contiene los antecedentes de los paciente*
Nombre largo: Métodos Criptográfico							
Aplicación: General		*Tipo: Maestro*					
No	*Nom Atributo*	*Llaves*	*Tipo de datos*	*Long*	*dec*	*null*	*Descripción*
1	*COD_CIFRA*	*Pk*	*VARCHAR2*	*5*	*0*	*No*	*Código del Cifrado*
2	*TIPO _ CIFRA*		*VARCHAR2*	*5*	*0*	*No*	*Tipo del Cifrado*
3	*TEXTO _ CIFRAR*		*VARCHAR2*	*30*	*0*	*No*	*Texto a Cifrar*
4	*TEXTO _ CIFRADO*		*VARCHAR2*	*30*	*0*	*No*	*Texto cifrado*
5	*TIEMPO_INICIO*		*VARCHAR2*	*20*	*0*	*No*	*Tiempo Inicio*
6	*TIEMPO_FINAL*		*VARCHAR2*	*20*	*0*	*No*	*Tiempo Final*
7	*TIEMPO_TOTAL*		*VARCHAR2*	*5*	*0*	*No*	*Tiempo Total*
8	*NUMEROS_INTENTOS*		*VARCHAR2*	*date*	*0*	*No*	*Números de Intentos*
9	*FECHA*		*VARCHAR2*	*30*	*0*	*No*	*Fecha*
10	*RESULTADO*						*Resultado*
Nombre de la tabla: ***DECRIPCION_INTENTOS***							*Contienen los cargos del medico*
Nombre largo: Cargo							
Aplicación: General		*Tipo: Maestro*					

No	Nom Atributo	Llaves	Tipo de datos	Long	dec	null	Descripción
1	COD_CIFRA	Pk	VARCHAR2	5	0	No	Código del Cifrado
2	TIPO _ CIFRA		VARCHAR2	5	0	No	Tipo _ cifrado
3	COD_INTENTO		VARCHAR2	5	0	No	
4	DES_ INTENTOS		VARCHAR2	100	0	No	Código de Intentos Descripción de Intentos

Nombre de la tabla: **USUARIOS**							Contiene los antecedentes de los paciente
Nombre largo: usuarios							
Aplicación: General		Tipo: Maestro					
No	Nom Atributo	Llaves	Tipo de datos	Long	dec	null	Descripción
1	CEDULA		VARCHAR2	20	0	No	Cedula del Usuario
2	NOMBRE		VARCHAR2	20	0	No	Nombre del Usuario
3	DIRECCION		VARCHAR2	30	0	No	Dirección del Usuario

4	*TELEFONO*		*VARCHAR2*	*20*	*0*	*No*	*Teléfono del Usuario*
5	*E_MAIL*		*VARCHAR2*	*30*	*0*	*No*	*E_mail del Usuario*
6	*USUARIO*		*VARCHAR2*	*20*	*0*	*No*	*Usuario*
7	*PASSWORD*		*VARCHAR2*	*20*		*No*	*Password*
8	*TIPO _ USUARIO*		*VARCHAR2*	*30*	*0*	*No*	*Tipo de Usuario*
9					*0*		
10							

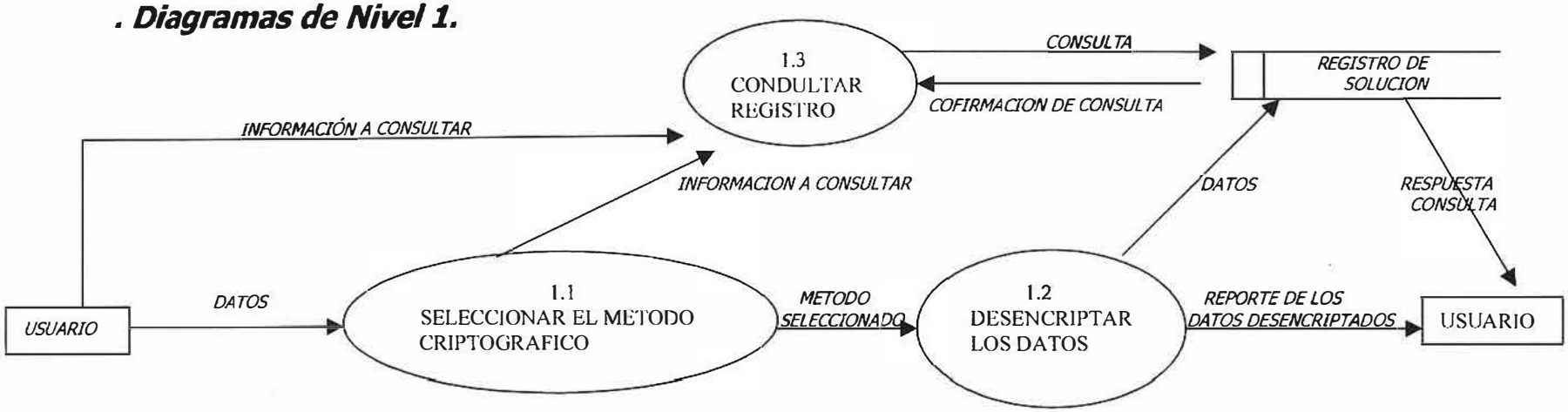
Nombre de la tabla: ***CIFRADO***							*Contienen los cargos del medico*
Nombre largo: Cifrado							
Aplicación: General		*Tipo: Maestro*					
No	*Nom Atributo*	*Llaves*	*Tipo de datos*	*Long*	*dec*	*null*	*Descripción*
1	*COD_CIFRADO*	*Pk*	*VARCHAR2*	*2*	*0*	*No*	*Código del Cifrado*
2	*DES_CIFRADO*		*VARCHAR2*	*30*	*0*	*No*	*Descripción del Cifrado*

5. MODELADOR DE PROCESOS

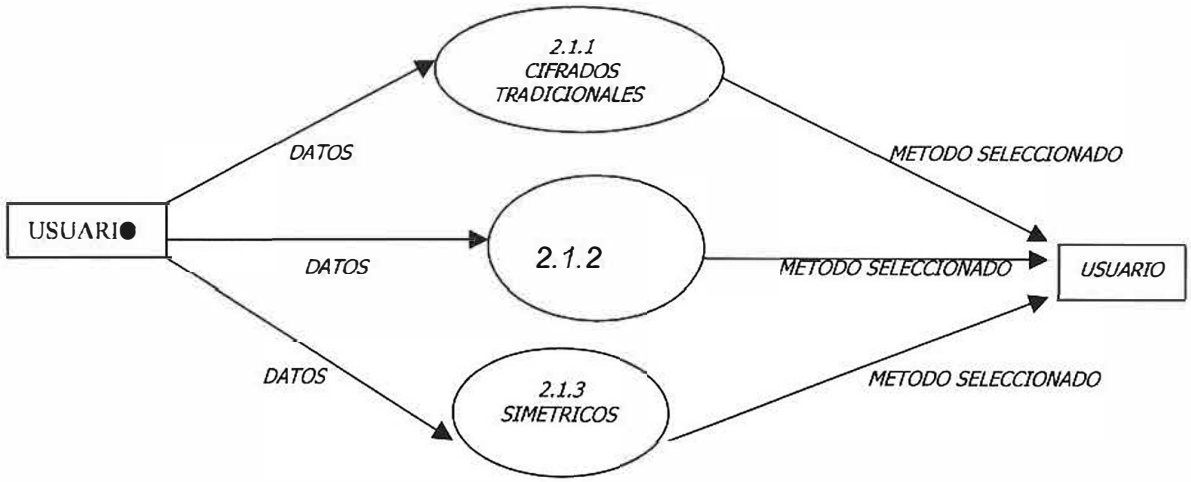
5.1 DIAGRAMA DE CONTEXTO

NIVEL 0

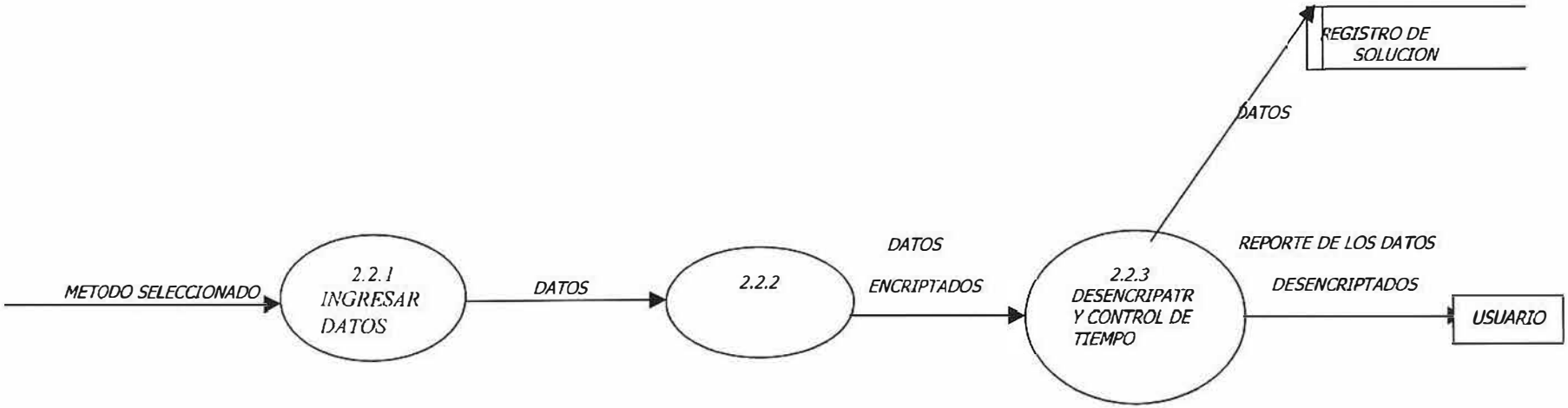
. Diagramas de Nivel 1.



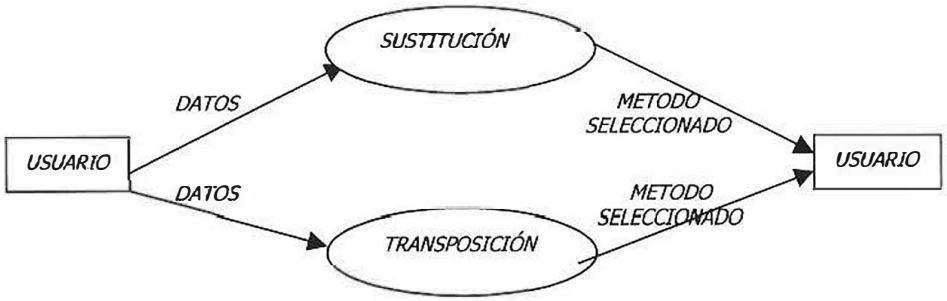
▪Diagramas de Nivel 2. Proceso 1.1



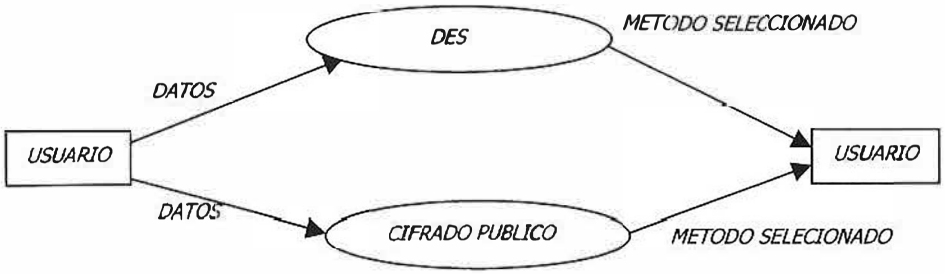
.Diagramas de Nivel 2. Proceso 1.2



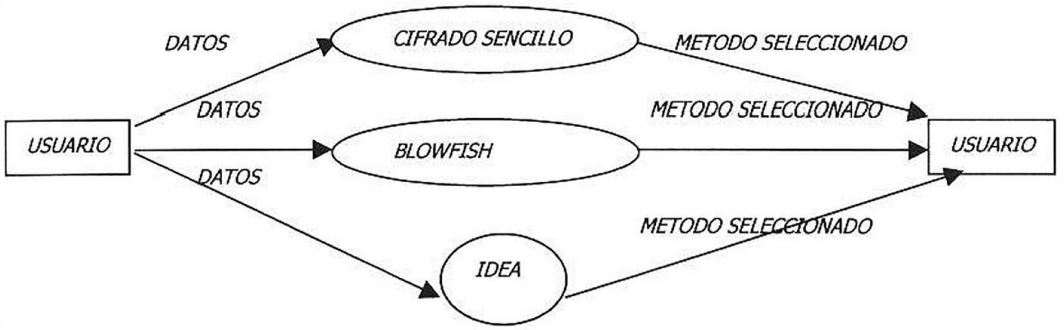
. Diagramas de Nivel 3. Proceso 2.1.1



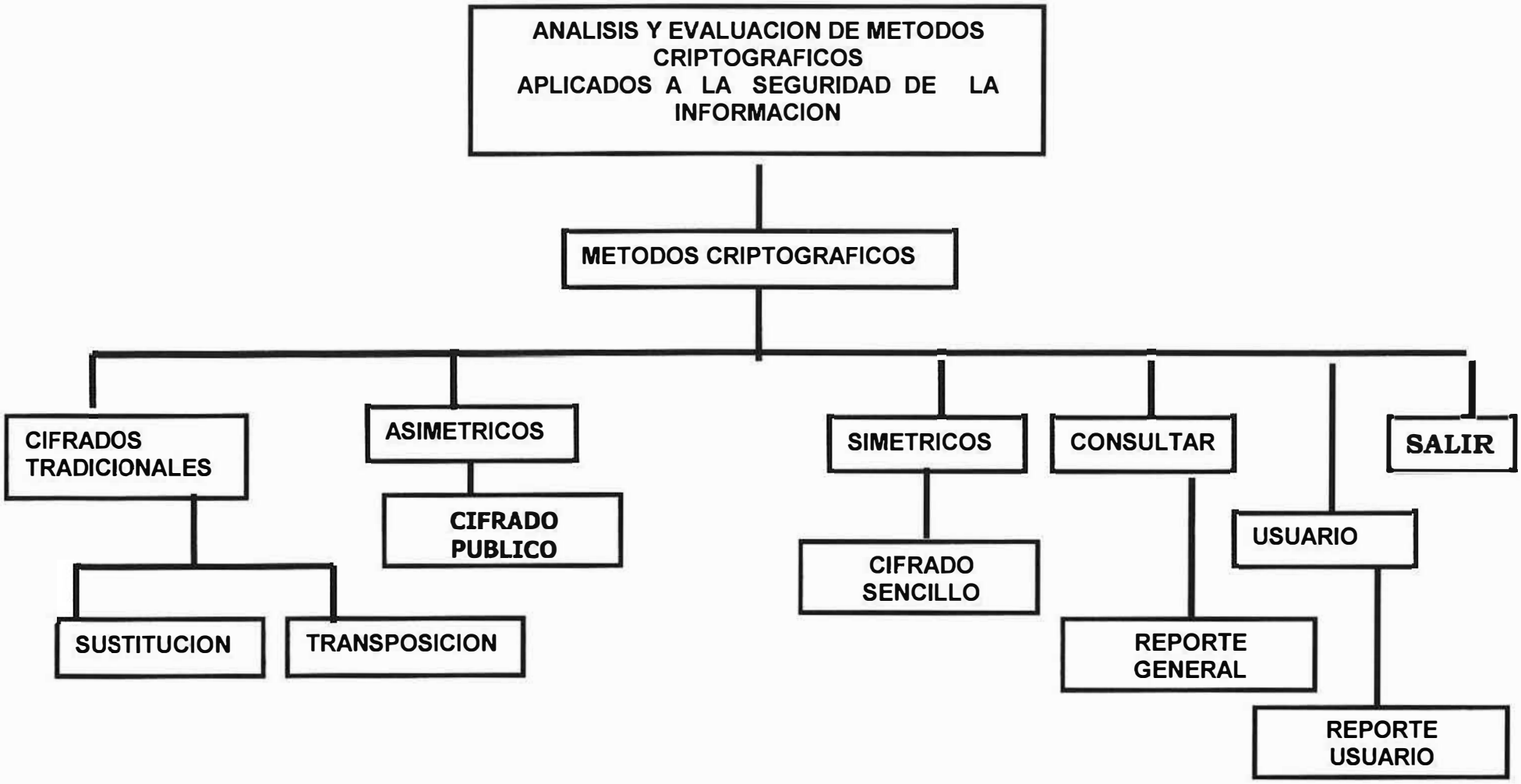
Diagramas de Nivel 3. Proceso 2.1.2



Diagramas de Nivel 3. Proceso 2.1.3



6. MAPA DE NAVEGACIÓN



7. FORMULARIOS

Para la elaboración del proyecto Diseño e Implementación de un software para Los usuarios de la Universidad Simón Bolívar se ha implementado los siguientes formularios:

Nombre	Formulario
CIFRA	*MENU*
CIF_PUB	*CIFRADO PUBLICO*
CIF_SEN	*CIFRADO SENCILLO*
OPCION1	*CIFRADO POR SUSTITUCIÒN*
OPCION2	*CIFRADO POR TRANSPOSICIÒN*
SUSTITUCIÓN	*CONTROL DE DESENCRIPCIÓN DE LA INFORMACIÓN DEL CIFRADO POR SUSTITUCION*
TRANSPOSICIÓN	*CONTROL DE DESENCRIPCIÓN DE LA INFORMACIÓN DEL CIFRADO POR TRANSPOSICIÒN*
UREPORTE	*REPORTE DEL USUARIO*
REPORTES	*REPORTE GENERAL, CONSULTAS*
EJECUCIÓN	*EJECUCIÒN, SE ENCARGA DE ENVIAR LA INFORMACIÒN DE UN FORMULARIO A OTRO*
CONTRASEÑA	*CONTRASEÑA, INICIO DE SECCIÓN*
EXPORTACIÓN	*EXPORTACIÒN DE DATOS*
IMPORTACIÓN	*IMPORTAR DATOS*
RESOLUCIÓN	*CONFIGURAR LA PANTALLA*

8. MÓDULOS

MODULO1	DESENCRIPCIÓN DE LA INFORMACIÓN
MODULO2	ENCRIPCIÓN DE LA INFORMACIÓN
MODULO3	AUDITORIA

9. PROCEDIMIENTOS

- ***NOMBRE DEL FORMULARIO: CONTRASEÑA***

  - *Private Sub desencriptar ()*

Este procedimiento compara si la clave es valida o no, se encarga de ir a la base de datos y comparar si la información es correcta, esto se hace al momento de desencriptara la información.

- ***NOMBRE DEL FORMULARIO: CIFRA***

  - *Private Sub cargar()*

Este procedimiento muestra en el TreeView el nombre de los diferentes formulario que se desee mostrar.

  - *Private Sub keyformularios()*

Este procedimiento llama al treeview cada uno de los formularios.

- ***NOMBRE DEL MODULO: AUDITORIA***

  - *Public Sub INGRESAR_MOVIMIENTO*

Este procedimiento es el que controla el ingreso de los usuarios al sistema indica cada una de las operaciones que se realizan.

  - *Public Sub llena*

Este procedimiento consulta en el vsflexgrid cada uno de los movimiento que realizan los diferentes usuarios que ingresan al sistema.

- ***NOMBRE DEL FORMULARIO:GENERAR_REPORTES***

  - *Private Sub TreeView1_NodeClick*

Este procedimiento carga el nombre de cada una de las tablas que se deseen mostrar para los reportes dinámicos.

  - *Private Sub CmdEjecutaQuery_Click()*

Este procedimiento se utiliza para ver la consulta que el usuario halla

hecho en el generador de reportes dinámicos para dar ejecución a la consulta es necesario cargar otro formulario que es donde se muestran los resultados(EXPORTAR).

- ***NOMBRE DEL FORMULARIO: IMPORTAR***

  - *Private Sub sitablas_Click()*

Este procedimiento es el que carga el nombre de las tablas y el nombre de los campos y el tipo de dato de cada campo a las que se le deseen Hacerles el proceso de importación. En este caso se cargan, las tablas que se encuentran en el generador de reportes dinámicos.

  - *Private Sub import()*

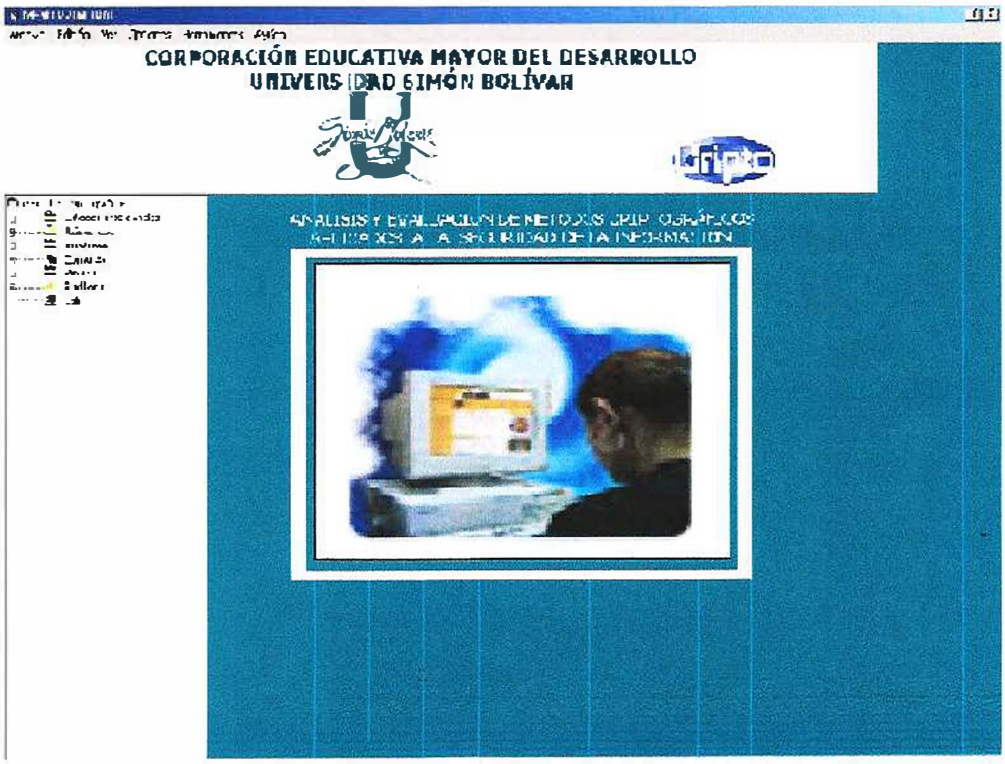
Este procedimientos es el que se encarga de seleccionar el archivo que se va a importar y además especifica que tipo de errores se presentan al momento de realizar este proceso.

  - *Public Sub cargar*

Este procedimiento es el que se encarga de hacer el proceso de importación a la base de datos, es el que hace el insert a la tabla que halla escogido el usuario para realizar este proceso.

CORPORACIÓN EDUCATIVA MAYOR DEL DESARROLLO SIMÓN BOLÍVAR

MANUAL DEL USUARIO



ANALISIS Y EVALUACIÒN DE LOS MÈTODOS CRIPTOGRAFICOS BASADOS EN LA SEGURIDAD DE LA INFORMACIÒN PARA LA CORPORACION EDUCATIVA MAYOR DEL DESARROLLO SIMON BOLIVAR

ULFRAN DIAZ BOLIVAR
JOHANA OSORIO MENA

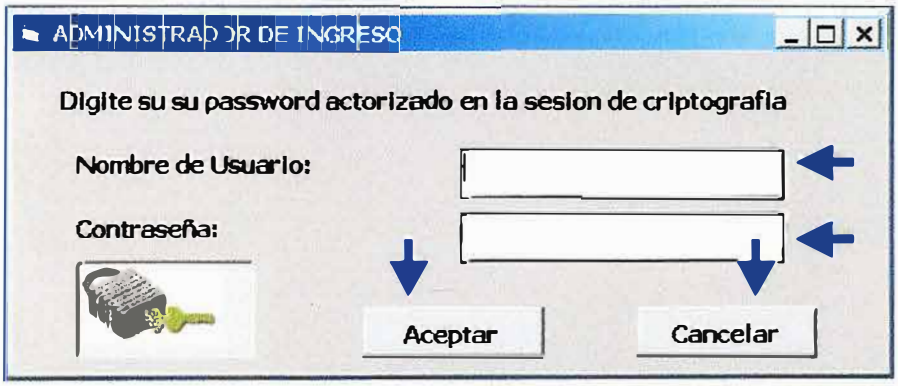
Director
Luisa Arrieta
Ingeniera de Sistemas

CORPORACIÓN EDUCATIVA MAYOR DEL DESARROLLO SIMON BOLIVAR
FACULTAD DE INGENIERIA DE SISTEMAS
AREA DE INVESTIGACIÓN FORMATIVA
BARRANQUILLA
AÑO 2003

$k#

1. INGRESO AL SISTEMA (METODOS CRIPTOGRAFICOS)

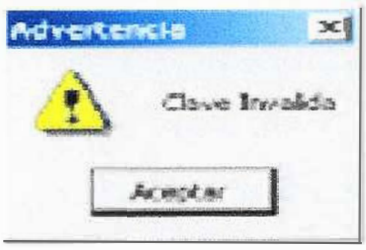
La persona que entra al sistema debe tener un perfil asignado por el administrador del sistema el cual le permitirá navegar por cada una de las opciones que compone Métodos criptográficos.



Usted debe realizar los siguientes pasos:

- *Escriba el nombre del usuario en la caja de texto* →
- *Escriba su contraseña o password* →
- *Oprima la tecla aceptar o presione enter* →

Observaciones: tanto el Nombre como el Password deben ser ingresado, si no son validos aparecerá el siguiente mensaje:



Si usted no desea entrar al sistema, elija la opción cancelar. →

[$] Ingreso al sistema
[k] Ingreso al sistema
[#] id 1

$K#

2. MENU PRINCIPAL

El Menú Principal esta compuesto por una barra de Menú y un Árbol de Navegación (Treeview), como aparece en la **Figura 1**.

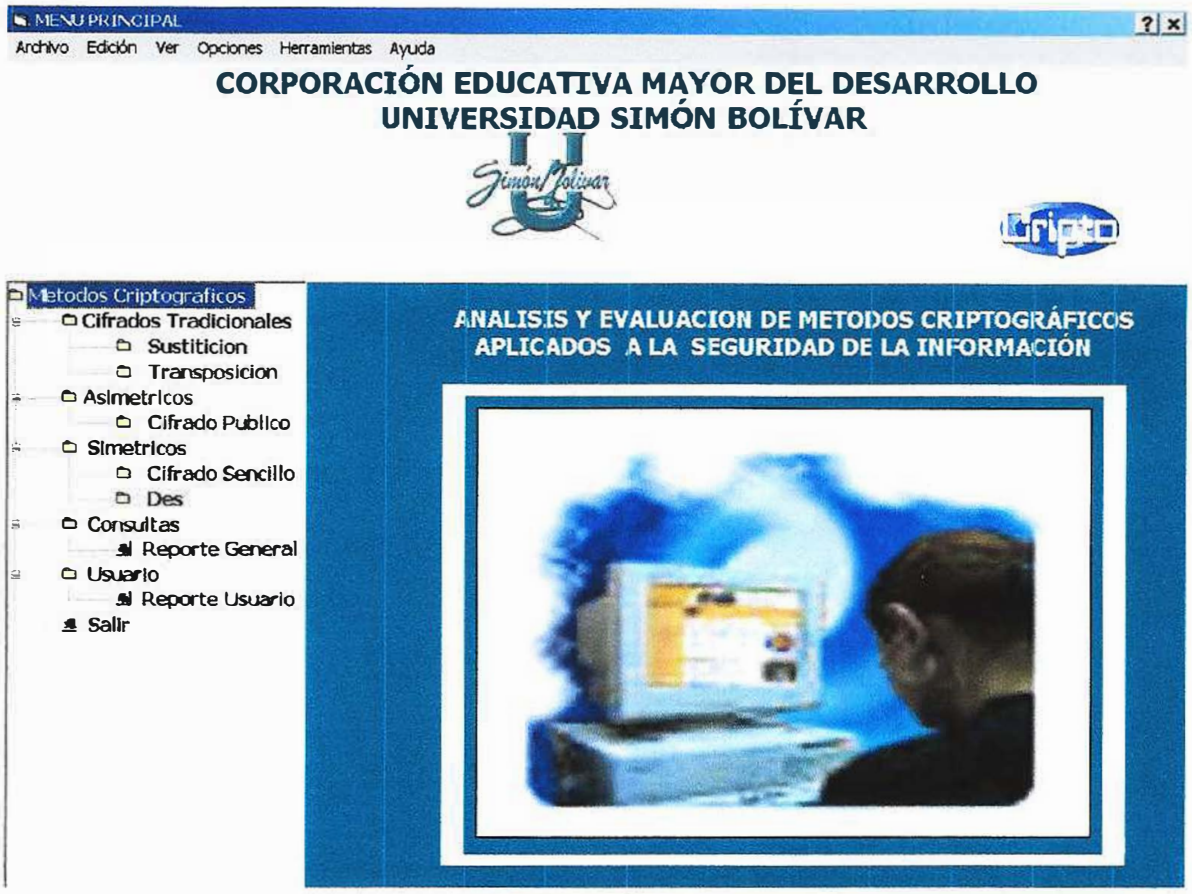


Figura 1

El Menú Principal contiene diferentes clases de carpetas como son:

- *Cifrado Tradicionales*
- *Asimétrico*
- *Simétrico*

$ Menú Principal
K menú principal y sus opciones
id2

$k#

2.1 CIFRADOS TRADICIONALES

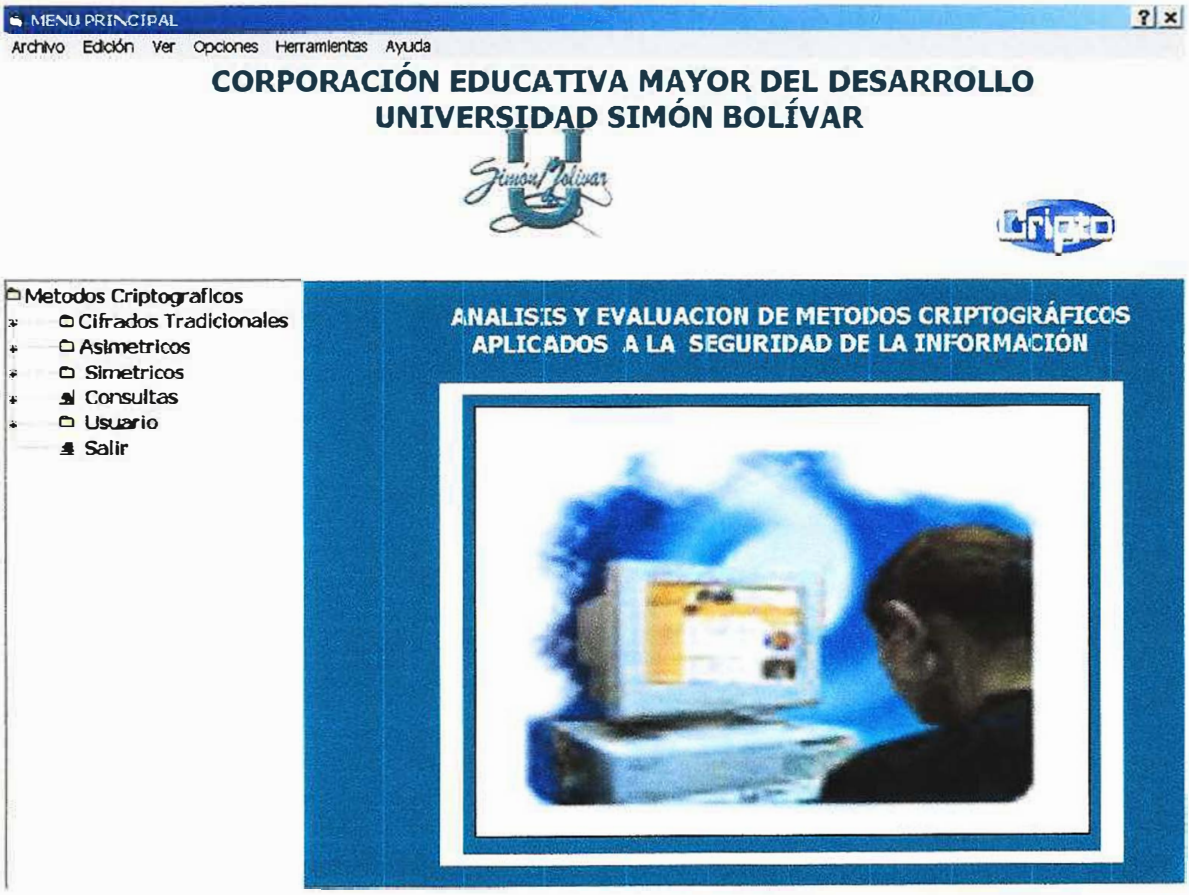


Figura 2

En esta carpeta el usuario se encarga de ingresar, guardar, consultar.

Dentro de la carpeta de Cifrados Tradicionales encontramos las opciones de:

- *Cifrados por Sustitución.*
- *Cifrados por Transposición*

[$] Cifrados Tradicionales
[k] Cifrados Tradicionales
[#] id 2.1

2.1.1 CIFRADO POR SUSTITUCIÓN

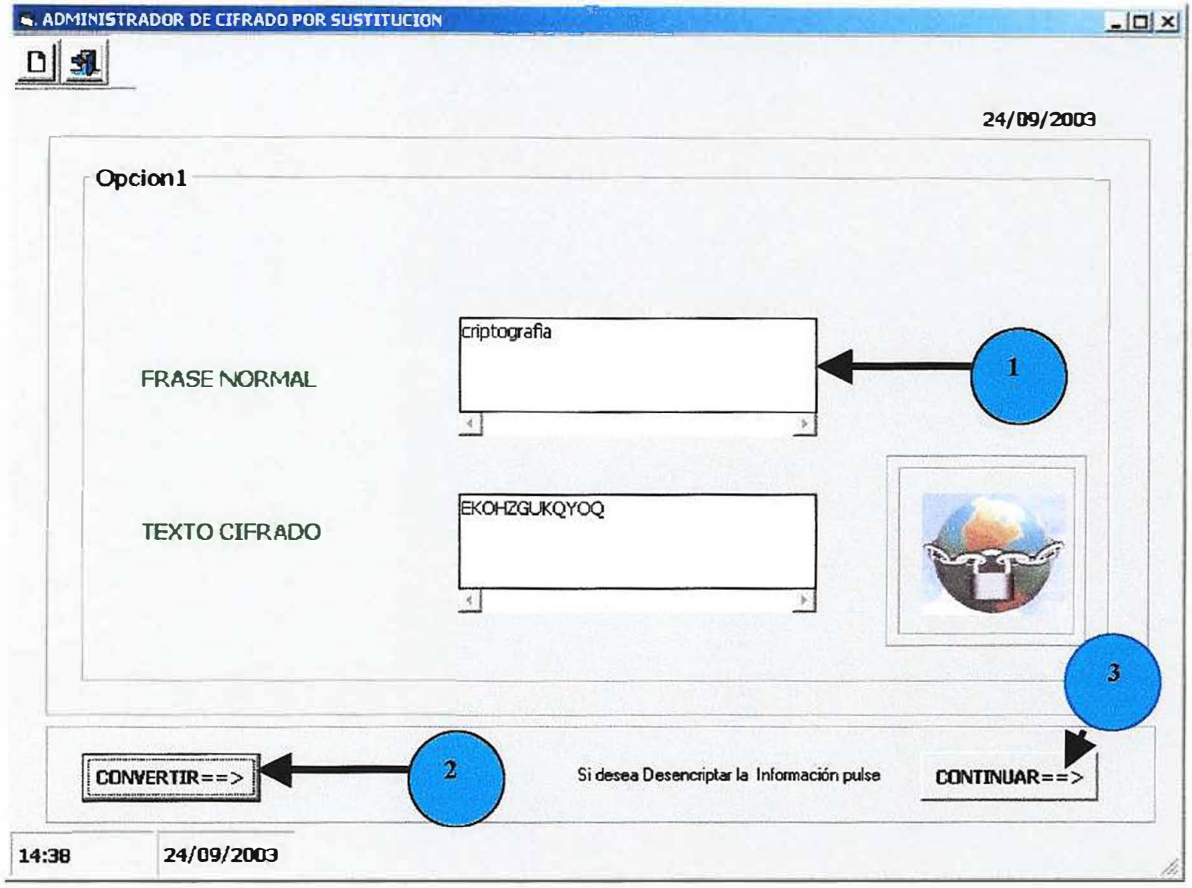


Figura 3

El Formulario de Sustitución encripta toda la Información que ingresan los Usuarios que ingresan al sistema.

Se maneja de la siguiente manera.

1. *Ingrese datos en **Frase Normal.***
2. *Haga clic en el botón Convertir como se muestra en* **Figura 3** *para que encripte la información.*

[$] Cifrados Tradicionales
[k] Cifrado por Sustitución
[#] id 2.1.1

- *Debe hacer clic en el botón que muestra la Figura* 3 *para que pueda desencriptar la información.*

Después le aparecerá la siguiente ventana:

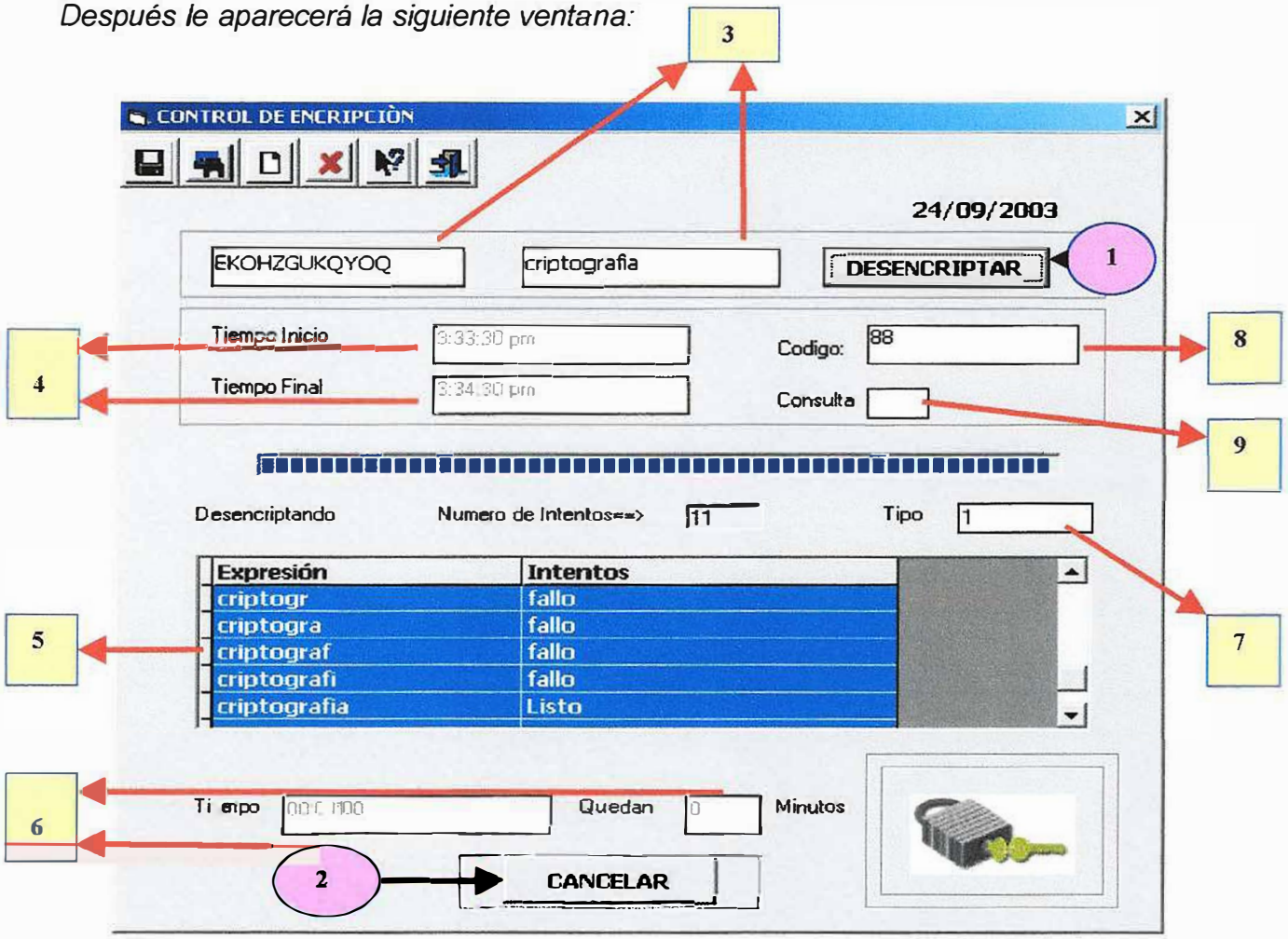


Figura 4

1 *Este botón nos permite desencriptar la información.*

2 *Haga clic como se muestra con la flechita para interrumpir al estar Guardando la información ya desencriptada.*

3	*Nos muestra mensaje Encriptado Y Desencriptado.*
4	*Nos indica el Tiempo en que inicia el Control de Encripciòn a Desencriptar la Información, tomando el tiempo en que inicia y el final.*
5	*Al desencriptar la información el nos muestra una a una los datos que se va Encontrando.*
6	*Nos indica los minutos que durará la ejecución y el tiempo restante.*
7	*Nos indica con que tipo de algoritmo estamos trabajando.*
8	*Información almacenada en la Base de Datos.*
9	*Nos permite consultar la Información deseada desde el mismo formulario.*

$k#

2.1.2 CIFRADO POR TRANSPOSICIÒN

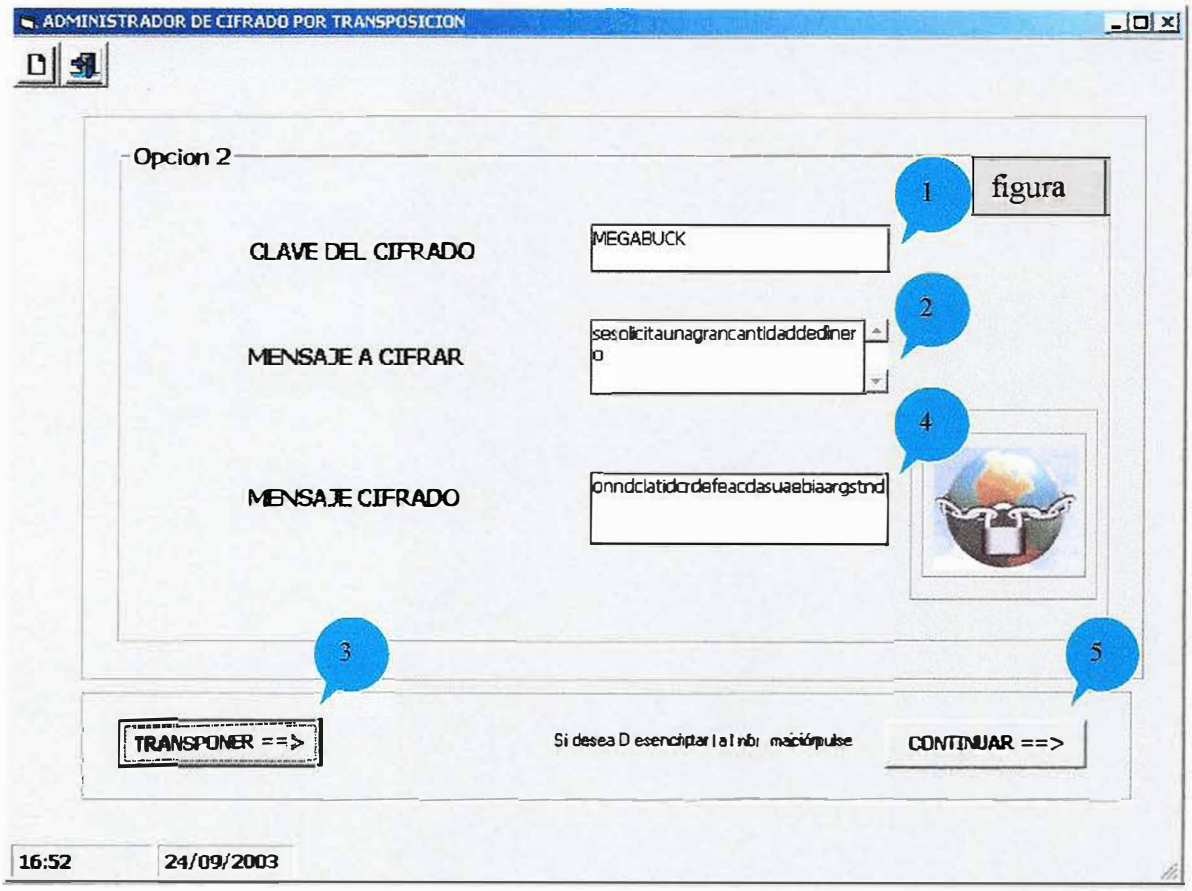


Figura 5

Al ingresar a este formulario usted debe digitar información en ***Clave del Cifrado*** *(como se muestra en la Figura 1) por la cual la Clave no debe contener letras repetidas. En caso de haber ingresado letras repetidas les mostrará una ventana como la siguiente:*

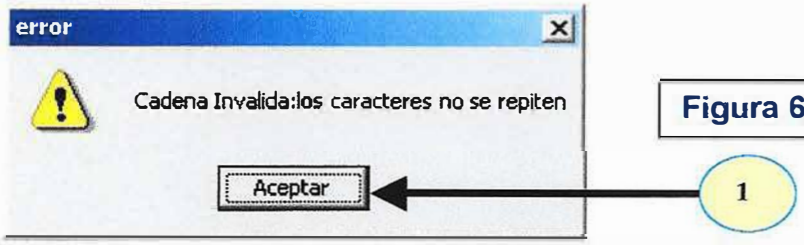


Figura 6

[S] Cifrados Tradicionales
[k] Cifrado por Transposición
[#] id 2.1.2

Luego pulsan donde nos muestra la figura 1 *y se ubican a ingresar de nuevo la* ***Clave del Cifrado*** *correctamente.*

Después ingresan el mensaje que desean enviar en ***Mensaje a Cifrar*** 2

3 *A continuación Pulsan el botón que muestra la figura para transponer la información ò sea Cifrada, el cual nos indica la figura* 4

5 *Este botón nos permite mostrar otro formulario el cual nos llevaría al Formulario control de Encripciòn que desencriptara la información.*

A continuación mostraremos el formulario de ***Control de Encripciòn:***

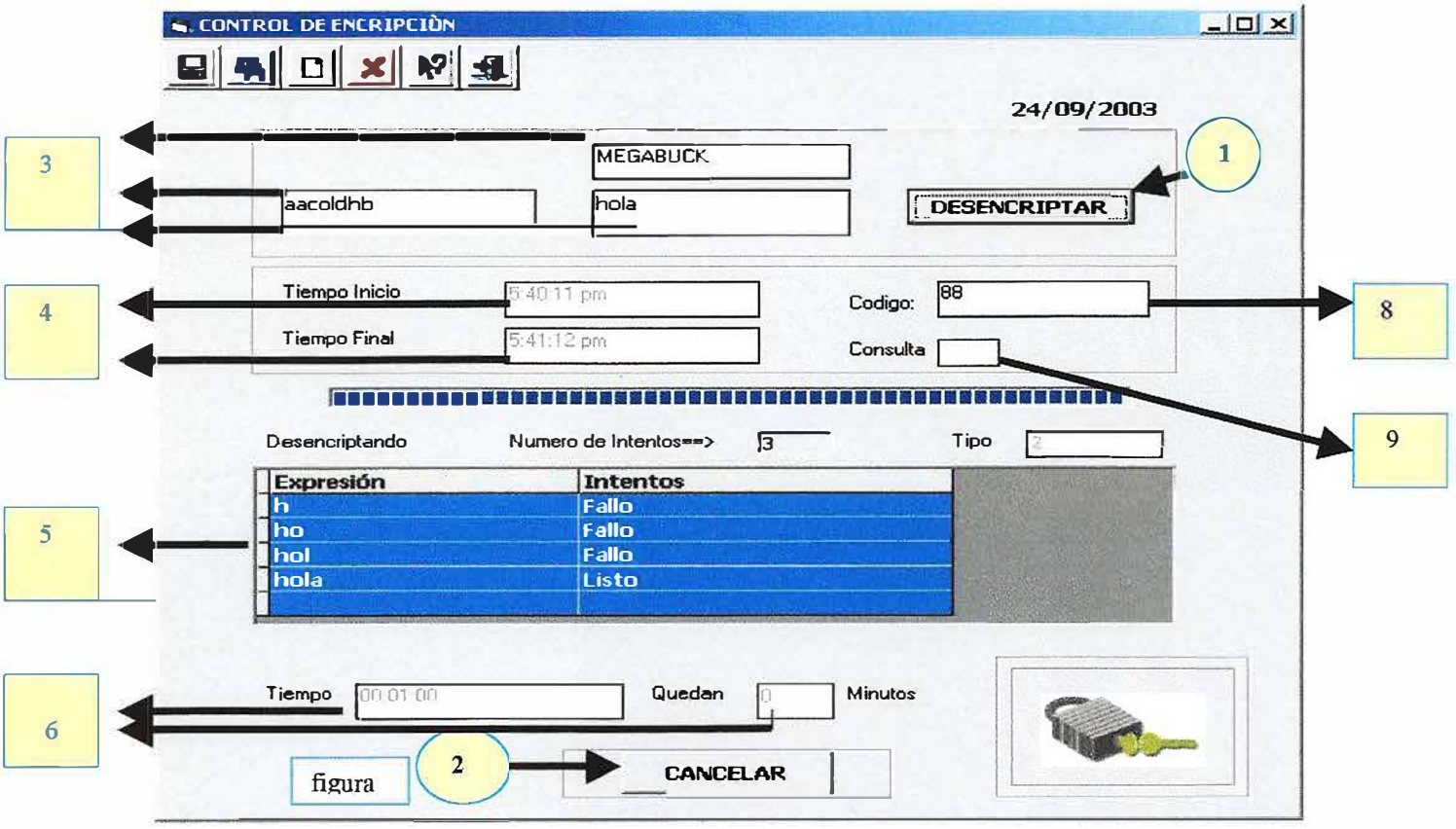


Figura 7

1 *Este botón nos permite desencriptar la información.*

2 *Haga clic como se muestra con la figura anterior para interrumpir al estar Guardando la información ya desencriptada.*

3 *Nos muestra mensaje Encriptado Y Desencriptado con su respectiva Clave.*

4 *Nos indica el Tiempo en que inicia el Control de Encripciòn a Desencriptar la Información, tomando el tiempo en que inicia y el final.*

5 *Al desencriptar la información él nos muestra una a una los datos que se van Encontrando.*

6 *Nos indica los minutos que durará la ejecución y el tiempo restante.*

7 *Nos indica con que tipo de algoritmo estamos trabajando.*

8 *Información almacenada en la Base de Datos.*

9 *Nos permite consultar la Información deseada desde el mismo formulario.*

2.1.3 CIFRADO PUBLICO

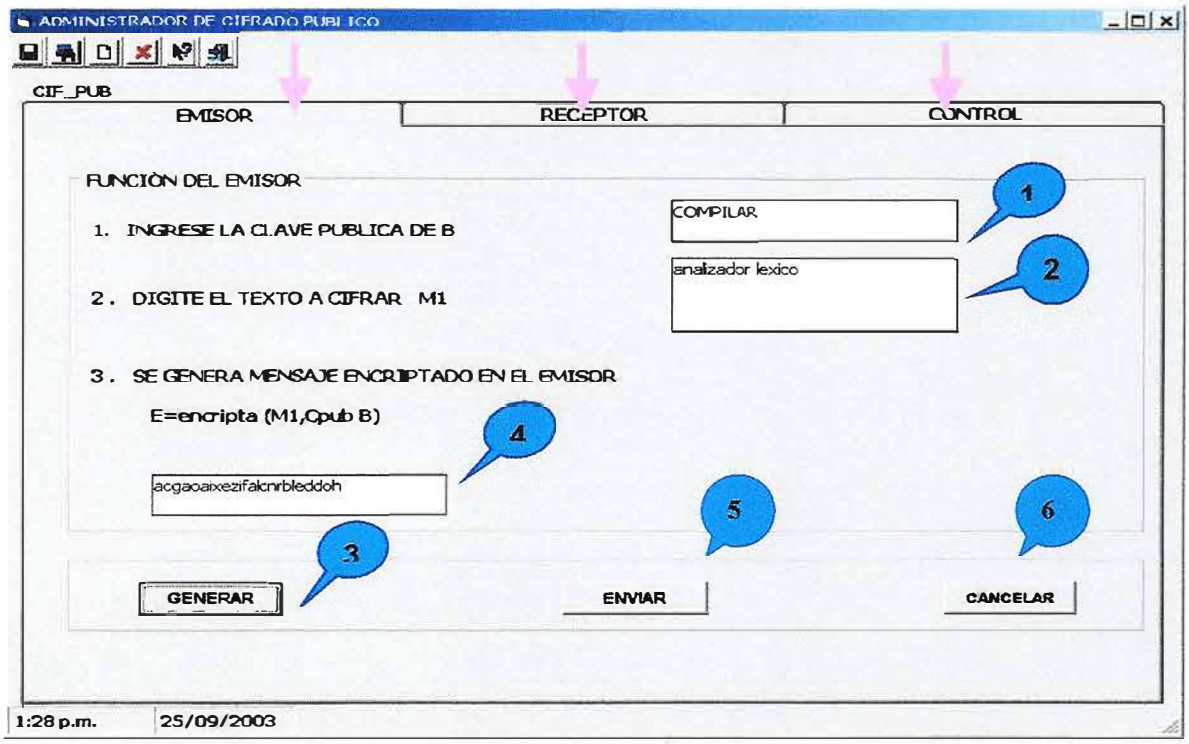


Figura 8

Este formulario se usa el componente TabStrip el cual esta dividido por tres pestañas como nos indica la flecha , la primera que es el Emisor que se encarga del envió de información, el segundo que es el Receptor que se encarga de recibir la información y por ultimo el control de Encripciòn que se encarga de verificar desencriptanto la información y llevando un control de tiempo que determinará cuanto tiempo duro la ejecución.

Aquí debe escribir la Clave el cual no debe contener letras repetidas.

***NOTA**: en caso de repetir las letras en la caja de texto donde debe ingresar la Clave les aparecerá el siguiente mensaje:*

[$] Cifrado Asimétrico
[k] Cifrado Publico
[#] id 2.1.3

Figura 9

y deberán pulsar donde les muestra la figura 1 *y luego deben ingresar la Clave correctamente.*

2 *Al ubicarse aquí deben ingresar el mensaje que desean enviar al emisor.*

3 *Haga clic en éste botón para que le encripte la información e inmediatamente le aparecerá la información cifrada.*

4 *En esta caja de texto muestra la información cifrada.*

5 *Luego presiono el botón enviar para que el Receptor reciba la información encriptada o cifrada que le ha enviado el Emisor.*

Al haber presionado aparecerá el siguiente formulario que es el que se encarga de enviarle la información al Receptor:

Figura 10

Cuando ya recibe la información el Receptor nos muestra el siguiente ítem:

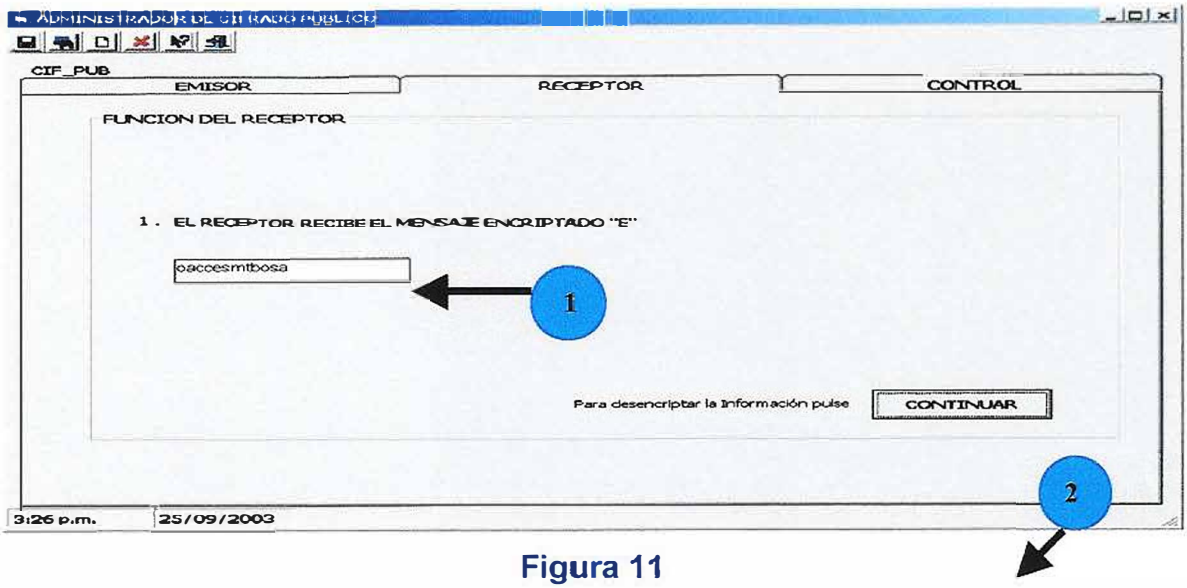


Figura 11

1 *Aquí nos indica que el receptor recibió la información encriptada.*

Luego debe presionar el botón que aparece en la figura 2 *para continuar con el proceso que nos llevará a la tercera pestaña la cual es la siguiente:*

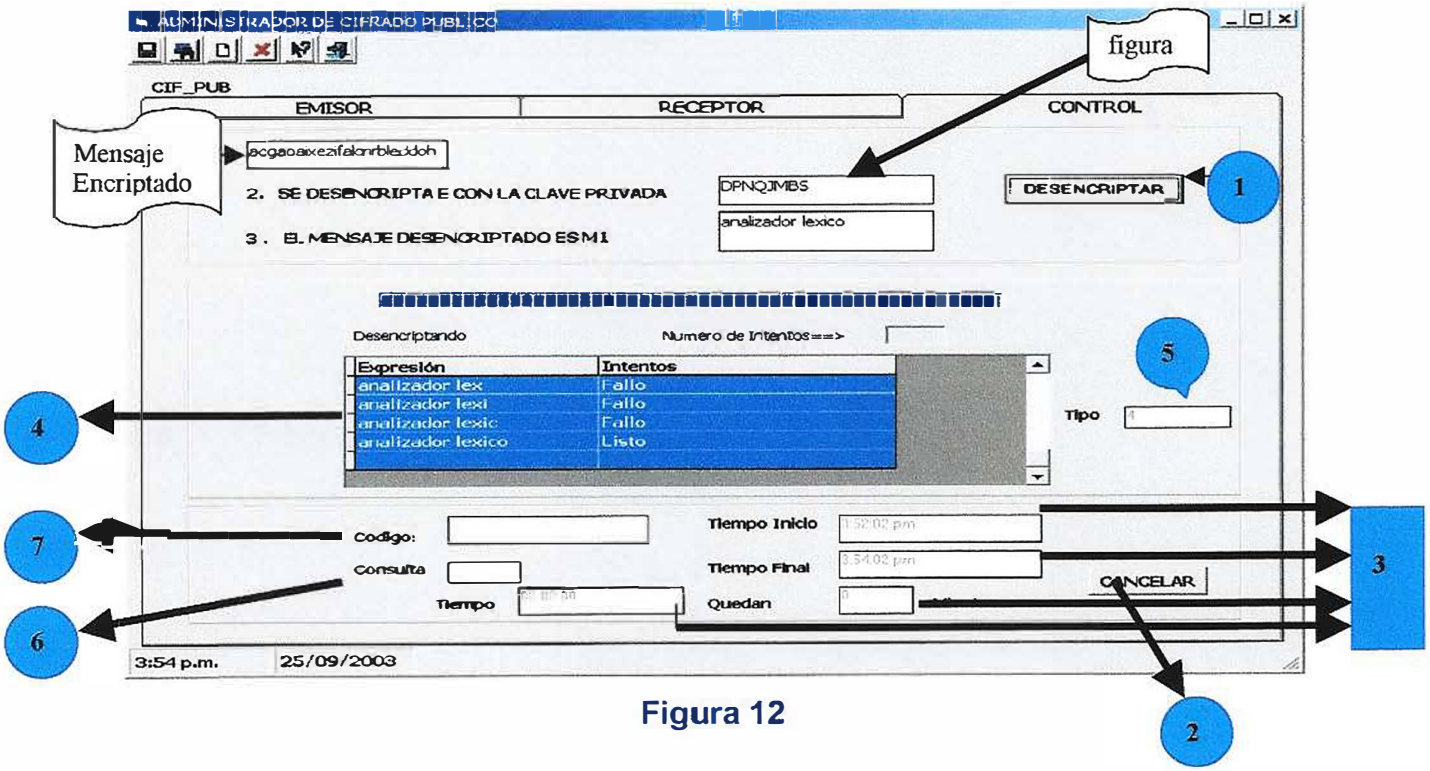


Figura 12

como se darán cuenta la Clave es diferente como muestra la figura que el Cifrado Publico trabaja con dos claves para obtener mayor seguridad.

1 *Este botón nos permite desencriptar la información.*

2 *Haga clic como se muestra con la figura anterior para interrumpir al estar Guardando la información ya desencriptada.*

3 *Nos indica el Tiempo en que inicia el Control de Encripciòn a Desencriptar la Información, tomando el tiempo en que inicia y el final.*

4 *Al desencriptar la información él nos muestra una a una los datos que se van Encontrando.*

5 *Nos indica con que tipo de algoritmo estamos trabajando.*

6 *Información almacenada en la Base de Datos.*

7 *Nos permite consultar la Información deseada desde el mismo formulario.*

2.1.4 CIFRADO SENCILLO

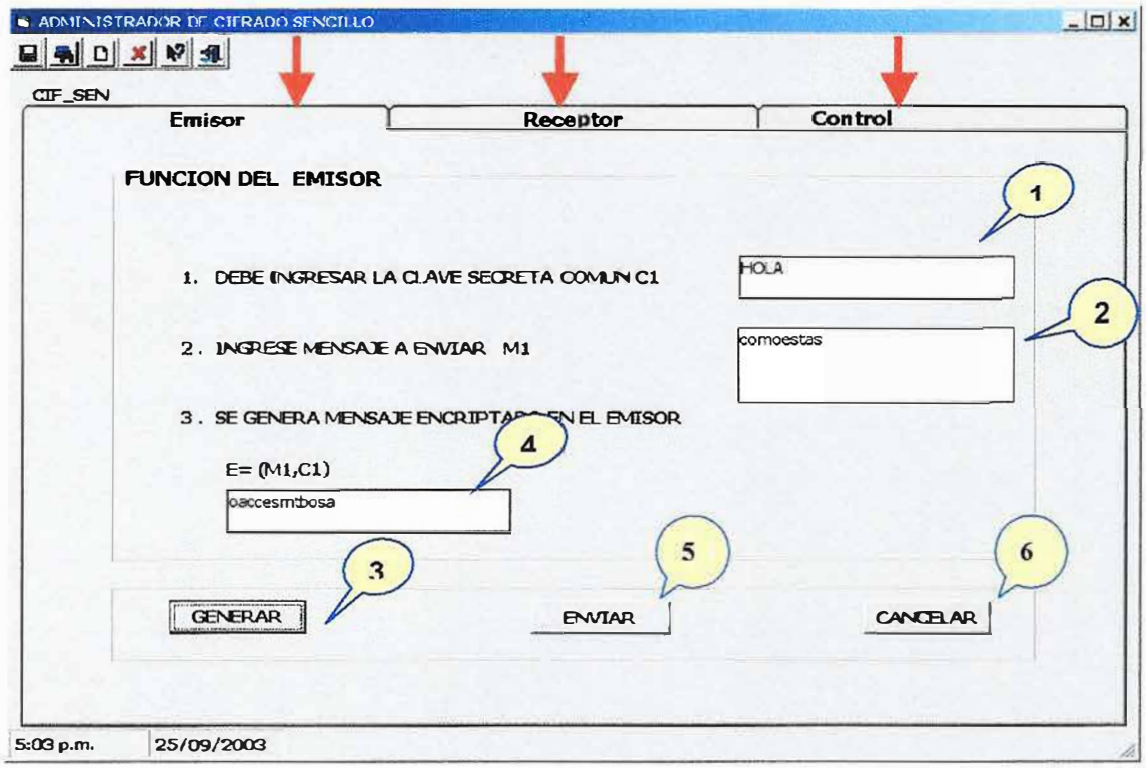


Figura 13

Este formulario también usa el componente TabStrip el cual esta dividido por tres pestañas como lo indica la flecha ➔ *, la primera que es el Emisor que se encarga del envió de información, el segundo que es el Receptor que se encarga de recibir la información y por ultimo el control de Encripciòn que se encarga de verificar desencriptanto la información y llevando un control de tiempo que determinará cuanto tiempo duro la ejecución.*

1 *Aquí debe escribir la Clave el cual no debe contener letras repetidas.*

[s] Simétricos
[k] Cifrado Sencillo
[#] id 2.1.4

***NOTA**: en caso de repetir las letras en la caja de texto donde debe ingresar la Clave les aparecerá el siguiente mensaje:*

Figura 14

y deberán pulsar donde les muestra la figura (1) *procediendo a ingresar la Clave correctamente.*

(2) *Al ubicarse aquí deben ingresar el mensaje que desean enviar al emisor.*

(3) *Haga clic en éste botón para que le encripte la información e inmediatamente le aparecerá la información cifrada.*

(4) *En esta caja de texto muestra la información cifrada.*

(5) *Luego presiono el botón enviar para que el Receptor reciba la información encriptada o cifrada que le ha enviado el Emisor.*

Al haber presionado aparecerá el siguiente formulario que es el que se encarga de enviarle la información al Receptor:

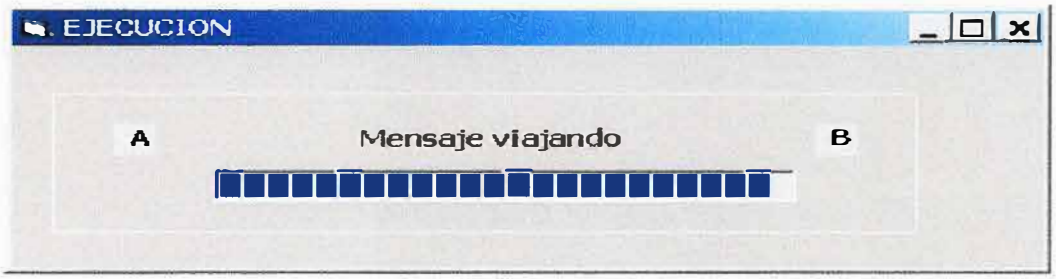


Figura 15

Cuando ya recibe la información el Receptor nos muestra el siguiente ítem:

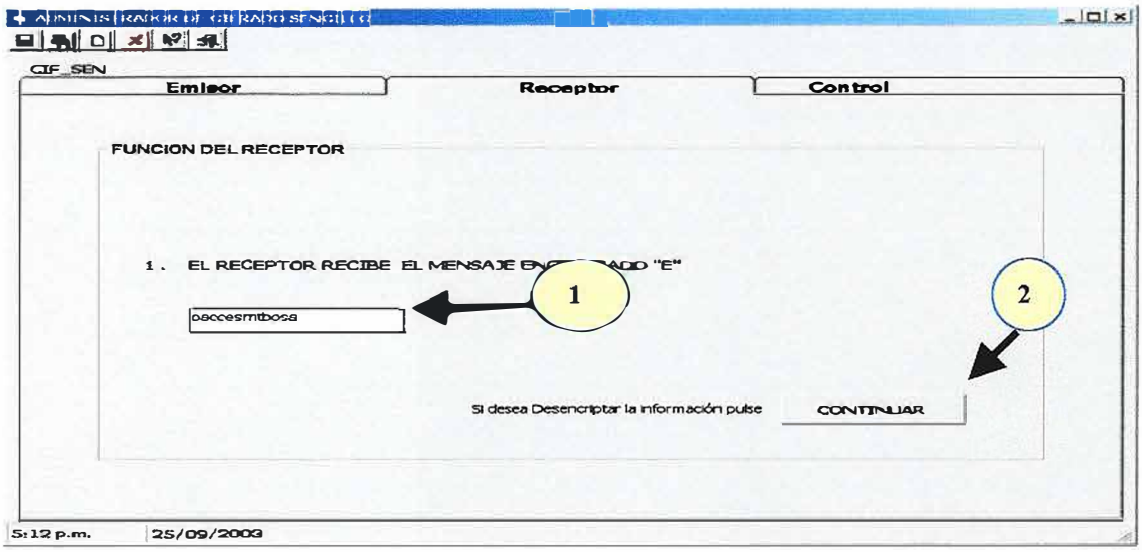


figura 16

1 *Aquí nos indica que el receptor recibió la información encriptada.*

Luego debe presionar el botón que aparece en la figura 2 *para continuar con el proceso que nos llevará a la tercera pestaña la cual es la siguiente:*

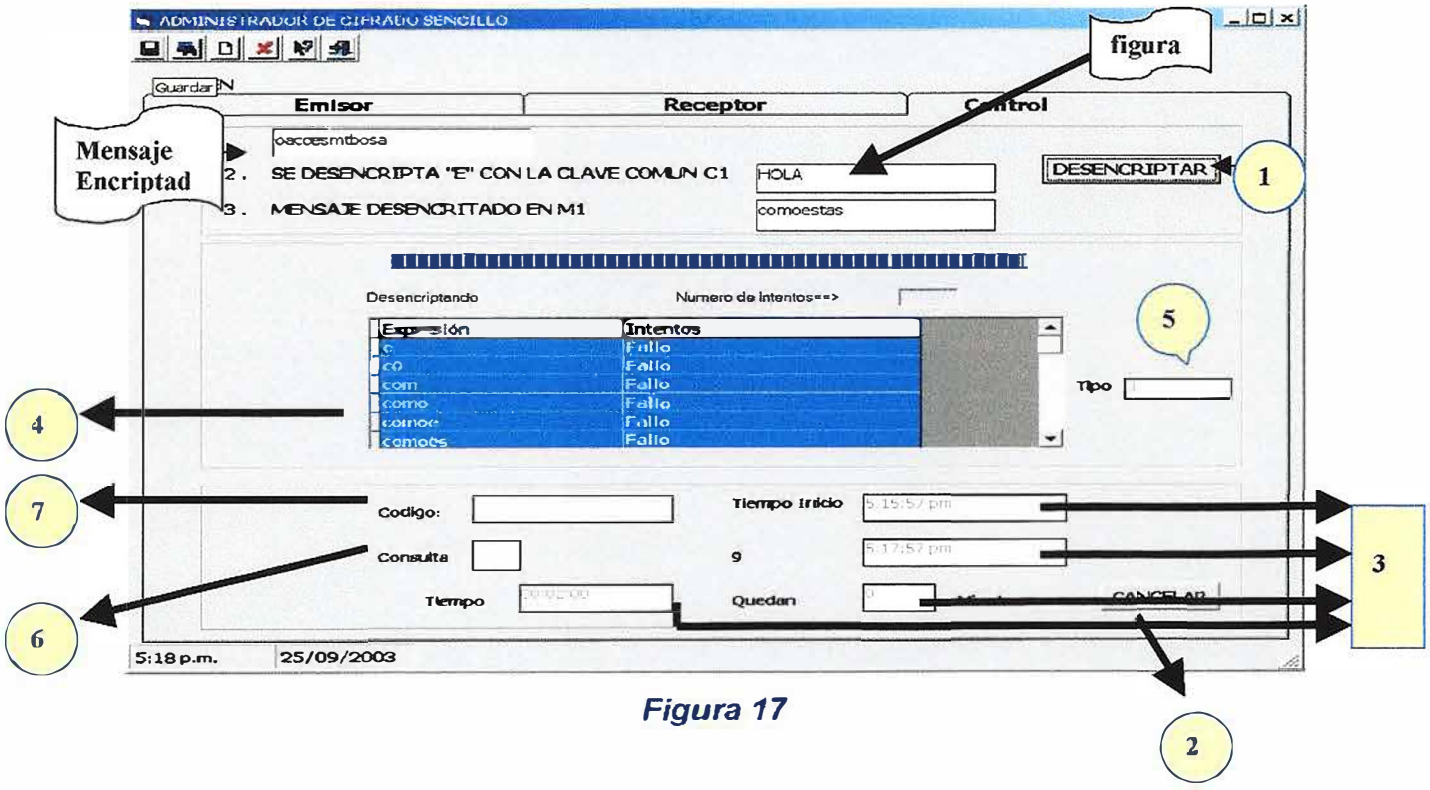


Figura 17

En la figura que nos Indica la Clave nos muestra la misma Información por que estamos tratando con un algoritmo muy sencillo el cual trabaja con única Clave tanto el Emisor como el Receptor

1 *Este botón nos permite desencriptar la información.*

2 *Haga clic como se muestra con la figura anterior para interrumpir al estar Guardando la información ya desencriptada.*

3 *Nos indica el Tiempo en que inicia el Control de Encripciòn a Desencriptar la Información, tomando el tiempo en que inicia y el final.*

4 *Al desencriptar la información él nos muestra una a una los datos que se van Encontrando.*

5 *Nos indica con que tipo de algoritmo estamos trabajando.*

6 *Información almacenada en la Base de Datos.*

7 *Nos permite consultar la Información deseada desde el mismo formulario.*

2.1.5. REPORTE GENERAL

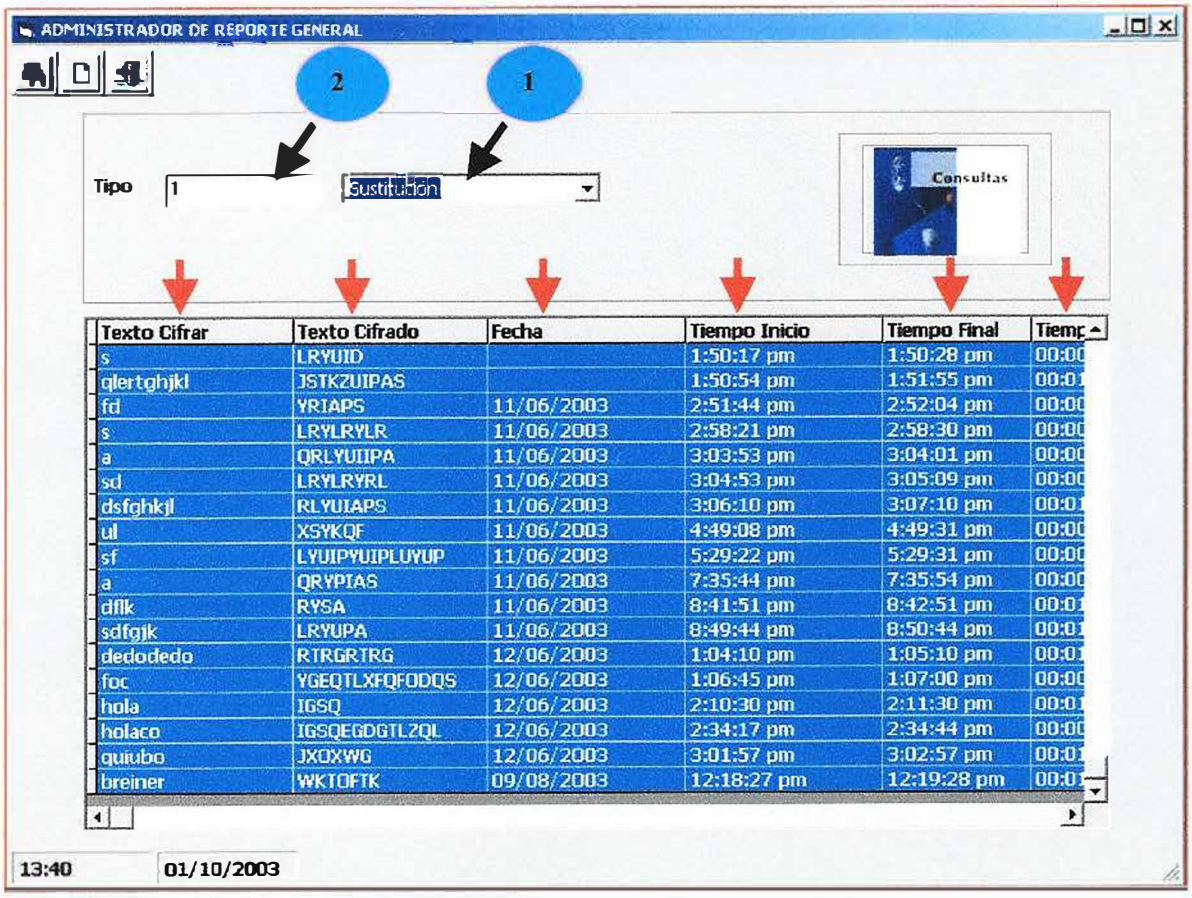


Figura 18

Al ingresar a este Formulario de Reporte General es porque tenemos la necesidad de consultar la Información que esta la Base de Datos, para obtenerla Presionamos donde nos indica 1 *y nos saldrá una serie de opciones por Ejemplo:*

[S] Consultas
[k] Reporte General
[#] id 2.1.5

Figura 19

Escogemos la opción requerida para obtener la Información como lo muestra anteriormente 3 *, y así sucesivamente para adquirir la información de todos los formularios.*

2 *Esta caja de texto nos indica que tipo de algoritmo es por medio de un Código, como esta en la Base de Datos.*

→ *Esta serie de Flechas nos muestra una a una las instrucciones que maneja este formulario por Ejemplo:* ***El Tiempo Inicio****, que nos indica el tiempo en que empezó la ejecución.* ***El Tiempo Final****, que seria el tiempo en que finalizo,* ***Fecha****, nos muestra la fecha en que fue ingresada la información al sistema.* ***Números de Intentos****, que seria cuantos intentos tuvo que hacer hasta encontrar la información.* ***Descripción de Intentos****, nos indica si la información fue exitosa dependiendo si en la ejecución no se interrumpió el proceso, de lo contrario nos mostrara que fue una ejecución No Exitosa.*

2.2. UREPORTE

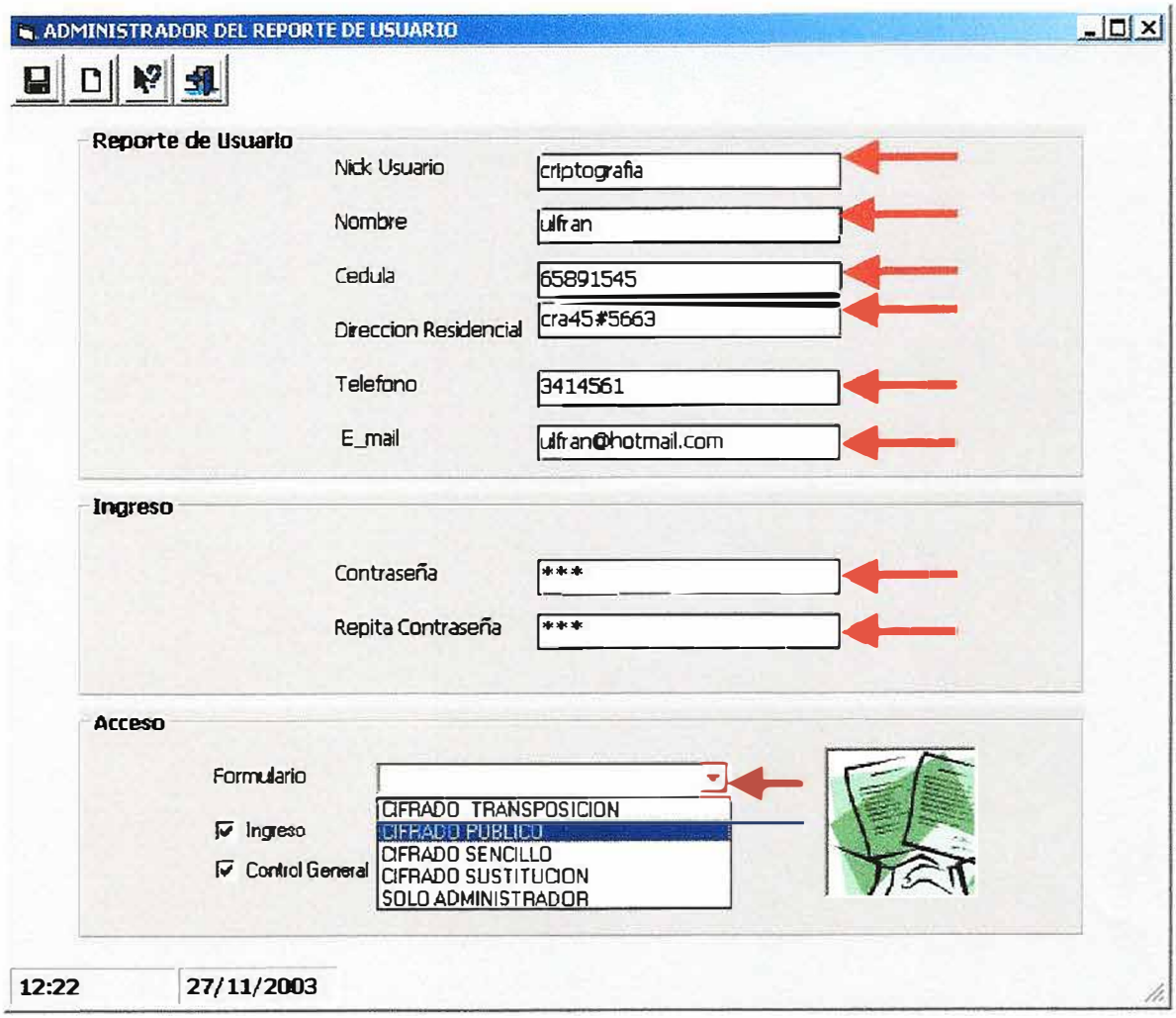


Figura 20

[$] Usuario
[k] UReporte
[#] id 2.2.

Este formulario es el registro del Usuario, donde el tiene que ingresar todos los datos que indica la flechita .
Luego debe pulsar Guardar.


Guardar.ico

Para almacenar la información en la Base de Datos, y así llevamos el control de quien ingresa al sistema para hacer más confiable la información confidencial.

El usuario que ingrese al sistema debe definir en que va trabajar y especificar si va ingresar información o va consultar, como lo muestra la flechita en el items de Acceso.

2.2.1 GENERADOR DE REPORTES

En éste formulario usted puede hacerle consultas a la base de datos y funciona de la siguiente manera:

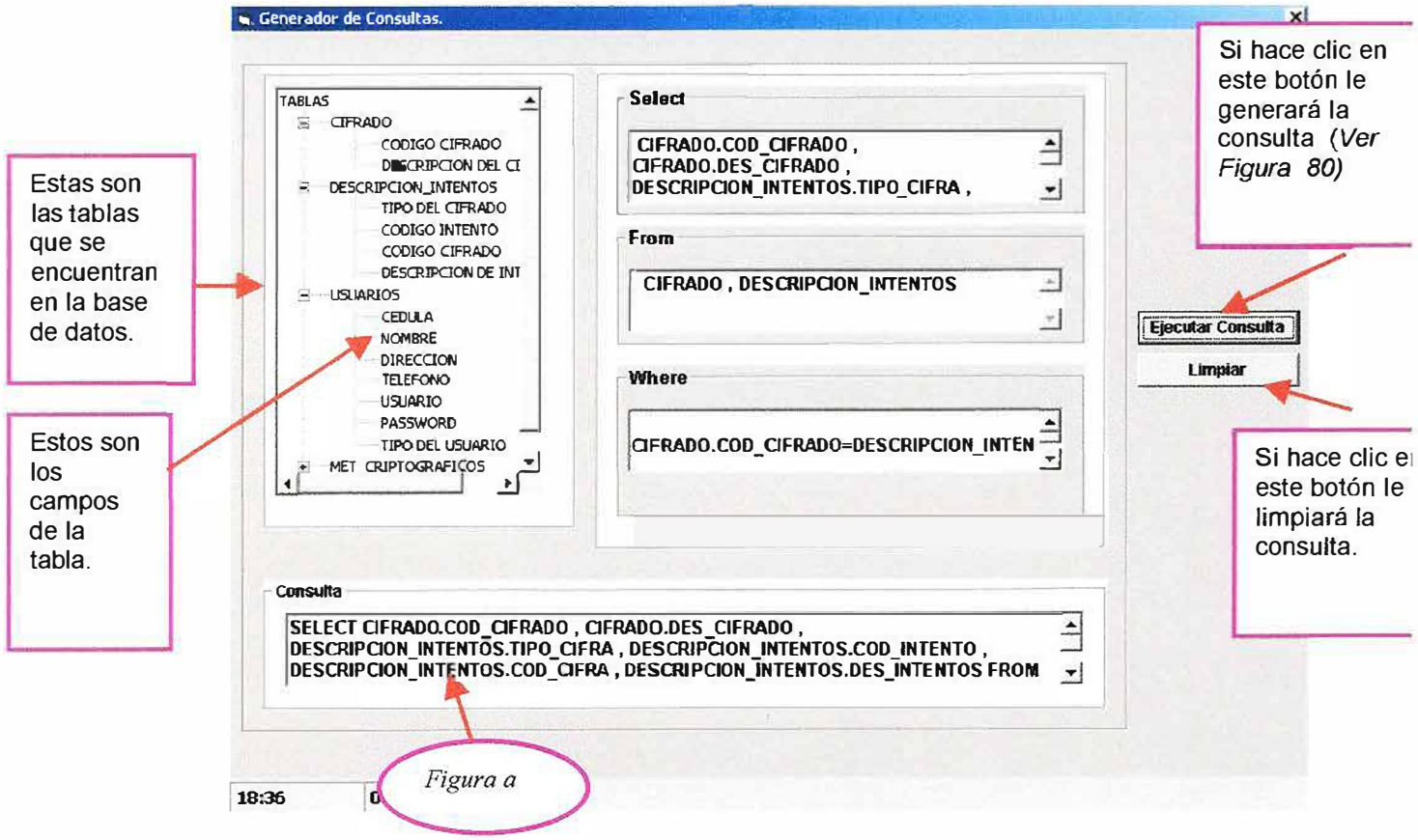


Figura 21

[$] Menú principal
[k] Generador de Reportes
[#] id 2.2.1

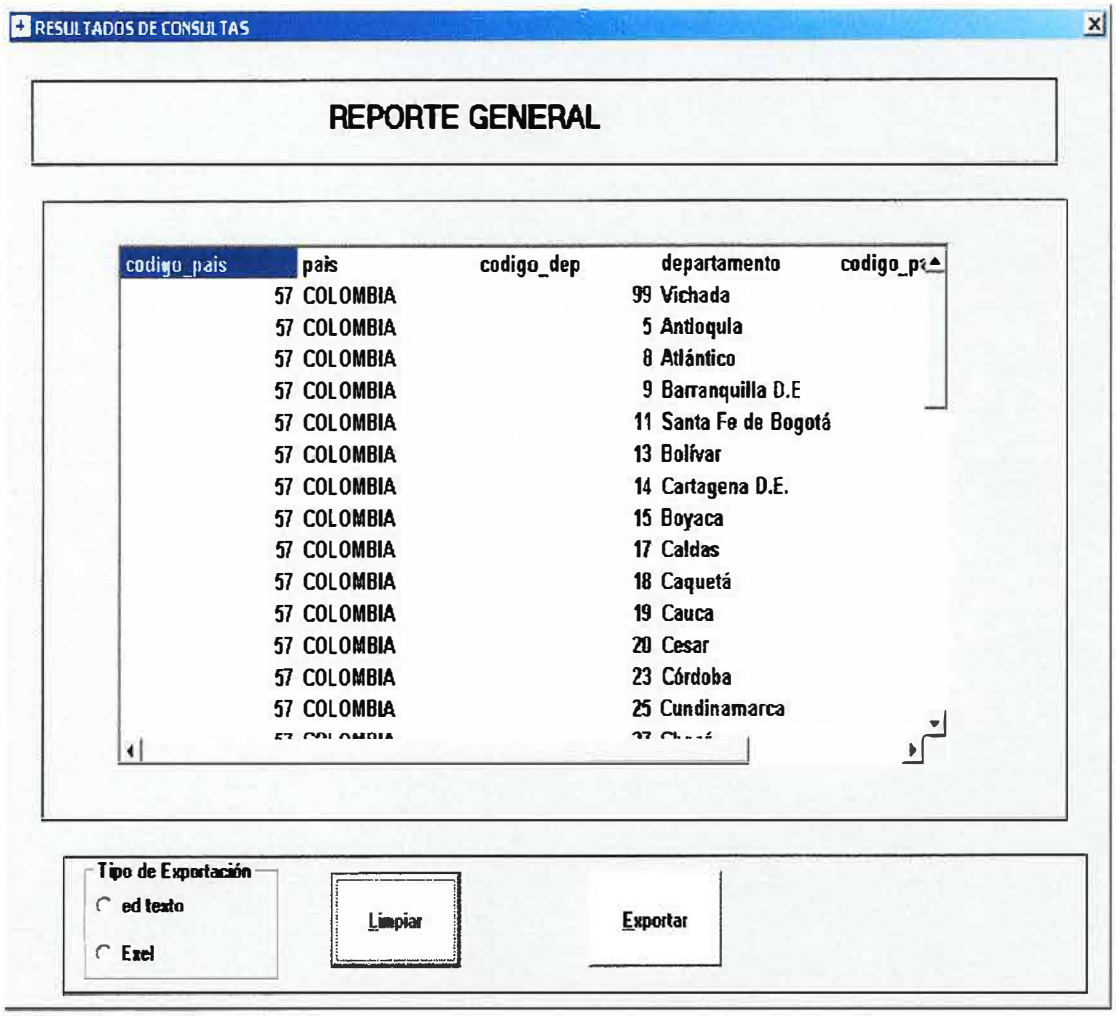


Figura 22

En éste formulario usted podrá observar la consulta que realizó y además tiene la opción de Exportar Datos; ya sea a un Archivo de Texto o a un Archivo de Excel.

Solo tiene que hacer clic en una de las dos opciones que aparecen en ***Tipo de Exportación*** *como por ejemplo* 

*Y después hacer clic en el botón **Exportar.***

Le aparecerá un cuadro de diálogo como el siguiente:

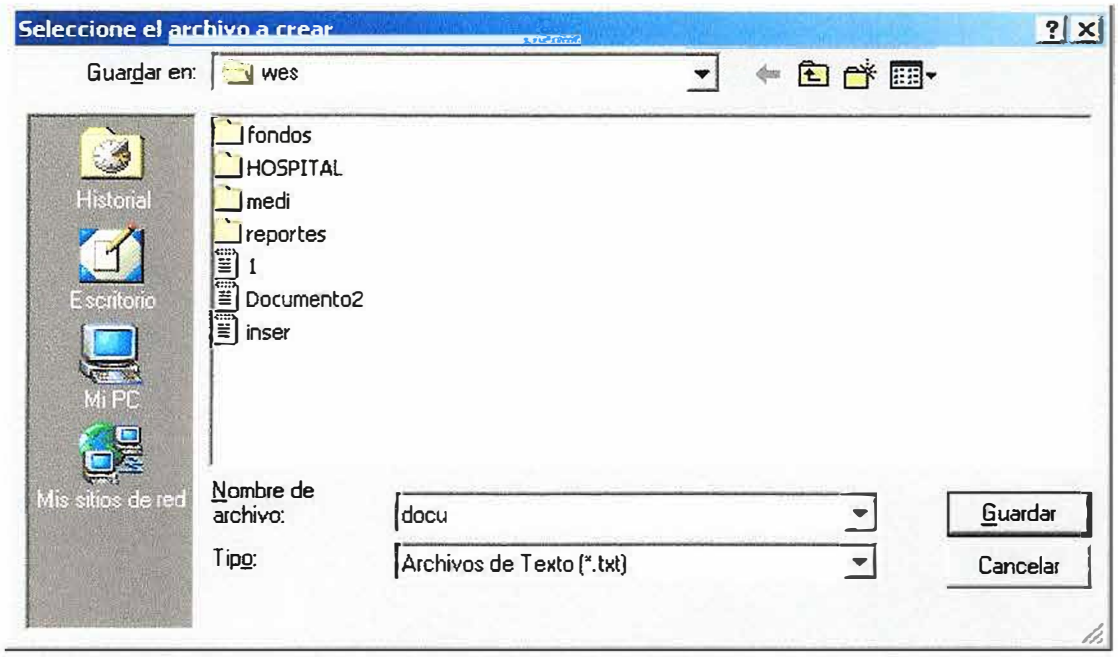


Figura 23

*En donde usted debe coloca el Nombre de Archivo al cual desea Exportarle los datos y luego hace clic en el botón **Guardar***

De inmediato le aparecerá el archivo de ésta manera:

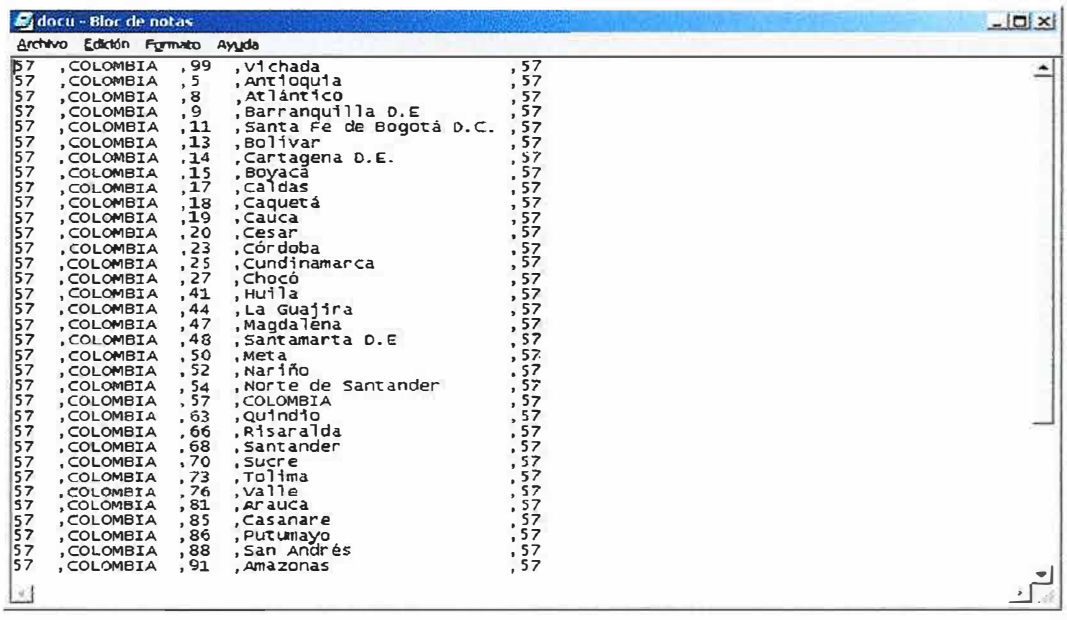


Figura 24

Sk#

2.2.2 IMPORTACION

Nuevamente en la Barra de Herramientas encontrará la opción de ***Herramientas*** *y puede escoger* ***Importación****. Como se muestra en la* ***Figura****.*

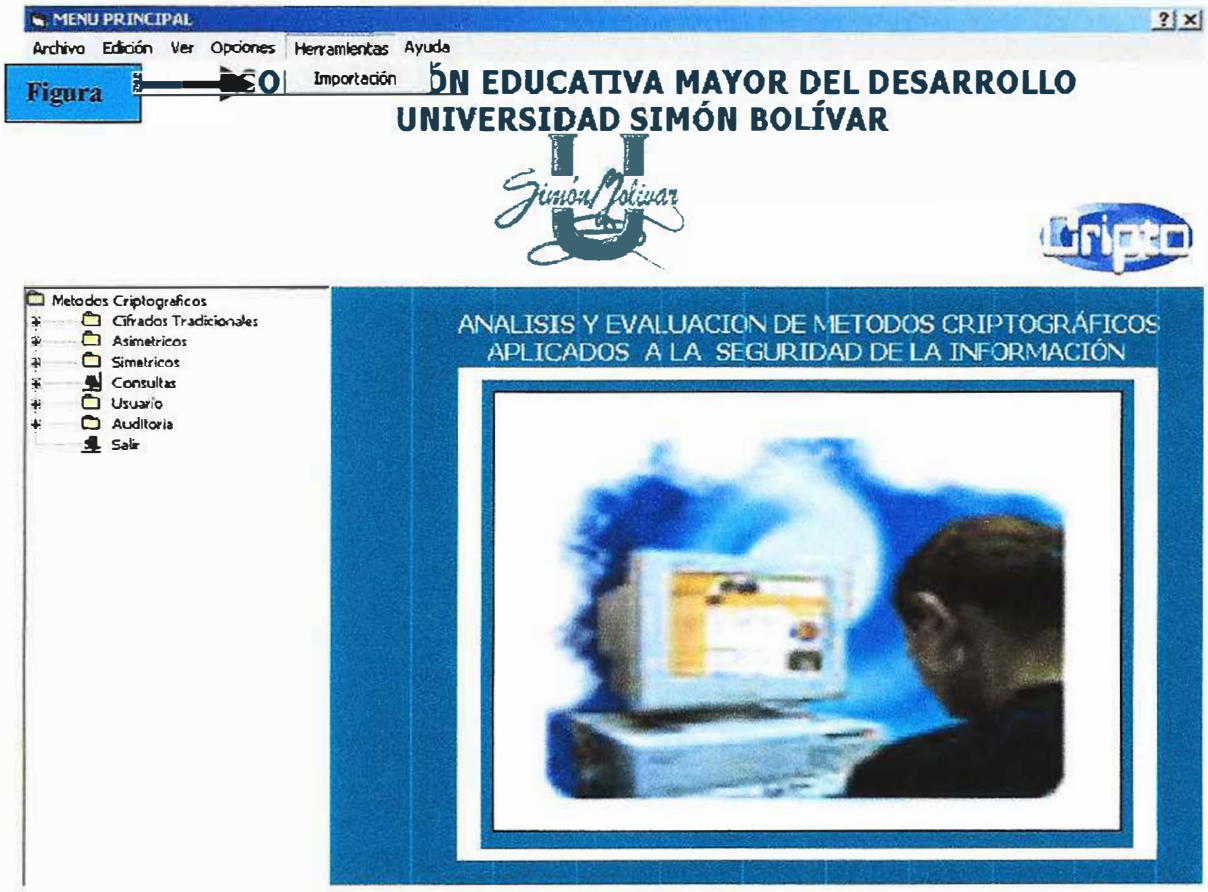


Figura 25

[$] Menú Principal
[k] Importar
[#] id 2.2.2

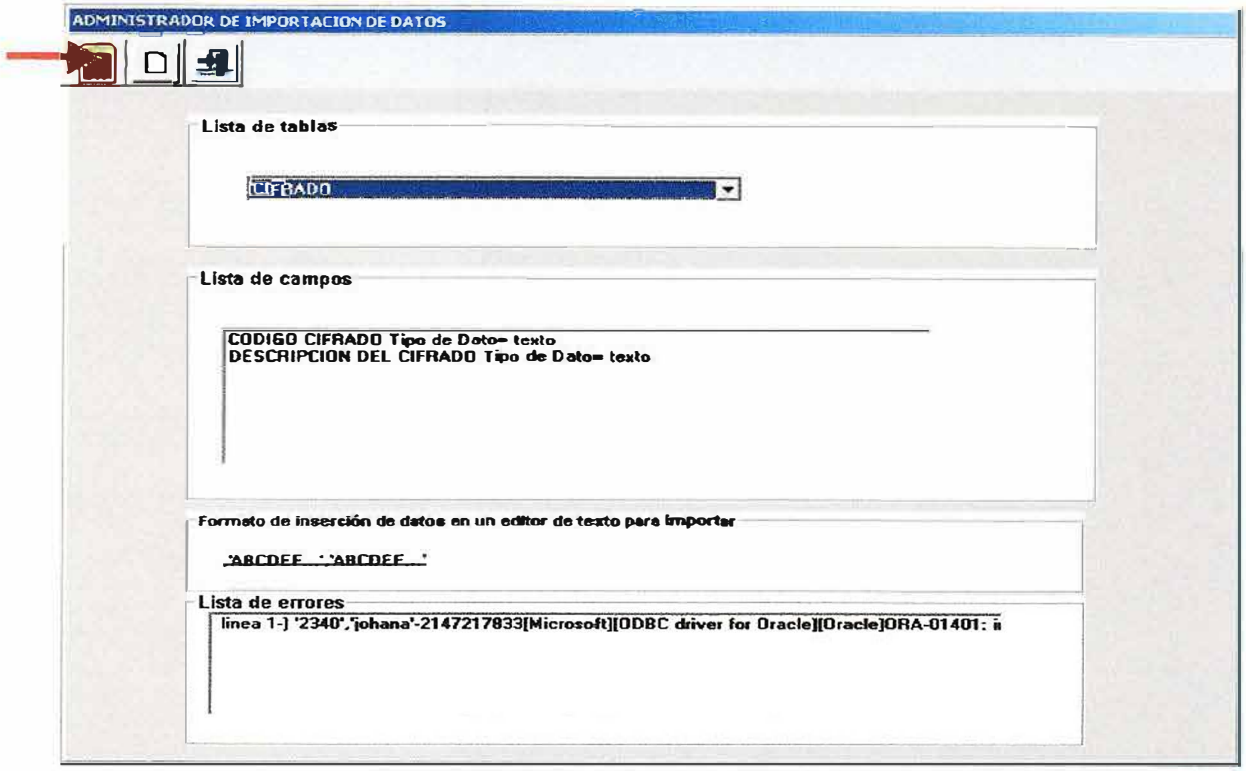


Figura 26

Por medio de éste formulario, usted podrá Importar Información a la Base de Datos. Funciona de la siguiente manera:

- *En **Lista de Tablas** seleccione la tabla a la cual desea hacerle la importación de datos.*
- *En **Lista de Campos** le va a mostrar todos los campos que tiene la tabla seleccionada.*
- *En **Formato de inserción de datos en un editor de texto para importar** le mostrará un ejemplo en donde le explica cual es la forma en que usted debe insertar los datos para el proceso de Importación.*
- *En **Lista de Errores** le visualizará los errores que tiene sus archivo.*

- *Hace clic en el botón **Importar***

Le aparecerá un Cuadro de Diálogo como el siguiente:

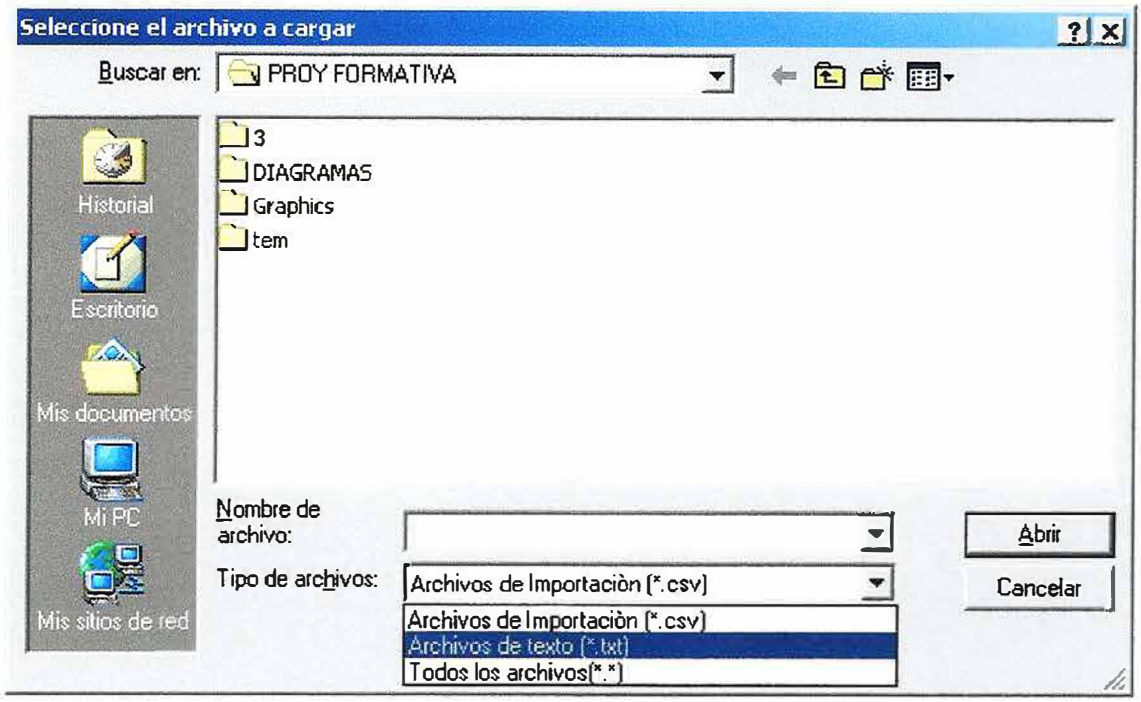


Figura 27

En donde usted escoge el archivo que desea cargar y le da clic en el botón ***Abrir****.*

Figura 28

Si su archivo no tiene ningún error le aparecerá un mensaje como el siguiente:

Pero si tiene error le saldrá un mensaje como éste:

Figura 29

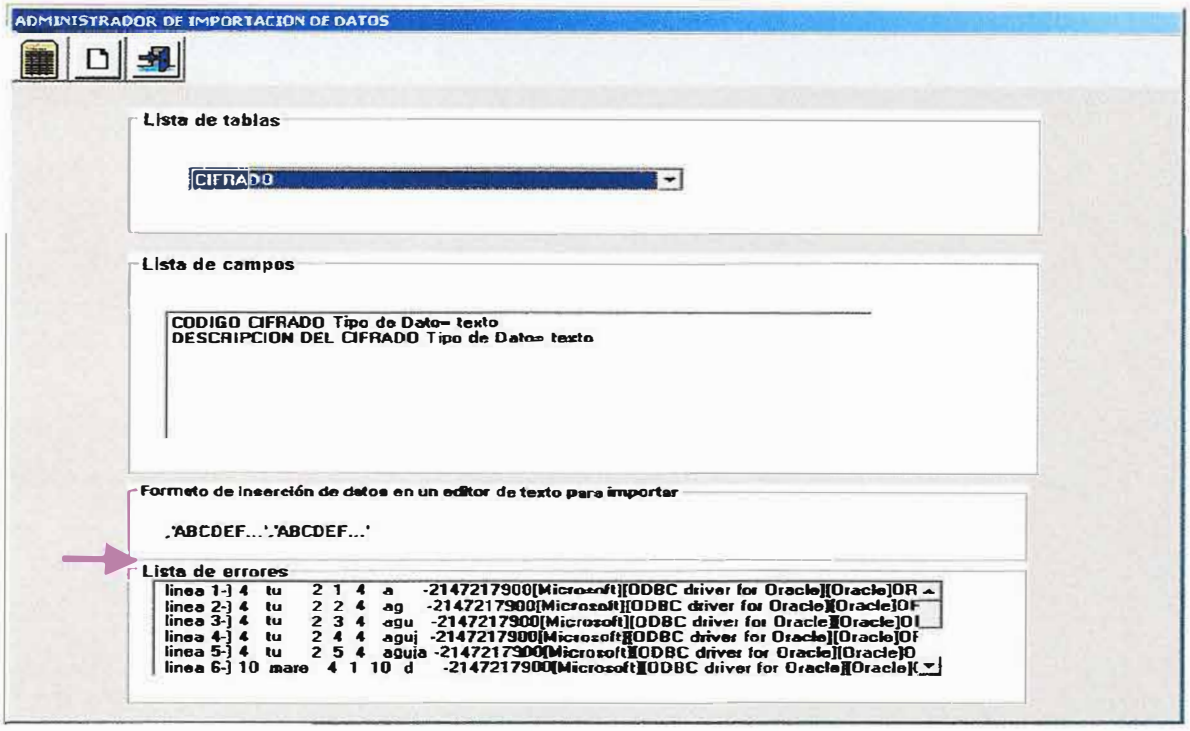


Figura 30

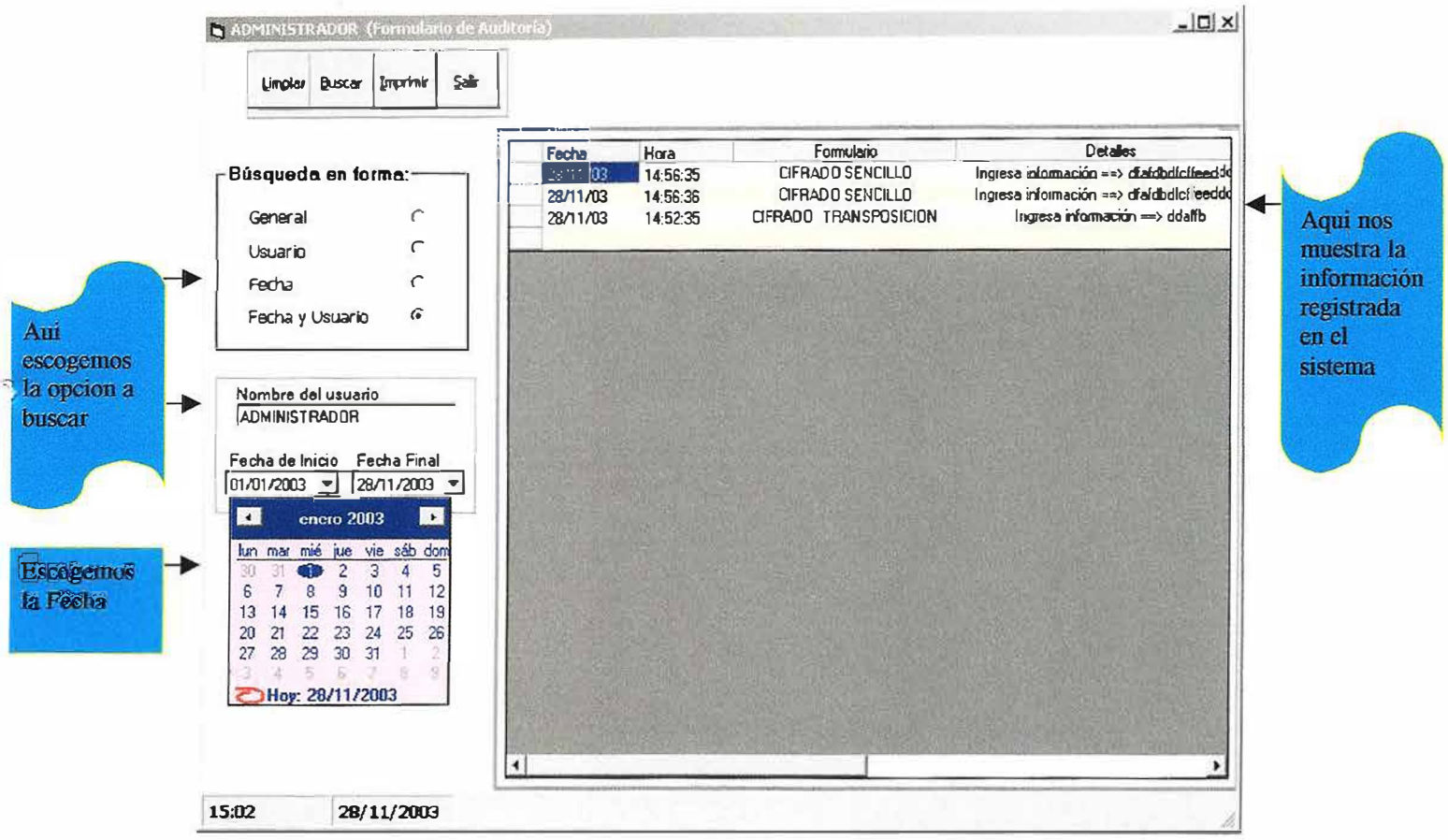
Y aquí mostrará cuales son los errores que tiene su archivo.

Este botón lo puede utilizar para limpiar los datos y hacer una nueva exportación.

Este botón se utiliza para salir del Formulario Importaciones.

$k#

2.2.3 AUDITORIA



En este Formulario puede el Administrador seleccionar cualquier opción de busqueda para Auditar y elegir que tipo de transacción quiere realizar que son:

General.

Fecha.

Usuario.

Fecha Usuario.

[s] Auditoria
[k] Adm. Auditoria

1. *Después de haber escogido ó seleccionado debe darle* ***Click*** *en el Botón* ***Buscar.*** *Para mostrar todos los datos.*

- *Al escoger la opción* ***General*** *solo tiene que pulsar* ***Buscar*** *y le mostrará el registro en general.*
- *Al escoger la opción* ***Usuarios*** *le aparecera una caja de texto donde deberá ingresar el usuario y luego pulsa* ***Buscar*** *y le mostrará el registro en general.*
- *Al escoger la opción* ***Fecha*** *se activará un calendario donde tengra que escoger la fecha de inicio y la fecha final al cual usted quiere que le muestre el registro pulsando* ***Buscar****.*
- *Al escoger la opción* ***Fecha Usuario*** *se activara la caja de texto para digitar el usuario al igual que el calendario para escoger la fecha y luego pulsan* ***Buscar*** *y muestra el registro.*

Para todas estas opciones al ser utilizadas y muestren información la describen de esta manera:

Fecha ⟶ *Nos indica la fecha en que ingresamos.*
Hora ⟶ *A que hora ingreso.*
Formularios ⟶ *A que formularios accedio.*
Detalles ⟶ *Nos muestra la información a la cual accedio el usuario.*
Usuario ⟶ *Que usuario ingreso si fue administrador o usuario.*
Equipo ⟶ *Nos indica en que equipo eatamos trabajando.*

Si Desea salir del Formulario le da ***Click*** *en le Botón* ***Salir.***

Salir

[#] id 2.2.3