APUESTAS POR MEDIO DE DISPOSITIVOS MÓVILES

DAVID FERNANDO ZAIDIZA

FUNDACIÓN UNIVERSITARIA SAN MARTÍN
FACULTAD DE INGENIERÍA
PROGRAMA DE ELECTRÓNICA Y TELECOMUNICACIONES
BOGOTÁ, COLOMBIA
2009, I

APUESTAS POR MEDIO DE DISPOSITIVOS MÓVILES

DAVID FERNANDO ZAIDIZA

031110

dzaidiza@hotmail.com

Asesor Técnico

ING. JORGE ARÉVALO

FUNDACIÓN UNIVERSITARIA SAN MARTÍN
FACULTAD DE INGENIERÍA
PROGRAMA DE ELECTRÓNICA Y TELECOMUNICACIONES
BOGOTÁ, COLOMBIA
2009, I

Nota de Aceptación:

_____

Jorge Arévalo
Asesor

_____

Diegos Días
Jurado

_____

Freud Romero
Jurado

Bogotá, Junio 11 de 2009

Párrafo de Dedicatoria

Dedico este proyecto de grado a todas aquellas personas que creyeron y confiaron en este reto. Aquellas que vieron como visión y ambición de conocimiento, a mis padres quienes estuvieron en la lucha por mi proyecto de vida.

AGRADECIMIENTOS

Durante el desarrollo de mi carrera he encontrado constates obstáculos, los cuales han sido superados gracias a la aparición de personas valiosas personas en mi camino hacia el éxito. La variedad de situaciones que te ponen a prueba permiten el forjamiento de un carácter y disciplina que será indispensable en el comienzo del vivir, es así que la mejor manera de agradecer a esas personas no es con palabras si no con acciones de logros.

Doy gracias a mi madre quien con sus manos labro un camino el cual seguir, mi familia quien creyó y apoyo la decisión de crecer en conocimiento, a mi padre quien me enseño el tener visón y en especial a esa persona que forjo mi persona, esa que mostro que la experiencia es importante, que el conocimiento se transmite y que puede ser vivido, gracias a mi abuela Orfelina.

Finalmente agradezco a mi asesor quien afronto el reto de tomar este proyecto y guiar mi ambición.

CONTENIDO

Pág.

LISTA DE FIGURAS

LISTA DE TABLAS

Pág.

LISTA DE ANEXOS

1. RESUMEN

Las comunicaciones nos han brindado gran variedad de herramientas, con las cuales se han facilitado los medios para poder apostar. En el desarrollo del proyecto de grado se han planteado las apuestas por medio de dispositivos móviles, es así que la seguridad es un requisito indispensable en la implementación de una aplicación prototipo con la cual se realizará una puesta prototipo que será enviada a través de la red celular de manera segura.

El envió de datos seguros se hará por medio de una combinación de algoritmos de cifrado, para esto primero se debe de conocer los algoritmos de cifrados existente y además saber cuáles de estos pueden ser soportados por dispositivos móviles.

El desarrollo de la aplicación prototipo se realizó en J2ME (Java 2 Micro Edition) que es un componente orientado al desarrollo de aplicaciones Java para dispositivos con pocos recursos como memoria, limitaciones de pantalla y capacidad de procesamiento.

Para el desarrollo del servidor, es necesario implementar el servidor WEB que permitirá realizar las validaciones de un usuario y contraseña. El servidor trabaja con Java en entornos web y a su vez es un contenedor de Servlets. Los Servlets son el aplicativo Java que se desarrolla para el manejo de información y seguridad en el servidor WEB.

El envió de información segura se podrá llevar a cabo por medio de certificados, estos permiten el envió de información segura por medio de la comunicación WAP, que es un entorno donde se encuentran implementados una variedad de protocolos que permiten la comunicación entre una terminal que envía datos por un canal con un ancho de banda limitado y un servidor WEB.

2. INTRODUCCIÓN

Las apuestas de chance son un juego cotidiano del pueblo colombiano. Los diferentes modos de juego de chance se han vinculado a la tecnología, dando como resultado las apuestas en red. Los dispositivos de apuesta aun son de gran tamaño, el más conocido de estos es el datafono.

La red celular brinda una gran opción para transportar datos, es así que el dispositivo celular presentas características que facilitan la creación de aplicaciones que permitan el envió de datos.

El desarrollo del proyecto de grado es dirigido a los expendedores de chance, con el fin de no afectar su estado laboral, proporcionar una herramienta sencilla, compacta, menos costosa, portable y segura. Otro aspecto importante en el desarrollo de un mercado es a quien está dirigido el servicio, en este caso a una población especifica a nivel del dispositivo, es decir a personas con conocimientos de apuestas, facilitando la enseñanza de la características del aplicativo.

Al desarrollar el proyecto de grado se plantea como problema y objetivo general el envió de datos cifrados por medio de dispositivos móviles. Como solución se desarrolla una aplicación prototipo que permita la realización de una apuesta por medio de un dispositivo móvil enviando datos cifrados a través de la red celular.

Al escoger el algoritmo adecuado de cifrado se exploran los algoritmos de cifrado existentes, además de cuales algoritmos pueden ser soportados en dispositivos móviles. Estos algoritmos pueden variar de dispositivo en dispositivo, es así que el algoritmo escogido además de garantizar el envió de datos de manera segura debe ser soportado por un equipo celular móvil.

Al escoger el dispositivo móvil celular en el cual se implementa la aplicación se tienen en cuenta las características técnicas del dispositivo, tales como sistema operativo, procesador, los algoritmos de seguridad que pueden soportar el dispositivo escogido, el paquete de desarrollo SDK (Software Development Kit - Kit de Desarrollo de Software) que son un conjunto de herramientas y programas de desarrollo que permiten al programador crear sus propios aplicativos para un determinado sistema operativo o paquete de software.

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Implementar una aplicación prototipo que permita el envió de datos cifrados por medio de dispositivos móviles.

3.2 OBJETIVOS ESPECÍFICOS

- Reconocer los algoritmos de cifrados disponibles
- Identificar cuales algoritmos de cifrado cuentan con las características para ser ejecutados en el dispositivo móvil
- Determinar el dispositivo adecuado para la implementación de la aplicación J2ME.
- Determinar el algoritmo de cifrado adecuado para un dispositivo móvil que cuente con transmisión vía celular para ser usado en juegos de apuestas de chance.
- Desarrollar una aplicación prototipo J2ME que permita realizar apuestas de chance en línea a través de un dispositivo móvil.

4. MARCO REFERENCIAL

4.1 ANTECEDENTES

Las apuestas de chance nacen en el departamento de Antioquia en los años 60 del siglo XIX. El impacto que se produjo fue de manera positiva. Tanto así que el juego se expandió por el territorio nacional. El congreso de la república respaldo el juego mediante la ley 1ª de 1982, esta ley se ha modificado con el fin de lograr pautas reglamentadas para el desarrollo del juego.

La evolución de la forma de apuesta de chance ha avanzado desde la tradicional tiquetera hasta los sistemas robustos de apuestas por medios móviles y redes de datos.

Las apuestas por tiquetera aun se llevan a cabo. Estas se realizan por medio de un expendedor de chance quien se encarga de elaborar el respectivo chance en un tiquete que contiene la información requerida para la apuesta. Al usuario se le hace entrega de una copia del correspondiente chance elaborado al igual que a la empresa que gestiona y regula el trámite de la apuesta. La responsabilidad de la validez del chance jugado por el usuario recae sobre el expendedor de chance, dado que si este no reporta la respectiva copia a la entidad el chance será invalido.

La evolución del chance va de la mano con la tecnología. La necesidad de expandir mercados y dar mayores facilidades permite el desarrollo de dispositivos que permitan apostar remotamente, evitando que el expendedor de chance se dirija a las empresas expendedoras de chance.

Nacen las máquinas Baloto que permiten realizar una variedad de apuestas en línea. Este dispositivo es costoso y de gran tamaño no permitiendo desplazarlo. La confirmación que arroja la máquina de apuesta es un recibo que contiene la información de la apuesta y los datos que el usuario necesita para conocer que su apuesta es totalmente legal.

La necesidad de movilizarse a menor costo obliga el desarrollo de equipos móviles que se basen en tecnología y red celular. Es así que nacen los datafonos cuyas características oscilan entre un diseño compacto, tecnología móvil, transacciones seguras y sin errores entre otras funciones. Su costo es más reducido que el de una máquina Baloto aunque su valor sigue siendo elevado, estando en un rango de 500 a 550 dólares.

Finalmente en la actualidad nace una tendencia a migrar al dispositivo más consumido en el mundo; el celular, ya que este permite una cobertura casi total.

En España se han implementado las apuestas celulares por medio de la compañía Movistar empleando la tecnología i-mode que es similar al protocolo de comunicaciones WAP, i-mode es utilizada en las apuestas Japonesas ofreciendo acceso a la red celular desde móviles.

4.2 MARCO TEÓRICO

Las comunicaciones y diferentes tecnologías han permitido el progreso en la necesidad de comunicarse de las personas. Avanzando a través de la etapas en la historia; comunicación oral, comunicación escrita, la imprenta que fue creada por Johannes Gutenberg cuyo invento cambio el transcurso de la historia, ya que permitió compartir la información entre el mundo. La evolución de las técnicas de comunicación nos lleva hasta los tiempos actuales donde la transmisión de datos se realiza de manera electrónica en el área de las telecomunicaciones.

La necesidad del ser humano por obtener conocimiento lo ha llevado a la revolución de la información, siendo así un ser totalmente dependiente de la información. La tecnología ha permitido estrechar las distancias, el desarrollo de nuevas entretenciones y la adquisición de información de manera móvil.

En toda comunicación es indispensable poseer un emisor, quien es el autor del mensaje, un código en cual se desarrolla el mensaje, un canal por el cual se envía dicho mensaje y finalmente una decodificación la cual consiste en la interpretación de los símbolos, signos y señales enviadas por el emisor.

La posibilidad de comunicarse e interpretar a través de la red permite el desarrollo de herramientas, proyecciones a futuro, entretenimiento y acceso a la información. El ser humano siempre ha querido visualizar el futuro, esto es uno de los factores que hace más fácil la creación de juegos de azar. En muchos países no son permitidas las apuestas, con Internet se brindan la posibilidad de acceso a una apuesta a aquellas personas que tengan limitaciones para realizar una; ya sea por leyes o por no tener un punto de acceso cercano.

4.2.1 Tecnología inalámbrica [MICROTEC]

El término inalámbrico se define como el uso de la tecnología sin cables la cual permite la comunicación entre varias terminales. La tecnología inalámbrica permite la creación de redes de datos en lugares donde la instalación de redes alámbricas es limitada. La cobertura inalámbrica es proporcional a la necesidad de alcance.

La accesibilidad de la tecnología inalámbrica nos permite el desarrollo de aplicaciones portátiles proporcionando comunicación desde cualquier dispositivo móvil como es el caso de los celulares, PDAs, palm entre otras.

La comunicación inalámbrica es un modelo de negocio que apenas está surgiendo, con herramientas que facilitan a los usuarios realizar trámites y obtener servicios que se adapten a sus necesidades. La tecnología inalámbrica ya

hace parte de la sociedad, tecnologías como el WIFI brindan a las empresas un ahorro significativo, a nivel estructural, físico y económico.

4.2.2 Jerarquía de las tecnologías inalámbricas

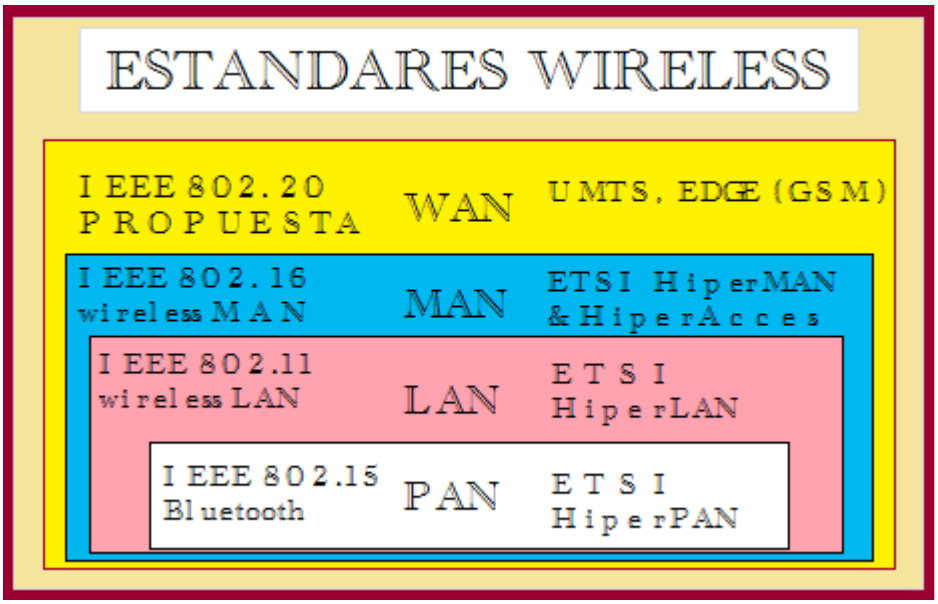


Figura. 1 Estándares de las redes Wireless **[TECI2008]**

- **Redes de Área Local**

Es utilizada para la conexión de varias terminales entre si con un servidor en común y una línea de comunicación dentro de un área geográfica limitada, aquí podemos encontrar diferentes redes como Wireless Ethernet, HIPERLAN que es un estándar global para anchos de banda inalámbricos LAN que funciona en la frecuencia de banda de 5Ghz.

Implementar una red LAN a nivel de hardware es costoso, pero proporciona un sistema eficaz para que un grupo de usuarios puedan compartir datos, además de permitir la comunicación electrónica entre sí.

Entre las redes de área local se encuentra la red WLAN (Red de área local sin cables) esta ofrece acceso sin cables a todos los recursos y servicios de una red corporativa (LAN). Estas redes proporcionan mayor libertad, flexibilidad al acceso de la información dentro de un área geográfica determinada, incluye el acceso en tiempo real.

- **Red de Área Personal [TECI2008]**

Se utiliza en la conexión de dos o más dispositivos, dentro de un área relativamente pequeña. Permitiendo así la conexión de impresoras, escáner, aparatos de fax, PDAs y ordenadores Notebook a equipos de escritorio, sin la necesidad de cables ni conectores para que sea efectivo el flujo de información. El objeto de esta comunicación es proporcionar una forma más cómoda de conexión ya que se tenían una variedad de puertos incompatibles como los USB, serial o paralelo. Estos sistemas presentaban ciertas limitaciones y la fiabilidad no era total.

El estándar de comunicaciones sin cables WPAN se centra en temas como el bajo consumo (para alargar la vida de los dispositivos portátiles), tamaño pequeño (para que sean más fáciles de llevar) y costos bajos (para que los productos puedan llegar a ser de uso masivo) es así que entre de los estándares más utilizados en el mundo se encuentra el Bluetooth (802.15).

Estas redes son limitadas por el área geográfica, hay dispositivos para estas redes que ofrecen mayor cobertura como es el caso del Bluetooth de segunda generación o Zigbee.

- **Redes de Área Extensa (WAN) [TECI2008]**

Se utiliza para el servicio de tecnología móvil dentro de este estándar se encuentran las redes wireless, ATM, UMTS.

Los dispositivos celulares al inicio solo ofrecían comunicación por voz. La evolución de la tecnología móvil ha llevado a que los celulares posean nuevas aplicaciones con interfaces y servicios inteligentes.

Esta red brinda una cobertura en un área geografía determinada, se subdivide en zonas más pequeñas llamadas celdas o células, cada una de estas corresponde a una zona cubierta por una estación radio base de baja potencia. Operando a frecuencias de radio individuales, estas frecuencias se repiten en celdas no vecinas a la misma.

El cambio de celda se hace cuando la señal del móvil es más baja, transfiriéndose a donde esta es más potente. Al ser las frecuencias distintas entre las células no se produce la interferencia.

Las ventajas de estas tecnologías son que constan de mayor cobertura, menor costo, movilidad accesibilidad y comodidad, entre otros factores.

Las tecnologías Bluetooth, IEEE 802.11, HomeRF e HiperLAN utilizan frecuencias de radio para sustituir los cables. Bluetooth utiliza entre 100 y 1000 veces menos potencia que sistemas establecidos como 802.11, mientras que HomeRF tiene

una implementación más cara que Bluetooth. HiperLAN tiene limitaciones de velocidad en la transmisión de información.

En el siguiente diagrama se puede observar las diferencias tecnológicas entre las tecnologías WAN, LAN y PAN. Se observan características como las frecuencias, el alcance que esta proporciona y las velocidades que manejan.

Tecnología	Aplicaciones	Frecuencia	Alcance	Velocidad
Bluetooth	WAN/LAN/PAN	2,45 GHz	Hasta 30 m	1 Mbps
IEEE802.11	WAN	2,45 GHz	Hasta 100 m	2Mbps
IEEE802.11b	WAN	2,45 GHz	Hasta 100 m	2 Mbps – 11 Mbps
IrDA	PAN	N/A	Hasta 8 m	4 Mbps
HomeRF	LAN	2,45 GHz	Hasta 30 m	1,6 Mbps
HiperLAN	LAN	5 GHz	TBA	55 Mbps

Tabla. 1 Tabla comparativa de redes

- **Diferencias entre tecnología móvil e inalámbrica [TECI2008]**

La tecnología móvil implica poder trasladar cierta cantidad de tareas de un sitio a otro mientras que la tecnología inalámbrica hace énfasis en conectar varios dispositivos entre si o a una red sin cables.

4.2.3 Tecnologías de acceso a celular

En el entorno de las comunicaciones se encuentra la tecnología móvil cuyo objetivo es la transmisión de datos de un punto a otro sin cables. La telefonía celular logra su comunicación por medio de las tecnologías de acceso a celular GSM, CDMA, TDMA y FDMA.

- **GSM (Global System for Mobile Communications) [TECH 2003]**

GSM Global Systems for Mobile Communications - Sistema Global de Comunicaciones Móviles, es un sistema digital de comunicación que transmite voz y datos. Este es considerado como la Segunda Generación (2G) ya que a diferencia de la primera generación de celulares, utiliza tecnología digital y la división de acceso de transmisión múltiple (TDMA). GSM digitaliza, comprime la información y luego divide cada canal de 200MHZ en ocho espacios de tiempo de 25MHZ. Este sistema opera en las bandas 900MHZ y 1800MHZ en Europa, África y Asia, en las bandas 850MHZ y 1900MHZ en Estados Unidos. La banda 850MHZ también se utiliza para GSM y 3GSM en Canadá, Australia y en varios países de Latinoamérica.

- **CDMA (Code Division Multiple Access) [UPVS2006]**

CDMA (Code Division Multiple Access), basada en el estándar IS-95 CDMA, para proporcionar servicios de alta calidad en fax, datos y voz. Este estándar parte de los métodos de transmisión digital patentados por QUALCOMM en los cuales los usuarios comparten tanto tiempo como frecuencia

Para diferenciar a los distintos usuarios, en lugar de frecuencias separadas se usan códigos digitales únicos. Los códigos son conocidos tanto por la estación móvil (teléfono celular) como por la estación base, y se llaman "Secuencias de Código Pseudo-Aleatorio". Por lo tanto todos los usuarios comparten el mismo rango del espectro radioeléctrico.

- **TDMA (Time Division Multiple Access) [TDMA2008]**

(Time Division Multiple Access). TDMA es una tecnología inalámbrica de segunda generación, que distribuye las unidades de información en ranuras alternas de tiempo, dando acceso múltiple a un número reducido de frecuencias. TDMA permite dar servicios de alta calidad de voz y datos.

TDMA divide un canal de frecuencia de radio en varias ranuras de tiempo (seis en D-AMPS y PCS, ocho en GSM). A cada usuario que realiza una llamada se le asigna una ranura de tiempo específica para permitir la transmisión. Esto permite que múltiples usuarios utilicen un mismo canal de frecuencia al mismo tiempo sin interferirse entre sí.

- **FDMA (Frequency Division Multiple Access) [FDMA2008]**

Acceso múltiple por división de frecuencia o FDMA, el espectro radioeléctrico se divide en secciones o ranuras de frecuencias asignadas a cada una de ellas. La configuración es rígida e invariable, pues cada estación debe transmitir siempre con la misma frecuencia central o portadora, y es válida cuando se puede garantizar que, durante la mayor parte del tiempo, cada una de ellas ocupará activo ese ancho de banda que se le asignó; por esta razón, también se le llama acceso múltiple con división de frecuencia con asignación fija. Es claro que su utilización radica principalmente en sistemas comerciales de alta capacidad.

4.2.4 Estándares para navegación

- XHTML - WAP 2.0 (Estándar Actual de las páginas Web)
- I-Mode (Utilizado en España, es soportado por ciertos móviles)
- W3C (Es un estándar que facilita la navegación en dispositivos móviles)
- OMTP (Open Mobile Terminal Platform)

4.2.5 Herramientas requeridas en el desarrollo de la aplicación

- **WML (Wireless Markup Language)[WMLCL]**

WML es un lenguaje muy similar al HTML, esta compuesto a base de etiquetas donde se puede incluir imágenes y formularios. Las extensiones de las páginas creadas en WML son *.wml*. Adicionalmente los gráficos son en formato wbmp.

- **Servlet** [ACM2007] [JAVSERVL] [TECSERVL]

Un servlet es muy similar a un applet de Java, cuya diferencia consiste en que este se ejecuta en el servidor web, en vez de ser descargados a la web del cliente. Una definición más exacta es que es una plataforma Java cuyo fin es la ampliación y el mejoramiento de los servidores web. Estos proporcionan un componente base que permite la construcción de aplicaciones basadas en web.

Estos son útiles para la construcción de páginas web basadas en datos enviados por el usuario, datos que cambian constantemente como estadísticas y resultados y finalmente paginas donde se accede a información almacenada en bases de datos corporativas.

Figura. 2 Arquitectura cliente servidor para la WEB

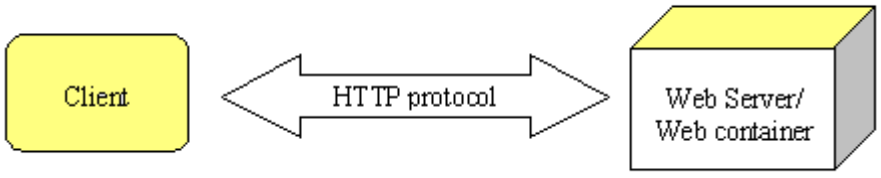


Figura. 3 Interacción cliente aplicación Servlet (Servidor)

En los diagramas se muestra que para la interacción entre el aplicativo servlet y la aplicación del móvil el protocolo es HTTP.

- **Tomcat**

Tomcat es un servidor Web que trabaja con Java en entornos Web, un contenedor de Servlets y JSP, se define como un Shell de ejecución que se encarga de tomar lo que se teclea y convertirlo en un programa ejecutable que maneja e invoca los Servlets que se utilicen por el usuario.

Dentro de los requisitos para la implementación de un servidor Tomcat se encuentran, tener instalada la máquina virtual de Java, jdk, j2sdk1.4, la versión de Tomcat a partir de la 4.1. Poseer un editor de Servlets, para el desarrollo de esta se utiliza el Netbeans.

- **Netbeans**

La plataforma Netbeans es un entorno de desarrollo integrado para aplicaciones Web, el cual es utilizado para editar programas en Java, ejecutarlo, compilarlo, depurarlo entre otras opciones.

Esta plataforma se utilizará para construir el aplicativo del móvil y en el desarrollo Servlets para el servidor Web, estos dos aplicativos serán utilizados para la conexión de los dispositivos.

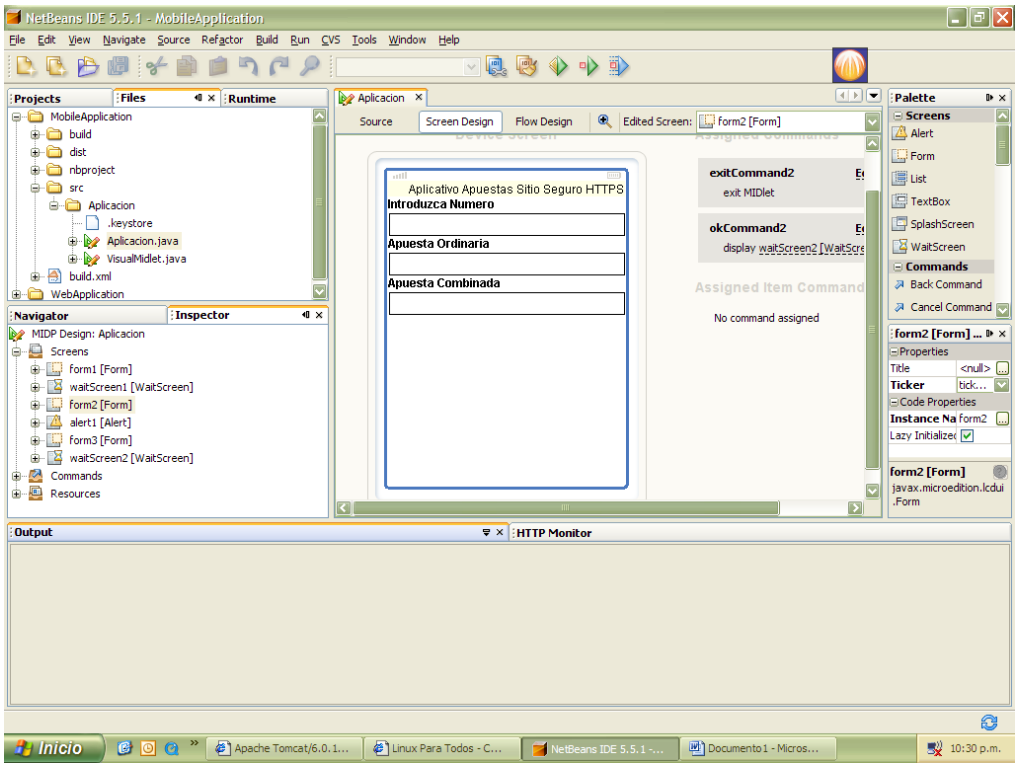


Figura. 4 Entorno de desarrollo Netbeans [NETMOV]

4.2.6 Plataforma JAVA 2 [J2ME2003]

La plataforma JAVA ha revolucionado los diferentes desarrollos de aplicaciones. Java proporciona ventajas en el entorno de programación de dispositivos inalámbricos, podemos encontrar ventajas claves en el uso de Java; *Seguro, Robusto, Portable.*

- Seguro, la Máquina Virtual de Java permite compatibilidad con otras aplicaciones, aun así las aplicaciones desarrolladas por Java solo se pueden colgar a la máquina virtual de Java, sin afectar ninguna de las otras aplicaciones del dispositivo.

- Robusto, el lenguaje de Java proporciona una plataforma de programación robusta en lo que respecta al consumo de recursos, la librería de objetos, el mecanismo de excepciones entre otros.

- Portable, la fortaleza más grande de Java es su portabilidad, debido a que permite escribir una aplicación y ejecutarla en distintos dispositivos, en un mundo inalámbrico hay gran variedad de aparatos y fabricantes. La portabilidad es importante en el mundo inalámbrico debido a que las aplicaciones se pueden distribuir a los dispositivos por medio de la red inalámbrica.

4.2.7 Estándares de JAVA 2[J2ME2003]

- J2SE - Java 2 Standard Edition, está orientado a ordenadores de escritorio. Comprende el JDK hasta ahora distribuido por Sun, posee clases adicionales que permiten el desarrollo de interfaces de usuarios de alta importancia.

- J2EE - Java 2 Enterprise Edition, abarca el J2SE y adiciona clases para el desarrollo en entornos corporativos. Esta edición está orientada al desarrollo de aplicaciones para servidores, al desarrollo de componentes y distribución de aplicaciones.
- J2ME - Java 2 Micro Edition, es un subcomponente del J2SE orientado al desarrollo de aplicaciones Java destinadas a dispositivos con pocos recursos, aplicaciones restringidas como la capacidad de memoria disponible, limitaciones de pantalla, capacidades de procesamiento y consumo.

Figura. 5 Dispositivos soportados por los estándares Java [JAVWIRE]

Figura. 6 Dispositivos soportados por los estándares Java [J2ME2003]

4.2.8 J2ME (Java 2 Micro Edición) [J2ME2003]

Es un componente de la edición estándar de Java orientada a ciertos dispositivos, estos de recursos reducidos. J2ME maneja un grupo de perfiles; entre esos se encuentra el MIDP (Movil Information Device Profile), este perfil es el adecuado para los teléfonos móviles.

La arquitectura de J2ME se basa en familias y categorías de dispositivos. Una categoría define el tipo de dispositivo; *celulares, busca personas, organizadores personales.* Una familia se compone de un grupo de categorías que poseen requisitos similares de memoria y capacidad de procesamiento.

La arquitectura J2ME está diseñada con escalabilidad y flexibilidad debido a que puede diseñarse nuevos dispositivos, así que la arquitectura J2ME deberá permitir adaptarse a ellos.

En la siguiente figura se observa la configuración básica constituida sobre el sistema operativo del dispositivo, esta consta de cuatro capas:

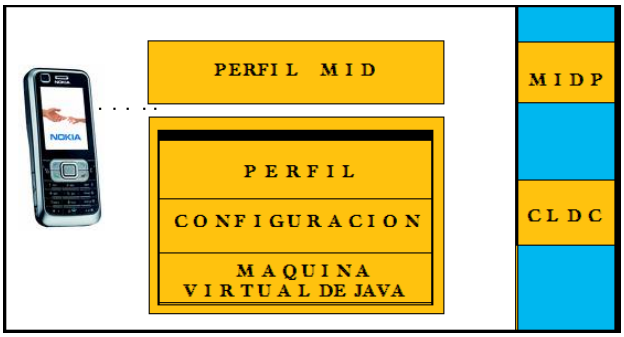


Figura. 7 Configuración básica del sistema operativo de un dispositivo móvil [J2ME2003]

La máquina virtual de Java se configura dependiendo de cada uno de los dispositivos, de forma que pueda adaptarse y soportar el sistema operativo que se ejecute en cada dispositivo.

La capa de configuración define el mínimo de características de la máquina virtual y las librerías Java que están disponibles para el conjunto de dispositivos. Esta capa es la más importante para el desarrollador de perfiles, porque define la parte común de las librerías y características de Java que estarán en el dispositivo dentro de una familia de dispositivos.

La capa de perfil, está orientada a la aplicación definiendo el mínimo conjunto de APIs disponibles para una determinada familia de dispositivos. Las librerías y clases son más específicas que la se pueden encontrar en la configuración.

La capa de perfil MID, consta de un conjunto de APIs Java que permite la creación de interfaz de usuario, conexión de red, manipulación de datos, sonido, seguridad etc. La unión de las tres primeras capas conforma la configuración del CLDC que junto a la capa MIDP (Perfil para Dispositivos de Información Móvil) forman el entorno estándar de ejecución para las aplicaciones y servicios que se pueden descargar dinámicamente sobre los dispositivos de los usuarios.

En la siguiente figura se observa la estructura de los distintos bloques de componentes en el entorno de J2ME:

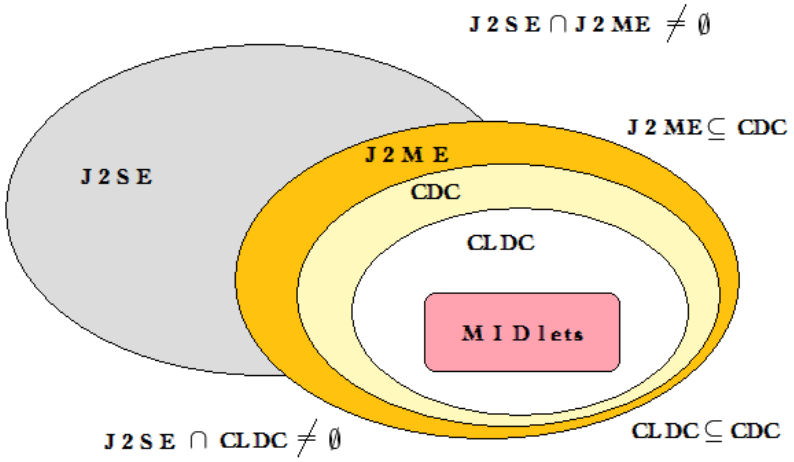


Figura. 8 Componentes del entorno J2ME [J2ME2003]

Una de las herramientas para el desarrollo de aplicaciones J2ME es el Eclipse 3.0, este permite la edición, compilación y simulación de la aplicación creada.

4.2.9 WAP (Wireless Application Protocol) [PAGW2007]

WAP es un estándar para comunicaciones inalámbricas que permite el acceso a Internet desde un dispositivo móvil.

Los dispositivos móviles cada vez son más robustos; más livianos y más potentes permitiendo a su vez un mayor rango de aplicaciones. Cuando se diseña un servicio por medio de la tecnología WAP se consideran ciertas características (Pantallas más pequeñas, teclados más limitados, limitaciones de memoria RAM, limitaciones de almacenamiento, limitaciones de procesador en comparación a un PC).

La versión más reciente de WAP (WAP 2.0) utiliza el lenguaje XHTML-MP (Movil profile), maneja color, maneja un mejor soporte de gráficos, en la capa de transporte se usa TCP y la de aplicación HTTP, puede tener funciones como cache Web.

4.2.10 Modelo de funcionamiento WAP

WAP define unos componentes estándares para poder conseguir una comunicación estable entre una terminal móvil y un servidor:

Figura. 9 Entorno WAP

Un modelo de nombres estándar, utilizando URIs (Identificar funciones como las de control de llamada) URLs para identificar el contenido WAP en los servidores de información.
Un formato de contenido estándar basado en tecnología WWW
Protocolos de comunicación estándares, que permita la comunicación entre el micro navegador de la terminal móvil con servidor WEB en red.

Tabla. 2 Requisitos para una comunicación estable WAP

4.2.11 Definición de Criptografía [UVES 1996]

Vista en términos sociales, es la ciencia de hacer que el coste de adquirir o alterar información de modo impropio sea mayor que el posible valor obtenido al hacerlo.

Vista en términos más formales, es la práctica y el estudio de técnicas de cifrado y descifrado de información, es decir, de técnicas para codificar un mensaje haciéndolo ininteligible (cifrar) y recuperar el mensaje original a partir de esa versión ininteligible (des cifrar).

Los algoritmos criptográficos se definen como los métodos matemáticos empleados para cifrar y descifrar un determinado mensaje. Funciona empleando claves que son números o cadenas de caracteres, que serán necesarias en la recuperación del mensaje.

Los sistemas criptográficos, están compuestos de algoritmos criptográficos, claves, y textos con el fin de cifrar y descifrar. Estos sistemas se basan en tres tipos de algoritmos; clave secreta o simétrica, clave pública o asimétrica y de resumen de mensajes.

Figura. 10 Diagrama de un sistema criptográfico

Se define a un algoritmo de cifrado o cifrar como aquel que pertenece a una familia de transformaciones invertibles, este obtiene representaciones de texto original sin sentido siendo controlado por una clave.

$$E_k : M \rightarrow C$$

- k: son la cantidad de claves
- M: Mensaje
- C: Criptogramas
- Ek: Mensaje

Representación del algoritmo de cifrado

$$E_k(m) = c$$

Si las claves k son diferentes los mensajes cifrados serán distintos.

$$E_{k_1}(m) \neq E_{k_2}(m) \text{ sii } k_1 \neq k_2.$$

Se considera descifrado al obtener un texto legible a partir de una representación sin sentido; el criptograma, haciendo uso de una clave.

Representación al algoritmo de descifrado

$$D_k = E_k^{-1}$$

$$D_k : C \rightarrow M$$

El proceso inverso retorna el mensaje

$$D_k(c) = m \text{ y } D_k(c) = D_k(E_k(m)) = m.$$

A la variable k se le considera como datos aleatorios que representan número con valor matemático, la clave para cifrar y descifrar no es necesario que sean la misma.

- **Algoritmos de resumen de mensajes**

Transforman mensajes de tamaño variable a textos cifrados de tamaño fijo sin emplear claves. Se emplean para convertir mensajes grandes en representaciones más manejables.

- **Algoritmos de claves públicas o asimétricas**

Estos cifran un mensaje generando un texto cifrado del mismo tamaño que el original. Usan una clave para cifrar el mensaje (clave privada) y otra para descifrar (clave pública). Tienen un coste computacional alto y se suelen emplear para distribuir las claves de los algoritmos simétricos.

- **Algoritmos de clave secreta o simétrica**

Convierten un mensaje en un texto cifrado del mismo tamaño que el original. Emplean una sola clave para cifrar y descifrar. Son los algoritmos empleados para transferir grandes cantidades de información de modo seguro.

- **Criptoanálisis**

Un criptoanálisis es el conjunto de procedimientos, procesos y métodos empleados para romper un *algoritmo criptográfico*, *descifrar* un *texto cifrado* o descubrir las *claves* empleadas para generarlo.

4.2.12 Algoritmos de cifrado simétrico [SCHNEI96]

- **DES [DESCSRC]**

DES (***Data Encription Standard* o *Estándar de Cifrado de Datos***) es un esquema de cifrar simétrico desarrollado en el año de 1977. Oficialmente el nombre del documento es FIPS (Federal Information Processing Standard) 46-1 del Instituto Nacional de Estándares y Tecnología (NIST) del Departamento de Comercio de Estados Unidos. En este documento se describe el DEA (*Data Encription Algorithm* o *Algoritmo de Cifrar de Datos*. Este es uno del algoritmo de cifrado simétrico más estudiado, mejor conocido y más empleado del mundo.

DEA se le conoce como DES, es un algoritmo de cifrado por bloques de 64 bits de tamaño. Emplea una clave de 56 bits durante la ejecución (se eliminan 8 bits de paridad del bloque de 64). El algoritmo fue diseñado para ser implementado en hardware. Cuando se utiliza en comunicaciones ambos participantes deben conocer la clave secreta (para intercambiarla se suelen emplear algoritmos de clave pública).

El algoritmo se puede usar para cifrar y descifrar mensajes, generar y verificar códigos de autentificación de mensajes (MAC) y para cifrar de un sólo usuario (p. ej para guardar un archivo en disco).

- **Triple-DES [DESCSRC]**

Consiste en cifrar tres veces una clave DES. Esto se puede hacer de varias maneras:

Figura. 11 Diagrama de bloques de triple DES [DESCSRC]

- DES-EEE3: Tres encriptaciones DES con tres claves distintas.
- DES-EDE3: Tres operaciones DES con la secuencia cifrar-descifrar-cifrar con tres claves diferentes.
- DES-EEE2 y DES-EDE2: Igual que los anteriores pero la primera y tercera operación emplean la misma clave.

Dependiendo del método elegido, el grado de seguridad varía; el método más seguro es el DES-EEE3.

- **AES [AESCSRC]**

AES (*Advanced Encription Standard* o *Estándar Criptográfico Avanzado*) es un algoritmo de cifrado por bloques destinado a reemplazar al DES como estándar. Este es un bloque simétrico que puede procesar datos de 128 bits, utiliza llaves cifrar de 128, 192 y 256 bits.

Tamaño de la llave = 128 bits, $0 \leq n < 16$
Tamaño de la llave = 192 bits, $0 \leq n < 24$
Tamaño de la llave = 256 bits, $0 \leq n < 32$

Tamaño del bloque = 128 bits, $0 \leq n < 16$

- **RC2**

RC2 es un algoritmo de cifrado por bloques de clave de tamaño variable diseñado por Ron Rivest de RSA Data Security (la RC quiere decir *Ron's Code* o *Rivest's Cipher*).

El algoritmo trabaja con bloques de 64 bits y entre dos y tres veces más rápido que el DES en software. Se puede hacer más o menos seguro que el DES contra algoritmos de fuerza bruta eligiendo el tamaño de clave apropiadamente. Este algoritmo está diseñado para reemplazar al DES.

- **RC4**

RC4 es un algoritmo de cifrado de flujo diseñado por Ron Rivest para RSA Data. Es un algoritmo de tamaño de clave variable con operaciones a nivel de byte. Se

basa en el uso de una permutación aleatoria y tiene un periodo estimado de más de 10^{100}. Además, es un algoritmo de ejecución rápida en software.

El algoritmo se emplea para cifrado de ficheros y para cifrar la comunicación en protocolos como el SSL (TLS).

- **RC5**

RC5 es un algoritmo adaptable con tamaño de bloque variable, tamaño de clave variable y número de rotaciones variable. Los valores más comunes de los parámetros son 64 o 128 bits para el tamaño de bloque, de 0 a 255 rotaciones y claves de 0 a 2048 bits. Fue diseñado en 1994 por Ron Rivest.

RC5 tiene 3 rutinas: expansión de la clave, cifrar y descifrar. En la primera rutina la clave proporcionada por el usuario se expande para llenar una tabla de claves cuyo tamaño depende del número de rotaciones. La tabla se emplea en el cifrado y descifrado. Para la cifrar sólo se emplean tres operaciones: suma de enteros, o-exclusiva de bits y rotación de variables.

La mezcla de rotaciones dependientes de los datos y de distintas operaciones lo hace resistente al criptoanálisis lineal y diferencial. El algoritmo RC5 es fácil de implementar y analizar y, de momento, se considera que es seguro.

- **IDEA**

IDEA (*International Data Encription Algorithm*) es un algoritmo de cifrado por bloques de 64 bits iterativo. La clave es de 128 bits. Al cifrar se dé precisa de 8 rotaciones complejas. El algoritmo funciona de la misma forma para cifrar que para descifrar (excepto en el cálculo de las subclaves). El algoritmo es fácilmente implementable en hardware y software, aunque algunas de las operaciones que realiza no son eficientes en software, por lo que su eficiencia es similar a la del DES.

El algoritmo es considerado inmune al criptoanálisis diferencial y no se conocen ataques por criptoanálisis lineal ni debilidades algebraicas. La única debilidad conocida es un conjunto de 251 claves débiles, pero dado que el algoritmo tiene 2^{128} claves posibles no se considera un problema serio.

- **SAFER**

SAFER (*Secure And Fast Encription Routine*) es un algoritmo de cifrado por bloques no propietario. Está orientado a bytes y emplea un tamaño de bloque de 64 bits y claves de 64 (SAFER K-64) o 128 bits (SAFER K-128). Tiene un número variable de rotaciones, pero es recomendable emplear como mínimo 6.

El algoritmo original fue considerado inmune al criptoanálisis lineal y diferencial, pero Knudsen descubrió una debilidad en el generador de claves y el algoritmo fue modificado (SAFER SK-64 y SAFER SK-128).

- **Blowfish**

Es un algoritmo de cifrado por bloques de 64 bits desarrollado por Scheiner. Es un algoritmo de tipo Feistel que posee un método de cifrado en bloque con su propia estructura particular. Cada rotación consiste en una permutación y una sustitución que depende de la clave y los datos. Todas las operaciones se basan en or-exclusivas sobre palabras de 32 bits. La clave tiene tamaño variable (con un máximo de 448 bits) y se emplea para generar varios vectores de subclaves. Este algoritmo se diseño para máquinas de 32 bits y es considerablemente más rápido que el DES.

El algoritmo es considerado seguro aunque se han descubierto algunas claves débiles, un ataque contra una versión del algoritmo con tres rotaciones y un ataque diferencial contra una variante del algoritmo.

4.2.13 Algoritmos de clave pública [SCHNEI96]

- **RSA**

RSA, llamado así por las siglas de sus creadores (*Rivest, Shamir y Adelman*), es el algoritmo de clave pública más popular. El algoritmo se puede usar para cifrar comunicaciones, firmas digitales e intercambio de claves.

La clave es de tamaño variable, generalmente se usan claves entre 512 y 2048 bits. Las claves más grandes aumentan la seguridad del algoritmo pero disminuyen su eficiencia y generan más texto cifrado. Los bloques de texto en claro pueden ser de cualquier tamaño, siempre que sea menor que la longitud de la clave. Los bloques de texto cifrado generados son del tamaño de la clave.

La *clave pública* del algoritmo tiene la forma (e, n), donde e es el exponente y n el módulo. La longitud de la clave es igual al número de bits de n. El módulo se obtiene multiplicando dos números primos grandes, p y q. Los números se seleccionan de forma aleatoria y se guardan en secreto.

La clave privada tiene la (d, n), donde d es el producto inverso de e modulo (p-1) (q-1) (es decir, (ed – 1) es divisible por (p-1)(q-1)).

El cálculo de d a partir de p y q es sencillo, pero es computacionalmente imposible calcular d sin conocer p y q para valores grandes de n, ya que obtener sus valores es equivalente a factorizar n, que es un problema intratable computacionalmente.

El funcionamiento del algoritmo es como sigue:

- Cifrado

Para cifrar un mensaje un usuario calcula c=m e modulo n, donde m es el texto en claro, c es el cifrado y (e, n) es la clave puvlica del destinatario.

- Descifrado

Para descifrar el mensaje el destinatario calcula c d modulo n, donde m es igual a tener (me) d modulo n = m, donde (d, n) es la clave privada del destinatario. La ultima sustitución es posible por el modo en que se escogen los números, ya que d es el producto inverso de e modulo n, por lo que med = m.

- Firmado

Si el emisor desea enviar el mensaje firmado usa su clave privada para calcular c = md modulo n y el destinatario lo valida calculando ce modulo n = (md) e modulo n = mde modulo n =m, donde (e, n) es la clave publica del emisor.

El algoritmo es lento, ya que emplea operaciones matemáticas que tienen un coste elevado y trabaja con claves de gran tamaño. Parte del problema está en la elección del exponente e, ya que un exponente de 512 bits escogido de forma aleatoria precisa 768 multiplicaciones en promedio. Para solucionarlo se suelen escoger los valores 3 ó 65537, que precisan 3 y 17 multiplicaciones respectivamente. La elección de un exponente fijo no disminuye la seguridad del algoritmo si se emplean esquemas de criptografía de clave pública adecuados, como por ejemplo el relleno de mensajes con bits aleatorios.

Adicionalmente, el uso de exponentes fijos hace que al cifrar sea más rápido el descifrado y la verificación de los datos más rápida que el firmado. Esta última característica es incluso deseable, ya que un usuario firma una vez un mensaje pero es posible que la firma se valide muchas veces.

Comparado con los sistemas de cifrado simétrico como el DES, el algoritmo de RSA es 100 veces más lento en software y de 1000 a 10000 veces más lento en hardware.

- **Diffie-Hellman**

El algoritmo de Diffie Hellman es un algoritmo de clave pública que permite el intercambio seguro de un secreto compartido. Generalmente se emplea junto con algoritmos de cifrado simétrico, como método para acordar una clave secreta. El algoritmo no se puede usar para cifrar conversaciones o firmas digitales.

El funcionamiento del algoritmo es como sigue:

- Paso 1

El emisor escoge un número primo grande p y un generador g (g<p) y se los envía al destinatario. A continuación escoge un número grande d_A como clave privada y calcula la clave pública correspondiente eA = gdAmodulop.

- Paso 2

De modo similar, el destinatario escoge una clave privada dB y una clave privada eB = gdB modulo p.

- Paso 3

Ambos participantes intercambian sus claves públicas y calculan un secreto compartido. El del emisor será sA = eBdA = (gdB)dA = gdBdAmodulop. Y el del destinatario sB = eAdB = (gdA)dB = gdAdB = gdBdA modulo p.

Con este sistema, aunque un tercero interceptara los números p y g y las claves públicas e_A y e_B, no podría calcular el secreto compartido sin tener una de las claves privadas, lo que equivale a calcular el logaritmo discreto de una de las claves públicas, que es un problema intratable computacionalmente.

El problema fundamental de este algoritmo es que es sensible a ataques activos del tipo *hombre en el medio*. Si la comunicación es interceptada por un tercero, este se puede hacerse pasar por el emisor de cara al destinatario y viceversa, ya que no disponemos de ningún mecanismo para validar la identidad de los participantes en la comunicación. Así, el *hombre en el medio* podría acordar una clave con cada participante y retransmitir los datos entre ellos, escuchando la conversación en ambos sentidos.

4.2.14 Seguridad

Al hablar de seguridad nos referimos a varios ítems que se deben de tener en cuenta; como proteger la información que se tiene, evitar que sea modificada o suplantada. Para poder proporcionar seguridad existen una serie de servicios a los cuales se le llaman servicios de seguridad. [WAPANA]

Servicios de Seguridad			
Autenticación de la Entidad		En este servicio se verifica la fuente de los datos, se puede hacer la autenticación de la entidad origen, destino o ambas.	
-	**Mecanismo de Autenticación**	Contraseña y Respuesta	Consiste en una autenticación de usuario donde solo el usuario conoce la contraseña
		Certificados	Se utiliza cuando no basta con

		Digitales	autenticar un usuario con contraseña, sino que requiere autenticar una Terminal, a través de la cual esta interactuado con un servidor o usuario del sistema. Estos son emitidos por una autoridad certificadora. Autentica cliente y servidor mediante una clave pública.
Confidencialidad de los datos		Este servicio evita que los datos de la comunicación sean revelados, de manera deliberada o accidental, es decir realizar cifrado de datos.	
-	**Claves para Cifrar**	Criptografía de clave pública o asimétrica	Utiliza dos tipos de claves para cifrar y descifrar. Se distribuye la clave públicamente entre los usuarios
		Criptografía de clave privada o simétrica	Se utiliza la misma clave para cifrar y descifrar datos. Ambas partes deben conocer dicha clave. La clave secreta se intercambia utilizando cifrado asimétrico.
Control de Acceso		Verifica que los recursos sean utilizados por quien debe hacerlo, se utilizan ACL para realizar Control de Acceso a Lista	
No repudio		Es la verificación a una tercera parte de cada entidad que ha participado en la comunicación.	
-			
Integridad de los Datos		En este servicio se verifica que los datos de una comunicación no se alteren, es decir que los datos recibidos por el receptor sean iguales a los recibidos por el emisor. Para esto se utilizan técnicas de resumen es decir Hashed, con esto se puede detectar cambios de contenido dentro del mensaje.	

Tabla. 3 Servicios de Seguridad

4.3 MARCO CONCEPTUAL

4.3.1 Azar [REAL2007]

El concepto azar viene del ár. hisp. *azzahr, y este del ár. zahr, dado (Sin rumbo ni orden.), literalmente 'flores'. Significa casualidad, caso fortuito, desgracia imprevista.

4.3.2 ¿Qué es un Juego de azar? [CORA2008]

Apostar dinero u otra cosa de valor a un evento futuro, posibilidad, o contingencia que es desconocida o incierta a los participantes. La característica esencial de un juego de azar es la apuesta o el riesgo que se corre, como tal.

En un juego de azar la posibilidad de ganar no depende de la habilidad del jugador; más bien del azar. El premio está determinado por la probabilidades estadísticas; entre menor sea la probabilidad de acertar la combinación correcta mayor deberá ser el premio.

4.3.3 Feceazar (Federación Colombiana de Empresas de Juegos de Azar) [FECE2008]

Feceazar es una entidad civil sin ánimo de lucro, tiene su oficina principal en Bogotá, D.C., pero por decisión de su Asamblea General podrá establecer Regionales, Capítulos, Seccionales, Delegaciones, Oficina u otras dependencias en cualquier lugar del país, en cuyo caso la Junta Directiva Nacional reglamentará su funcionamiento.

4.3.4 Juegos de azar

Dentro de los diferentes tipos de juegos de azar podemos encontrar los siguientes:

Bingo	Parqués
Cara o Cruz	5 de oro
Chino	Ruleta
Dados	Rifa
Lotería	Keno
Lotería primitiva o “lotto”	Quiniela
Lotería instantánea o “rasca”	Máquina tragaperras o tragamonedas

Tabla. 4 Tipos de juegos de azar

4.3.5 Chance [REAL2007]

Esta definición se refiere a oportunidad o posibilidad de conseguir algo. “Es aquel que sin ser rifa o lotería y utilizando los resultados de los sorteos ordinarios de las loterías, permite que una persona seleccione una, dos, tres o cuatro cifras apostando con ellas una suma de dinero, pudiendo lograr un premio en dinero si coincide su apuesta con la última, dos, tres o cuatro últimas cifras o la combinación de éstas en cualquier orden, del premio mayor del sorteo de la lotería con que se apuesta, de acuerdo con el plan de premios establecido.”

4.3.6 Requisitos legales para juegos de azar con expendedor de chance [ETES2007]

- Prohibida la venta a menores de diez y ocho (18) años.
- Este boleto constituye un documento al portador.
- El apostador es responsable de la integridad del boleto y acepta que los números impresos en este boleto representan con certeza su selección.
- El derecho a cobrar un premio caduca a los sesenta (60) días calendario, contados a partir del día siguiente al sorteo correspondiente.
- Los boletos, su validez y el pago de premios están sujetos a los reglamentos de los juegos y leyes sobre la materia.
- Las fechas de los sorteos dependen de la programación de las loterías, quienes podrán autorizar cambios en la fecha, sin que esto genere responsabilidad en el operador del juego
- Pagos de premios menores se pagarán en los puntos de venta; los demás premios:
- Para juegos autorizados por ETESA, serán pagados por ETESA a través de su fiduciaria y/o red bancaria designada.
- Para los demás juegos en línea que se expidan en la Terminal serán pagados en las oficinas del concesionario y/o entidad responsable del pago de premios, el cual aparece en el frente del boleto.

4.3.7 Procedimiento para jugar un chance [ETES2007]

Las apuestas de chance se toman de uno a cuatro dígitos. Entre uno y dos dígitos se hace de manera directa, entre tres y cuatro dígitos se pueden realizar de manera directa o combinada. La mínima apuesta es de 300 pesos.

Figura. 12 Chance colombiano [ELTIEMP]

4.3.8 Formas de apostar en un chance [ETES2007]

- **Apuesta Directa o en orden**

  La apuesta se gana solo al acertar el número en el orden jugado en el chance.

- **Apuesta Combinada**

  La apuesta se gana si todos los dígitos del número apostado aparecen en cualquier orden en el resultado final de la lotería.

4.3.9 Plan de premios para chance

TIPO DE APUESTA	*EN ORDEN (DIRECTO)*	*EN CUALQUIER ORDEN COMBINADO*
4 cifras	4500 veces lo apostado	208 veces lo apostado
3 últimas cifras	400 veces lo apostado	83 veces lo apostado
2 últimas cifras	50 veces lo apostado	
última cifra	5 veces lo apostado	

Tabla. 5 Rango de cifras y pagos según lo apostado [ETES2007]

4.3.10 Información requerida para realizar un chance [ETES2007]

- Cantidad de cifras
- Lotería con la cual desea jugar
- Tipo de apuesta
- La apuesta en pesos colombianos

4.3.11 Información requerida para el formato de chance de la empresa que gestiona la apuesta en línea [ETES2007]

- Nombre de la empresa
- Contrato
- Dirección
- Nit

4.3.12 Loterías que se juegan en Colombia [ETES2007]

Lotería De Medellín

Baloto
Lotería el Dorado
Lotería de Bogotá
Lotería de Boyacá
Cash
Lotería Cauca
Lotería Cruz Roja
Lotería de Cundinamarca
Lotería del Huila
Lotería de Manizales
Paisita 1
Paisita 2
Play Four
Lotería del Quindío
Lotería de Risaralda
Lotería de Santander
Lotería de Tolima
Lotería del Valle

Tabla. 6 Loterías que juegan en Colombia [ELCO2008]

4.3.13 Apostar [JUEG2008]

El concepto apostar procede del latín apposĭtum, de apponĕre, colocar. Significa dicho de una persona. Pactar con otra u otras que aquel que se equivoque o no tenga razón, perderá la cantidad de dinero que se determine o cualquier otra cosa.

Arriesgar cierta cantidad de dinero en la creencia de que algo, como un juego, una contienda deportiva, etc. En caso de acierto se recupera el monto apostado a expensas de quienes han perdido.

4.3.14 Tipos de Apuestas [JUEG2008]

Existe una variedad de apuestas en línea, de cuales unas son más comunes que otras

- **Apuestas a Largo Plazo**

Tal y como su nombre indica se trata de una apuesta que abarca un largo periodo de tiempo. Normalmente se puede apostar meses antes por el ganador del campeonato del mundo de Fórmula 1, del Tour de Francia o de Wimbledon o a unas elecciones presidenciales desde muchos meses antes.

- **Apuestas Americanas**

En las apuestas europeas normalmente el pronóstico de empate viene normalmente unido a cuotas altas pero en las apuestas americanas el empate se considera "no action", es decir, como si el partido no se hubiera disputado. De esta forma el empate se valora normalmente con una cuota 1,0 y usted recibe el importe apostado en su cuenta de apuestas.

- **Apuestas Combinadas**

La apuesta combinada es una forma muy atractiva de apostar, ya que las ganancias pueden ser espectaculares. Seleccione varias apuestas y combínelas en una sola. La cuota total deriva de multiplicar las cuotas de las diferentes apuestas. Sin embargo, para poder ganar la apuesta deberá acertar todos los resultados. Ejemplo: Real Madrid – Barcelona: Real Madrid gana (cuota 2,5) Y Dallas Cowboys-St. Louis Rams: St. Louis Rams ganan (cuota 3) Y Deportivo - Alavés: Descanso 1:0 (cuota 4) – Cuota total: 2,5 x 3 x 4 = 30

- **Apuestas de Posición**

La apuesta de posición son DOS apuestas en una. Consisten en que el jugador seleccionado gana y otra a que termina entre las primeras posiciones.

Una apuesta de Posición de 100 euros tiene un coste total de 200 euros (100 para cada parte de las apuestas). El rango de las primeras posiciones varía según el evento y siempre está especificado. Si su elección gana, ambas partes de su apuesta ganan. Si su elección termina entre las posiciones especificadas (2do, 3er, 4to lugar, etc.), una parte de su apuesta gana y la otra pierde.

- **Apuestas Especiales**

Apuesta por situaciones determinadas que sucederán en el partido. Algunas situaciones donde aplican las apuestas especiales. Las apuestas especiales son diferentes según el deporte. Ejemplo: ¿En qué minuto marcarán el primer gol? O ¿Quién se detendrá de primero en los boxes? O ¿Cuántos sets jugarán?

- **Apuestas Handicap**

Si un contrincante supera a otro con una clara ventaja, los corredores de apuestas suelen darle al más débil de ellos una ventaja que suele sumarse al resultado real del encuentro. Por ejemplo: Real Madrid – Rayo Vallecano. Hándicap 0:2 (esos goles se sumarán a los del Rayo. El Real Madrid gana 1:0. Pero con el hándicap de la apuesta el resultado sería de 1:2. El que haya apostado por el Rayo gana.

- **Apuestas Live**

Las apuestas LIVE son apuestas en tiempo real. Usted apuesta mientras se celebra el acontecimiento deportivo. Para poder cerrar apuestas LIVE necesita el Flash-Plug-In. Instalado en su navegador.

- **Apuestas WAP, Desde teléfonos móviles.**

Ahora podrá usted puede hacer sus apuestas deportivas desde cualquier lugar; en medio de un emocionante partido desde el propio estadio o cómodamente desde

su casa, apuestas: cuando quiera y como quiera desde su móvil. Esta modalidad de apuestas es una forma muy innovadora que permitirá al usuario realizar apuesta desde su propio móvil, solamente ingresando a la dirección WAP suministrada por la casa de apuestas.

- **Apuestas Simples**

La apuesta simple o individual es la forma más sencilla de apostar: un encuentro y un pronóstico. Hay tres tipos de individuales o simples: las 1X2, las 12 (con dos o tres posibles resultados) o las apuestas por victoria.

- **Victoria (Apuesta por victoria)**

En esta modalidad de apuesta todos quieren ganar pero solo uno lo consigue, un ejemplo calaro es el Tour de Francia o en el Gran Prix de Formula 1 o en el Mundial.

- **1X2 (Uno, X, Dos)**

Tres resultados posibles, sólo uno es correcto. La simple apuesta de fútbol: victoria, empate o derrota es la típica apuesta 1X2. Ejemplo: Bayern München frente al Manchester United 1, X, 2?

- **12 (Uno, Dos)**

Apuestas sobre encuentros con dos resultados posibles. Solo uno es el correcto. Esta apuesta funciona con deportes como el baloncesto o el tenis donde no existe el empate. Además hay numerosas apuestas especiales que funcionan mediante este principio. Agassi contra Kuerten. ¿Quién gana?

- **Apuestas Sistema**

Con las apuestas sistema puede hacer varios pronósticos con un cierre de apuesta. Si todos los pronósticos son correctos, usted gana todas las apuestas de su sistema. Y si no ha acertado todos los pronósticos, todavía le queda una oportunidad de ganar, porque uno o varios pronósticos pueden ser acertados. Por ejemplo: una apuesta sistema 3/5, es decir 3 de 5, significa que usted apuesta sobre todas las posibles combinaciones de 3 dentro de las 5 parejas de juegos seleccionados. Si ha acertado con todos sus pronósticos se asegura la ganancia máxima. En caso de que un pronóstico sea falso tendrá cuatro de las 10 correctas y con dos pronósticos erróneos seguirá teniendo una combinación de tres.

4.4 ESTADO DEL ARTE

En el entorno de las comunicaciones existen las terminales **TPV** cuyas iníciales significan **Terminal Punto de Venta** sus siglas en inglés "POS" quieren decir "Point Of Sale". Estas hacen referencia a los programas y tecnologías que ayudan en tareas de gestión de un negocio de venta al público.

Un TPV posee sistema informático que gestiona un proceso de venta, por medio de una interfaz accesible al usuario. Este sistema permite la generación de los tiquetes las actualizaciones en las bases de datos y otros requerimientos. [TPVM2008]

Dentro de entorno de apuestas digitales encontramos múltiples sistemas de juego. Entre ellos aparece el Datafono, que es uno de los métodos utilizados actualmente en juegos de chance en Colombia, estos manejan un nivel de seguridad a nivel de red celular.

El Datafono es un dispositivo de diseño compacto y ligero, que por medio de la tecnología móvil inalámbrica y red celular logra mantener siempre una conexión realizando transacciones rápidas, seguras y sin error, ofreciendo voz y transmisión de datos algunas en tiempo real. Adicionalmente los Datafonos cuentan con lectores de tarjetas y bandas magnéticas que permiten la interacción con el usuario. [MERC2006].

Entre las tecnologías existentes podemos encontrar la i-mode, que es un conjunto de tecnologías y protocolos diseñados para poder navegar a través de mini páginas o portales WAP, diseñados específicamente para dispositivos móviles tales como teléfonos o PDAs. El diseño de dichos portales se hace utilizando lenguajes de programación similares al de HTML, estas son variaciones de lenguaje modificados para los teléfonos móviles.

El lenguaje i-mode fue creado en 1999 por empresas de telefonía japonesas, soportando imágenes y móviles a color. Dentro del estándar i-mode se incluye una tecnología para hacer aplicaciones basadas en Java, llamada doja, aunque no todas las terminales i-mode lo soportan.

La tecnología i-mode compite con WAP y otras tecnologías de navegación para dispositivos móviles. La velocidad de transferencia máxima que soporta i-mode es de 9.6 kbps, la cual es superada por la nueva tecnología UMTS que tiene un límite de velocidad de 384 Kbps.

5. METODOLOGÍA

Dentro del desarrollo de tecnologías celulares encontramos herramientas que nos permiten modelar el mundo que nos rodea. Estas herramientas facilitan la implementación de soluciones a necesidades del entorno humano. La creación de aplicaciones en dispositivos móviles se origina del lenguaje de programación J2ME (*Java 2 Micro Edition*). El cual permitirá el desarrollo de la aplicación prototipo que se conectara al servidor por medio del protocolo de Internet del celular para realizar una apuesta.

Antes de determinar el dispositivo adecuado se determinan los algoritmos de cifrado disponibles a los cuales se les identifican las características que presentan cada uno de estos, esto permitirá conocer los algoritmos existes, para seleccionar el método más seguro y eficiente que puede ser implementado en el dispositivo. Se deberá conocer que sistemas operativos existen en el mercado celular, que características de seguridad presentan cada uno de estos y que herramientas para el desarrollo están al alcance.

Dentro del los algoritmos de cifrado que pueden ser implementados dentro de un dispositivo móvil existen ciertas características que se tienen en cuenta para la elección del dispositivo móvil. Entre las limitaciones que presentan los celulares se encuentran los recursos de procesamiento, las limitaciones de memoria, sistema operativo y finalmente los algoritmos que sean compatibles con el sistema operativo, debido a que las especificaciones varían de uno a otro dispositivo.

En la determinación del algoritmo DE cifrado adecuado se evalúan las opciones más acordes a los dispositivos móviles celulares, en la demostración de implementación de la forma de transmisión segura escogida se requiere de una aplicación prototipo, con la cual se realiza la apuesta.

Los requerimientos básicos constan de la identificación de usuario y contraseña a nivel de servidor, para realizar autenticación. Se ingresa a un menú de apuesta, donde se selecciona el número a jugar, el tipo, monto de la apuesta, para finalmente hacer entrega de un número de identificación.

Estos datos viajaran por la red de datos celular de manera cifrada por la aplicación creada para el móvil en J2ME. La conexión será permanente mientras se realiza la apuesta.

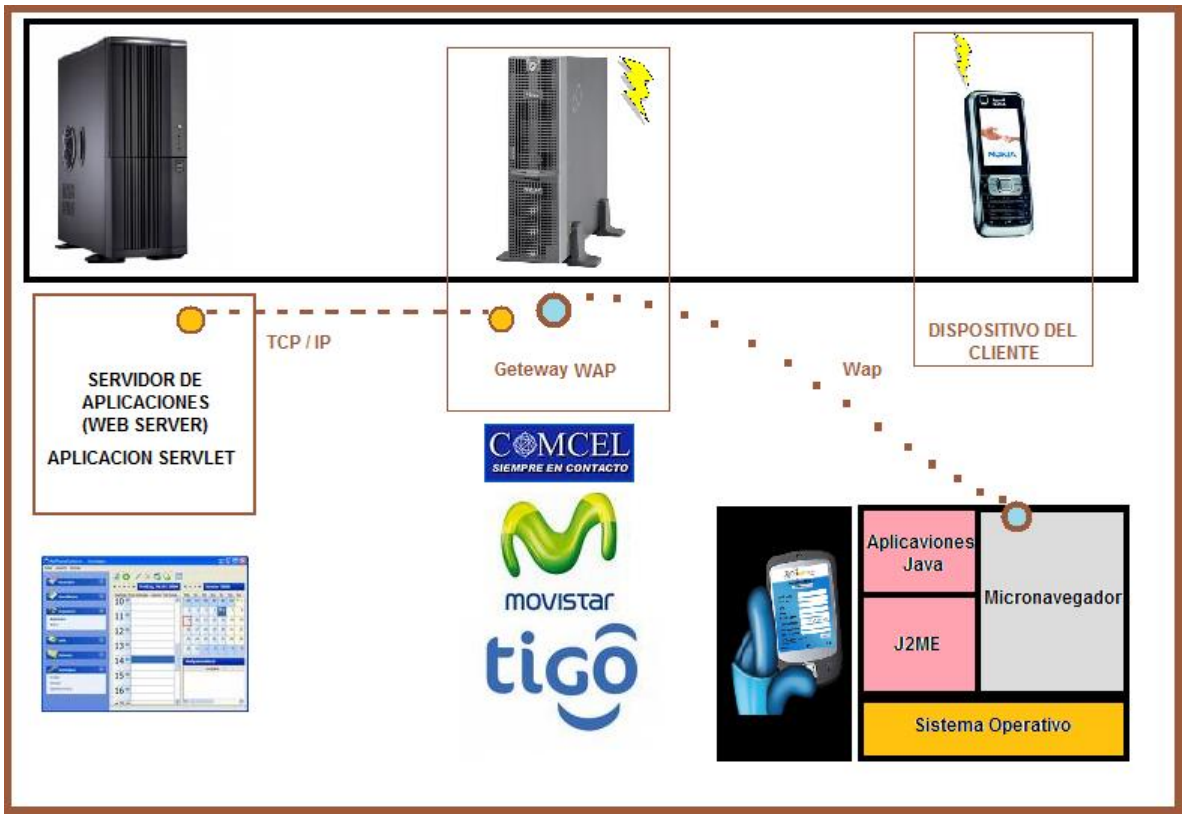


Figura. 13 Forma de comunicación entre el móvil y el servidor [J2ME2003]

6. DESARROLLO

Para el desarrollo de la comunicación segura entre el dispositivo móvil celular y el servidor WEB es necesario conocer las diferentes formas para hacer un canal seguro, estas formas se encuentran basados en los algoritmos de cifrados.

Los algoritmos de cifrado son técnicas matemáticas con las cuales se pueden crear mensajes no entendibles para usuarios no admitidos o intrusos. Los algoritmos se pueden clasificar según su utilidad:

- Sistemas de Cifrados Simétricos (Para cifrar información)
- Criptografía Hash (Para resumen de información)
- Firmado Digital (Autenticación)
- Sistemas de Cifrados asimétricos (Para cifrar información)

Adicionalmente cada algoritmo presenta unas características que los identifican los unos de los otros como el desarrollo matemático de cada uno, un modo de empleo ya sea en bloque, por flujo de datos, firmado digital o un resumen de mensaje. Por otro lado se encuentran las llaves de cada algoritmo, que indican la robustez del mismo.

En la siguiente tabla se encuentran los algoritmos existentes clasificados por subsistemas de cifrado, con sus correspondientes llaves y modo de empleo.

Tabla comparativa de los algoritmos de cifrado		
Sistemas de cifrados simétricos	**Tamaño de la llaves**	**Uso**
Advanced Encryption Standard (AES)	128/192/256	Cifrado de bloques simétrico
Data Encryption Standard (DES) and Triple DES	56/112/168	Cifrado de bloques simétrico
SAFER	64/128	Cifrado de bloques simétrico
IDEA	64	Cifrado de bloques simétrico
RC4	Variable	Flujo de datos cifrado simétrico
RC2	Variable	Cifrado de bloques simétrico
Criptografía hash	**Tamaño de la llaves**	**Uso**
SHA-1/SHA-256/SHA-384/SHA-512	160/256/384/512	Resumen de mensaje
MD5	128	Resumen de mensaje

MD2	128	Resumen de mensaje
Sistemas Firmado Digital	**Tamaño de la llaves**	**Uso**
RSA Signature (using PKCS #1 v1.5)	<= 8192-bit modulo	Firma Digital
Digital Signature Algorithm (DSA)	<= 1024-bit modulo	Firma Digital
Sistemas Cifrado Asimétricos	**Tamaño de la llaves**	**Uso**
RSA Key Agreement (using PKCS #1 v1.5)	<= 8192-bit modulo	Cifrado en Bloques Asimétrico
Diffie-Hellman Key Agreement (using PKCS #3)	<= 2236-bit modulo en NSS 3.9.3+	Cifrado en Bloques Asimétrico
	<= 1024-bit modulo en NSS 3.9.2-	

Tabla. 7 Tabla comparativa de los algoritmos de cifrado según su uso y el tamaño de la llave

En la siguiente figura se ve la conformación da la capa de seguridad SSL, sobre esta capa de implementa el certificado SSL, que permite el envió de datos cifrados, firmados e íntegros.

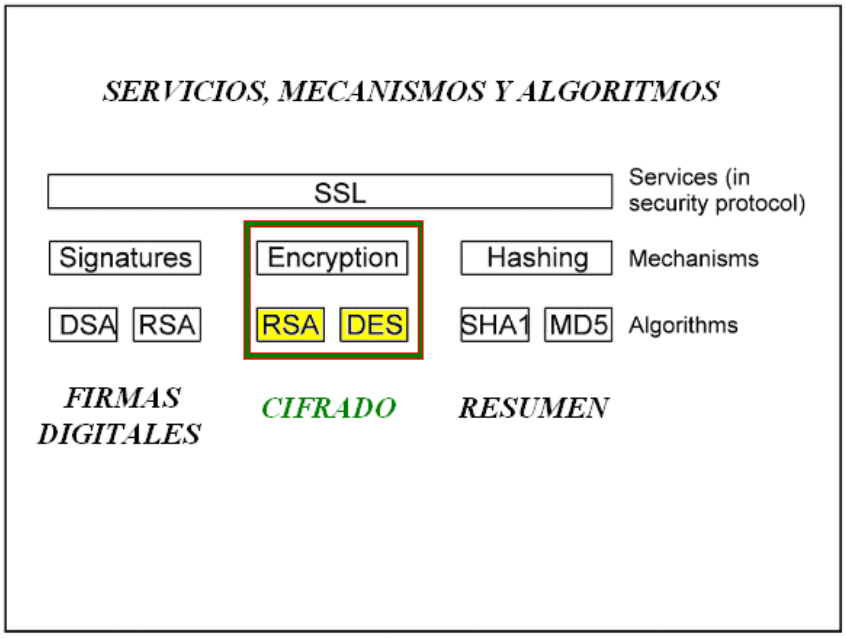


Figura. 14 Diagrama según el mecanismo para los algoritmos de cifrado

6.1 Sistemas Operativos para Dispositivos Móviles [SisOpeCon1993]

En el desarrollo del proyecto de grado se ha elegido como dispositivo móvil el celular, ya que este presenta portabilidad e infraestructura de comunicación ya implementada. Existen gran variedad de celulares con diferentes características en los recursos, estas son indispensables en la elección del dispositivo adecuado, ya que unos móviles presentan características más robustas y otros muy limitadas para la implementación de la aplicación móvil que permitirá la prueba del canal de transmisión segura.

En la actualidad se tiene una tendencia a la unificación de un sistema operativo que integre la compatibilidad entre varias empresas. Esto permitirá la compatibilidad de aplicaciones desarrolladas para los distintos dispositivos. Es así que se crea el sistema operativo Symbian siendo un producto de la alianza de varias empresas de telefonía móvil entre las que se encuentran Nokia, Sony Ericsson, PSION, Samsung, Siemens, Arima, Benq, Fujitsu, Lenovo, LG, Motorola, Mitsubishi Electric, Panasonic, Sharp, etc. [COMOV]

Los sistemas operativos no pertenecen solo a los PCs, en los dispositivos móviles se encuentran gran variedad de estos, es así que en los celulares existen una variedad de sistemas operativos que no son tan conocidos como los del PC.

Para poder determinar un sistema operativo que sea óptimo para la implementación de la aplicación móvil, primero se deben conocer las características que pueden ofrecer. Se deben determinar que sistemas operativos hay actualmente en el mercado, que interfaz presenta para el usuario, si esta es amigable y que empresas los desarrollan.

En el siguiente recuadro se pueden encontrar los SO (Sistemas Operativos) disponibles en el merado móvil celular:

Symbian
BlackBerry
Sony Ericsson
Apple
Palm Web OS
Android
Windows Mobile

Figura. 15 Sistemas Operativos Móviles [ENGA2009]

Para cada dispositivo móvil existente se ha diseñado una interfaz, que muchas veces no es creada por el mismo proveedor del sistema operativo, es el caso de Palm Web OS cuyo sistema operativo es PalmOS y su Kernel interfaz es diseñada por Linux.

Figura. 16 Sistema Operativo Móvil con su Interfaz [ENGA2009]

En los diferentes sistemas operativos existentes, se encuentran características que constituyen el perfil y funcionalidad de cada uno de estos. En SO la parte más importante es el núcleo, también llamado Kernel que define la interfaz hacia el usuario. [SisOpeCon1993]

Dentro de los SO se encuentra Android, que es un sistema operativo el cual utiliza Linux mezclado con Java como núcleo; Iphone que se basa en OS X que es una variante de Unix, siendo Unix uno de los SO más poderosos en el mercado. Dentro de los SO más estables se encuentra Windows Mobile y S60 de Symbian, estos aparte de ser estables son SO muy maduros en cuanto a desarrollo y tiempo en el mercado. Otro de los SO más importantes es RIM (Research in Motion) BlackBerry posee un motor de Java tal como Android. [SisOpe1993]

En el siguiente diagrama se pueden encontrar las características de cada sistema operativo, incluye la versión actual, la adaptabilidad del sistema operativo frente a otros, el tipo de Kernel que maneja cada uno y las diferentes tecnologías que soportan. [Stall2001]

Sistemas Operativos / Mercado		Versión del Software	Adaptabilidad	Kernel o tipo de Núcleo	Tecnología
BlackBerry RIM	Research in Motion	BlackBerry OS 4.7	Buena	Propietario	GSM WIFI CDMA
htc smart mobility Windows Mobile	Windows Mobile	Windows Mobile 6.5.	Excelente	Windows CE	GSM WIFI CDMA
palm webOS	PalmOS	Palm WebOS	Excelente	Linux	GSM WIFI CDMA
symbian	Symbian	S60	Excelente	Symbian	GSM WIFI
android	Linux	Android con Cupcake	Excelente	Linux	GSM WIFI
Sony Ericsson Java	Sony Ericsson Propietario	MIDP 2.1	Excelente	Propietario	GSM WIFI
Mac OS X	Mac OS X	iPhone OS 3.0	Mala	OS X	GSM WIFI

Tabla. 8 Características de los Sistemas Operativos Móviles [ENGA2009]

6.2 Algoritmos de cifrado en dispositivos móviles celular

En el desarrollo de algoritmos de cifrado para dispositivos móviles celular es importante tener en cuenta que cada compañía adopta variedad de sistemas operativos, en la siguiente tabla se pueden encontrar una gama de proveedores con uno de los sistemas operativos que han implementado. [SisOpeCon1993]

Se pueden encontrar las características de seguridad, como los paquetes de desarrollo correspondiente a cada uno de los SO, las correspondientes herramientas para el desarrollo de aplicaciones para el respectivo SO, los diferentes mecanismos de seguridad respecto a cada uno de los operativos: [SisOpe1993] [JAVMIDP]

Proveedor	Compatibilidad	Descripción de los paquetes de desarrollo	Herramientas de desarrollo, compatibilidad	Seguridad
Nokia – Symbian	MIDP 2.0 y 1.x, CLDC 1.0	Orientado al desarrollo de aplicaciones para dispositivos Nokia compatibles con J2ME. Versiones de Nokia para J2ME 1.0 2.0	Net Beans Mobility Pack 4.0 (Beta) Eclipse ME J2ME Development for Eclipse 3.0M7	Mecanismos de algoritmos criptográficos, llaves de hash, certificados y firmado digital
Motorola - Linux	MIDP 2.0 y 1.x, CLDC 1.0	Desarrollo de aplicaciones J2ME orientadas a móviles iDEN compatibles con J2ME. Existen varias versiones del Motorola iDEN: SDK 3.0.0 + update i860 SDK Java Application Loader J2ME Open Windowing Toolkit	¡Vise Sun Java Studio Mobility 6 Net Beans Mobility Pack 4.0 (Beta)	Aun no en el mercado presenta problemas de seguridad, que se han tratado de solucionar utilizando SDK
Sony Ericsson – Propietario	MIDP 2.0, MIDP 1.0 y CLDC 1.x, JSR-184(3D)	Sony Ericsson J2ME SDK y herramientas y documentación asociada. Existen distintas versiones de las SDK: J2ME SDK 1.0 J2ME SDK 2.0 J2ME SDK 2.1.2 (beta)	Eclipse ME J2ME Development for Eclipse 3.0M7 Net Beans Mobility Pack 4.0 (Beta)	Certificado HTTPS – Firmas criptográficas- perfil de certificados MIDP X509 – Algoritmos de Cifrados SHA-1 y RSA

		J2ME SDK 2.1.3 (beta) P900 J2ME SDK		
Siemens – Symbian	MIDP 1.0 y CLDC 1.0	Desarrollo de aplicaciones J2ME orientadas a teléfonos móviles Siemens compatibles con J2ME. Está compuesta por el *SMTK Core Pack* y paquetes de emuladores para los modelos *45 y 55* o *65*.	Sun Java Studio Mobility 6 Net Beans Mobility Pack 4.0 (Beta) Eclipse ME J2ME Development for Eclipse 3.0M7	Mecanismos de algoritmos criptográficos, llaves de hash, certificados y firmado digital
HTC – Windows Mobile	MIDP 2.0, MIDP 1.0	J2ME SDK 1.5	Sun Java Studio Mobility 6 Eclipse ME J2ME Development for Eclipse 3.0M7	Criptografía DES, 3DES,MD5, RSA, SHA1, Autenticación y certificados SSL, TSL, WTLS
BlackBerry - RIM	MIDP 2.0, MIDP 1.0	BlackBerry Sync Server SDK version 4.1.2 Java® SDK Versión 1.5	Mobil Data System v4.1 - BlackBerry MDS Services - BlackBerry MDS Developer Tools - BlackBerry MDS Device Software	Utila TRIPE DES – AES, Mobil Data Service MDS – HTTP, HTTPS, WTLS
Iphone Mac OS X	Propietario de Macintosh	Propietario de Macintosh	Flurry	A nivel de seguridad presenta muchos fallos de seguridad, se pueden tomar los datos y los sms utilizando una página web.
Samsung - propietario	JAVA™ MIDP 2.0, CLDC 1.1	SDK Java para desarrollo de aplicaciones compatibles J2ME para teléfonos Samsung. Existen dos versiones de la JDK: 1.0 2.0	Sun Java Studio Mobility 6 Net Beans Mobility Pack 4.0 (Beta)	AES, DES, 3DES, MD5 Y SHA

Tabla. 9 Características de seguridad con respecto a los Sistemas Operativos Móviles

6.3 Selección del Dispositivo Móvil

Para poder seleccionar el dispositivo móvil adecuado se deben tener en cuenta todas las características que proporciona cada sistema operativo.

- Enfoque en el que desarrollo el dispositivo
- Tecnologías soportadas
- Adaptabilidad
- Compatibilidad
- Herramientas de desarrollo
- Paquetes de desarrollo
- Seguridad soportada

6.3.1 Enfoque en el que desarrollo el dispositivo

Al desarrollar un dispositivo cada empresa, no solo se centra en el desarrollo tecnológico. Otro de sus objetivos es brindar servicios al usuario, es así que dispositivos que cuenten con sistemas operativos como BlackBerry, PalmOS y Windows Mobile desarrollan interfaces orientadas a servicios, como Office, editores, Outlook, mensajería instantánea, etc. Dispositivos con sistema operativo Mac OS, se enfocan en calidad y entretenimiento al usuario con sus propios desarrollos. Android el sistema operativo Linux con una alianza con google cuyo objetivo es dar al usuario un acceso más cómodo al desarrollo de interfaces al gusto.

Symbian es uno de los sistemas operativos más estable, es fruto de una alianza de empresas que buscan la unificación de un mercado, permite el desarrollo de aplicaciones. Finalmente Sony Ericsson un sistema operativo propietario enfocado al desarrollo de aplicativos Java, ya que su sistema operativo es totalmente Java brinda la facilidad de implementación aplicativos desarrollados por el usuario sin tener problemas de librerías.

6.3.2 Adaptabilidad

La adaptabilidad es uno de los puntos más fuerte en cuanto a desarrollo de aplicativos. Muchos de los aplicativos diseñados para determinados sistemas operativos, no son compatibles por las características a nivel de desarrollo que maneja cada compañía. Empresa como Macintosh desarrollan sus propias aplicaciones en su propio lenguaje con sus propios criterios de diseño, esto hace que la adaptabilidad de sus aplicativos a muchos dispositivos móviles sea nula, ya que muchos de estos utilizan Java con su propio perfil. BlackBerry tiene sus propios desarrolladores, estos incluyen desarrollos Java, poseen buena adaptabilidad. En cuanto a Sony Ericsson Propietario, Symbian, Windows Mobile, PalmOS, Linux poseen una adaptabilidad excelente porque desarrolla en Java aunque posean su propio perfil.

6.3.3 Compatibilidad

La compatibilidad de cada dispositivo se puede observar en la Tabla 7. MIDP es el perfil para dispositivos de información - Mobile Information Device Profile, MIDP, es el encargado de la compatibilidad de las aplicaciones. El entorno de ejecución en un dispositivo móvil se conforma al unir el MIDP y la configuración básica del dispositivo, CLDC.

El MIDP verifica si el aplicativo puede ser ejecutado en el dispositivo móvil. Se han desarrollado varias versiones que indica que cada vez hay aplicativos que antes no eran compatibles con el dispositivo y ahora pueden ser soportados por las nuevas versiones de equipos.

Existen dos versiones de MIDP 1.0 y MIDP 2.0, cada sistema operativo presenta compatibilidad con la versión actual del MIDP.

6.3.4 Tecnologías Soportadas

En cuanto en las tecnologías soportadas por cada uno de los sistemas operativos actuales en el mercado colombiano, no hay inconvenientes, ya que todos los sistemas operativos soporta la implementada actualmente la GSM, cuyas siglas significan Global Systems for Mobile Communications - Sistema Global de Comunicaciones Móviles.

Es un sistema digital de comunicación que transmite voz y datos. Este sistema digital es considerado como la Segunda Generación (2G) ya que a diferencia de la primera generación de celulares, combina tecnología digital y la división de acceso de transmisión múltiple (TDMA).

Para el envió de información GSM inicialmente digitaliza los datos, seguido de esto comprime la información y luego divide cada canal de 200MHZ en ocho espacios de tiempo de 25MHZ. Este sistema opera en las bandas 900MHZ y 1800MHZ en Europa, África y Asia, en las bandas 850MHZ y 1900MHZ en Estados Unidos. La banda 850MHZ también se utiliza para GSM y 3GSM en Canadá, Australia y en varios países de Latinoamérica.

6.3.5 Herramientas de desarrollo

En el mercado hay gran variedad de herramientas de desarrollo para aplicaciones Java. Para el desarrollo de un aplicativo Java para el móvil se dispone de un paquete de desarrollo Java. Este paquete se le conoce como J2ME, este se enfoca en la creación de aplicativos para dispositivos móviles con limitaciones de memoria, pantalla y procesamiento.

Algunos sistemas operativos presentan sus propias plataformas de desarrollo, en el siguiente diagrama se muestran plataformas de desarrollo para aplicativos móviles.

- Eclipse ME J2ME Develoment for Eclipse 3.0
- Netbeans Mobility Pack 4.0
- Sun Java Studio Mobility 6
- Mobil Data System v4.1 – BlackBerry

Teniendo en cuenta que hay variedad de herramientas de desarrollo para aplicaciones móviles, se hace selección de las herramientas conocidas para desarrollo de aplicaciones. Como herramienta seleccionada por su compatibilidad y conocimiento sobre ella se escoge Netbeans Mobility Pack 4.0. Con la selección de esta herramienta que limitan las opciones de sistemas operativos a cuatro por su compatibilidad con el SO (Sistema Operativo). En los siguientes ítems se pueden observar los sistemas operativos restantes después del sesgo:

- Symbian
- Linux
- Sony Ericsson Propietario
- Samsung Propietario

6.3.6 Paquetes de desarrollo

Cada uno de los sistemas operativos posee su propio paquete de desarrollo, el cual permite desarrollar los aplicativos compatibles con sus terminales móviles. Estos paquetes se descomponen en varias versiones como se puede observar en la tabla 7. Cada paquete esta enfocado a la gama y versión del dispositivo celular.

6.3.7 Seguridad Móvil

Aquí se definen las características de seguridad que pueden brindar los sistemas operativos (SO). Teniendo en cuenta los SO restantes (Symbian, Linux, Sony Ericsson Propietario, Samsung Propietario).

Para la implementación del canal seguro se utiliza un certificado SSL, por medio de este enviaran los datos cifrados y certificados. Los Sistemas Operativos que se adecuan al perfil de certificados son:

- Symbian
- Sony Ericsson Propietario

Estos Sistemas Operativos manejan algoritmos criptográficos llaves de hash certificados y firmado digital.

En la siguiente tabla se observa la selección de los sistemas operativos para la implementación del aplicativo

	Enfoque	Tecnología	Adaptabilidad	Compatibilidad	Herramientas de desarrollo	Paquetes de desarrollo	Seguridad soportada
Symbian	Aplicativos unificando un solo sistema operativo	Compatible	Excelente	MIDP 2.0	Netbeans Mobility Pack4.0 Eclipse ME j2ME Development for Eclipse 3.0	Propietario del Sistema Operativo	Algoritmos de cifrado – llaves hash – certificados –firmado digital
Sony Ericsson Propietario	Aplicativos para un sistema operativo basado en Java	Compatible	Excelente	MIDP 2.0	Netbeans Mobility Pack4.0 Eclipse ME j2ME Development for Eclipse 3.0	Propietario del Sistema Operativo	Algoritmos de cifrado – llaves hash – certificados HTTPS – firmado digital
Amdroid	Software libre	Compatible	Excelente	MIDP 2.0	Netbeans Mobility Pack4.0	Propietario del Sistema Operativo	Problemas de seguridad
Mac OS X	Entretenimiento	Compatible	Mala	MIDP 2.0	Flurry	Propietario del Sistema Operativo	Problemas de seguridad
BlackBerry	Empresarial	Compatible	Buena	MIDP 2.0	Mobil Data System v4.1	Propietario del Sistema Operativo	Algoritmos de cifrado – llaves hash – certificados HTTPS
Palm OS	Empresarial	Compatible	Excelente	MIDP 2.0	Sun Java Studio Mobility	Propietario del Sistema Operativo	Algoritmos de cifrado – llaves hash – certificados HTTPS
Samsung Propietario	Aplicativos compatibles con J2ME	Compatible	Excelente	MIDP 2.0	Sun Java Studio Mobility Netbeans Mobility Pack4.0	Propietario del Sistema Operativo	Algoritmos de cifrado – llaves hash
Windows Mobile	Empresarial	Compatible	Excelente	MIDP 2.0	Sun Java Studio Mobility Eclipse ME j2ME Development for Eclipse 3.0	Propietario del Sistema Operativo	Algoritmos de cifrado – llaves hash – certificados HTTPS

Tabla. 10 Características de seguridad con respecto a los Sistemas Operativos

Finalmente resultan dos sistemas operativos que reúnen las características para la implementación del aplicativo móvil.

6.3.8 Symbian [TecMov2005] [SOSYMB]

Symbian cuenta con cinco interfaces de usuario o plataformas para su sistema operativo, las denominadas Serie 60, Serie 80, Serie 90, UIQ y MOAP. La mayoría de los móviles Nokia utilizan la Serie 60, Algunos Sony Ericsson trabajan su propia interfaz bajo UIQ, así como también Motorola.

UIQ es una plataforma para terminales móviles que utilicen el sistema operativo Symbian. Esta versión es derivada del sistema operativo para teléfonos móviles Symbian, y se caracteriza por agregar soporte para pantallas táctiles.

La plataforma S60 *(Interfaz de usuario de serie 60*) está diseñada para las terminales que utilicen el sistema operativo Symbian OS. Este sistema operativo abarca una comunidad de desarrolladores de software, integración de sotware tales como Elektrobit, Teleca y Sysopen Digia, compañías de semiconductores como Texas Instruments, STMicroelectronics, Broadcom, Renesas y Freescale, además de proveedores de telefonía con los cuales desarrollan aplicación y servicios basados en esta plataforma.

La plataforma S60 permite la implementación de aplicaciones en Java MIDP 2.0, Symbian, C++, Flash y Phyton. El sistema operativo Symbian ofrece una variedad de características para la seguridad, la criptografía y la privacidad de los datos. En la tabla siguiente se pueden encontrar los algoritmos de cifrado que emplean según el tipo de mecanismo, ya sea para funciones de hash (resumen), firmado digital y finalmente cifrado.

Sistema Operativo Symbian	
cifrado con llave privada	DES, 3DES, AES, RC2, ARC4
cifrado con llave publica	RSA, DSA, DH
funciones de Hash	MD5, SHA-1/224/256/384/5112, HMAC
adaptación de plug-in	Mejorar la aceleración criptográfica
protocolos de comunicación	X.509, TLS 1.0, SSL v3, OCSP

Tabla. 11 Seguridad Criptografía y Privacidad de Datos del Sistema Operativo Symbian

Figura. 17 Sistema Operativo Symbian implementado en le Dispositivo Móvil Nokia 6120 [SOSYMB]

6.3.9 Sony Ericsson [SONYDEV]

Figura. 18 Sistema Operativo Propietario implementado en el Dispositivo Móvil Sony Ericsson K850i [SONYDEV]

Este dispositivo cuenta con Sistema Operativo Propietario MIDP (Mobile Información Device Profile) que es el perfil para dispositivos de Información Móvil, es decir que el Sistema Operativo fue desarrollado por los mismo Ingenieros de Sony Ericsson. Dado que fue creada por ellos la forma en que está constituido internamente dicho Sistema Operativo es desconocido. Este se encuentra basado en JAVA, esto permite la instalación de programas o aplicaciones desarrolladas en el entono JAVA en la plataforma J2ME o JAVA 2 MICRO EDICION. [JAVMIDP]

Características del Sistema Operativo	**Internet Móvil**	
Sistema operativo Propietario MIDP	wap version wap 2.0 / xHTML browser wap GPRS Clase 10 Navegador HTML NetFront Lector de feeds RSS	
Descarga de aplicaciones Java Java MIDP 2.0		

Tabla. 12 Características del Sistema Operativo Sony Ericsson [SONYDEV]

El K850i presenta la versión de Java Plattaform 8 o JP-8 que posee las características más recientes del Sistema Operativo Sony Ericsson.

Al poseer MIDP como plataforma para este dispositivo móvil ofrece excelente seguridad. La seguridad que se encuentra dentro del modelo programación de JAVA, es decir la máquina virtual de Java

Adicionalmente poseen una interfaz Javax.microedition.pki, la cual permite trabajar certificados de seguridad para autenticar información.
Este Sistema Operativo viene con licencias que evitan que estos sean copiados y alterados. [MIDPJ2ME]

Sistema Operativo Propietario MIDP (MID Profile)	
MIDP 2.0	Perfil de Certificados MIDP X.509
Constitución de MIDP MID Profile	Paquetes de Interfaz de Usuario Persistencia de Paquetes Aplicación del ciclo de vida de un paquete Creación de Paquetes de Red Paquete de Audio Paquete de Clave Publica
Creación de Paquetes de Red	Permite la una conexión limitada a la configuración del dispositivo. Conexión HTTP Conexión HTTPS
cifrado con llave publica	RSA
funciones de Hash	SHA-1
protocolos de comunicación	X.509, TLS 1.0, SSL v3, HTTP, WAP

Tabla. 13 Características del Sistema Operativo Ericsson

6.3.10 Selección del dispositivo entre el sistema operativo Symbian y Sony Ericsson Propietario

Las características que cada sistema operativo presenta se adaptan a los requisitos para la implementación de un aplicativo móvil. para la selección del dispositivo se tiene encuentra la compatibilidad del sistema operativo con aplicaciones desarrolladas en Netbeans, el sistema operativo como tal, el soporte de certificados SSL y conexiones HTTPS.

Se escoge como dispositivo aquel que posee el sistema operativo Sony Ericsson Propietario, ya que este reúne las características necesarias que se estudiaron

para la implementación de la aplicación móvil, adicionalmente, brinda un paquete de librerías más extenso, el cual facilitará que el aplicativo corra sin problemas. Este dispositivo cuenta con un SO totalmente Java y se encuentra diseñado para soportar aplicaciones J2ME, adicionalmente la seguridad que ofrece se encuentra dentro del modelo de programación JAVA.

6.4 Implementación de los algoritmos de cifrado para el dispositivo móvil celular

En la seguridad móvil se debe tener en cuenta varios aspectos, tales como la forma de transmisión segura y el modo de implementación de esta. Los beneficios que ofrece actualmente la tecnología móvil nos permite tener al alcance servicios de Internet Móvil, dando la posibilidad de realizar diversas operaciones transaccionales. Esto se puede llevar a cabo gracias a la tecnología WAP; este en un estándar que permite el acceso a Internet desde terminales móviles. Los creadores de dicho protocolo son Ericsson, Motorola, Nokia y Unwired Planet. [WAPANA]

El objetivo principal de la conexión a Internet móvil es poder comunicarse con el fin de compartir información con usuarios en el mundo sin tener limitaciones de movilidad. La cuestión es que no toda la información que viaja vía Internet la puedan tener al alcance usuarios no deseados.

6.4.1 Arquitectura de Seguridad WAP (Protocolo de Aplicaciones Inalámbricas) [WAPFOR]

Esta arquitectura está diseñada para dar un entorno escalable para el desarrollo de aplicaciones destinadas a dispositivos móviles. Se define como un modelo de capaz, donde cada capa es accesible por la capa superior a ella. [WAPANA]

Aplicación
Sesión
Transacciones
Seguridad
Transporte

Tabla. 14 Modelo WAP [WAPFOR]

WAP es escalable permitiendo que las aplicaciones dispongan de la capacidad que brinda cada pantalla de los dispositivos móviles; recursos según la necesidad.

Dentro del desarrollo de la aplicación se plantean dos posibles soluciones, teniendo en cuenta los algoritmos que puede soportar el dispositivo móvil. [GUIWAP2002]

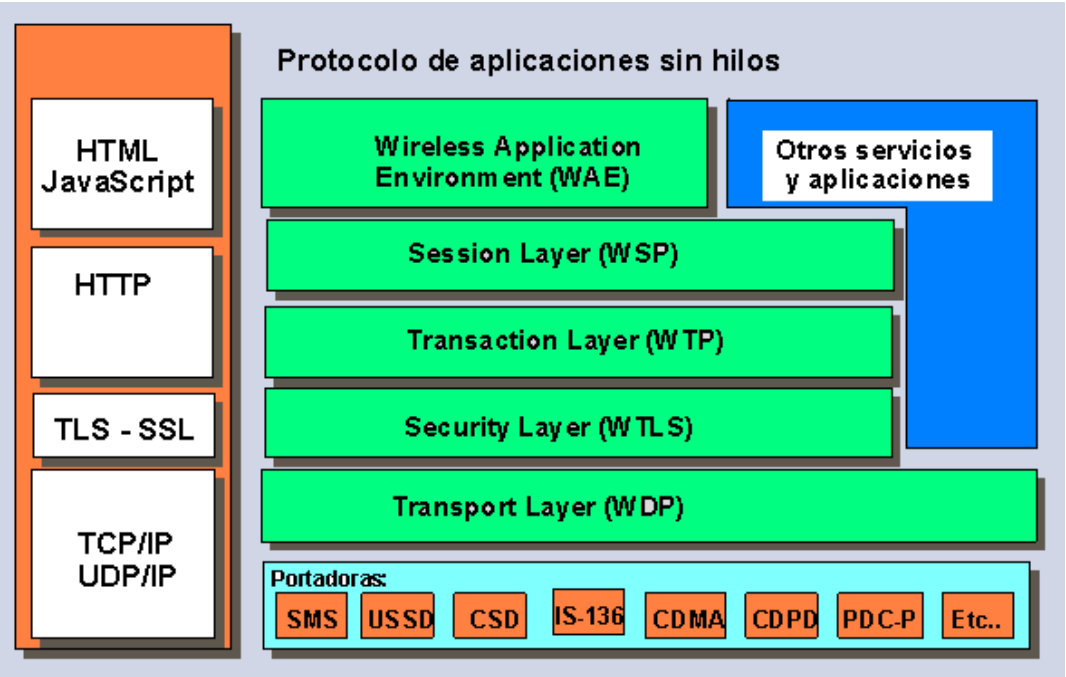


Figura.15 Tecnologías Implementadas en el modelo WAP [WAPFOR]

- Capa WAE (Wireless Application Environment) Entorno Inalámbrico de Aplicaciones.

Su objetivo es construir un entorno de aplicación que permita que los operadores y proveedores construir aplicaciones y servicios destinado a una variedad de plataformas, este entorno se encuentra enfocado a los aspectos del cliente.

El entorno del micro navegador se encuentra compuesto de las siguientes funcionalidades:

- Un lenguaje WML (Wireles –Markup Lenguage) similar al HTML pero a diferencia de este es optimizado para el uso en terminales móviles.
- Un lenguaje WMLScrip

- Capa WSP (Wireless Session Protocol) Protocolo Inalámbrico de Sesión

Provee a la capa de aplicación dos servicios de sesión: un servicio orientado a conexión que se encuentra encima de la capa de transacciones y otro no orientado a conexión que funciona sobre la capa de transporte.

- Permite establecer una conexión fiable entre el cliente y servidor, y terminar la conexión.
- Intercambia contenidos entre cliente y servidor utilizando una codificación compacta.
- Permite suspender y restablecer la sesión.

- Capa WTP (Wireless Transaction Protocol) Protocolo Inalámbrico de Transacción

Se establece para proporcionar los servicios necesarios que soporten aplicaciones de navegación es decir aplicativos del tipo petición y respuesta. Al proceso de petición / repuesta se le llama transacción.

- Utiliza identificadores únicos de transacción, elimina duplicados y retransmisiones
- Seguridad opcional Usuario a Usuario
- Permite transacciones asíncronas

- Capa WTLS (Wireless Transport Layer Security) Capa Inalámbrica de Seguridad de Transporte

Es un protocolo basado en el estándar SSL cuyo protocolo es utilizado para proporcionar seguridad en la transferencia de datos; fue diseñado para los protocolos de transporte WAP y optimizado para comunicaciones de un ancho de banda angosto.

Se definen tres características principales para el protocolo:

- Integridad de los datos
- Privacidad de los datos
- Autenticidad de los datos

- Capa WDP (Wireless Datagram Protocol)

Se comunica de forma transparente sobre las capas de sesión transacción y transporte. Esta capa ofrece servicio a los protocolos superiores, tales como direccionamiento por número del puerto, segmentación re ensamble, detección de errores, algunos de manera opcional.

6.4.2 Funcionamiento de SSL (Secure Sockets Layer) [VERISSL]

SSL Secure Sockets Layer, es un protocolo diseñado por Netscape Comunications Corporation con el fin de brindar seguridad a las sesiones de navegación a través de la red de Internet.

SSL como protocolo de seguridad de transporte proporciona servicios tales como: [VERISEC]

- Confidencialidad

Aquí se encuentra la seguridad por medio del cifrado de datos, se garantiza que la información transferida es indescifrable para personas ajenas al objetivo del mensaje. El cifrado que se utiliza es criptografía de clave simétrica con una clave se sesión que se acuerda mientras se establece la conexión, verificando la identidad de las partes y determinado los parámetros que se utilizaran.

- Autenticación

El usuario puede estar seguro de la identidad del servidor, al cual se conecta y valida con el fin de intercambiar información de manera confiable. Esto se hace mediante certificados basados en criptografía de clave pública. Por lo general el que se autentica es el servidor mediante un certificado digital.

- Integridad

No se permite modificar el mensaje mientras viaja a través de la red de Internet. Esto lo hace utilizando códigos de integridad, los cuales se calculan utilizando HASH (SHA – MD5).

- Autenticación del cliente

Esta es opcional, permite al servidor conocer la identidad del usuario para brindar privilegios de acceso.

Se utiliza SSL en compras, transacciones, cuando se debe facilitar datos personales y bancarios tales como una compra con tarjeta de crédito.

La seguridad de SSL se basa en dos tecnologías; criptografía de llave pública y llave privada. Para intercambio de datos entre cliente y servidor utiliza algoritmos de cifrado asimétrico, tales como DES, Triple – DES, IDEA. Para autenticar y cifrar la clave de sesión utilizada para los algoritmos de intercambio de datos utiliza cifrado de clave pública RSA.

La clave se sesión es la que se utiliza para cifrar los datos que llegan y envían hacia el servidor, cuando ya se haya establecido el canal. Esta clave varia para cada transacción, así aunque sea deducida por un atacante, no le será de utilidad ya que se renueva en cada sesión.

Un certificado de seguridad es un conjunto de documentos electrónicos emitidos por una entidad certificadora de confianza, la función del certificado es cifrar la información antes de enviarla a un servidor, con el fin de hacerla indescifrable para cualquier intruso, además permitiendo identificar la fuente de los mismos. El certificado se debe instalar en el servidor web con el fin de proteger los datos.

La Web que posea el certificado presenta las siguientes características:

- En la barra de direcciones presenta las siglas https, además de un candado visible

Figura. 19 Imagen de indicación de la página con certificado [VERISEC]

- En la parte derecha aparece el certificado indicado por el candado, en la parte izquierda en la barra de direcciones la dirección aparece como **https:**, la "s" indica que pagina utiliza certificados SSL.

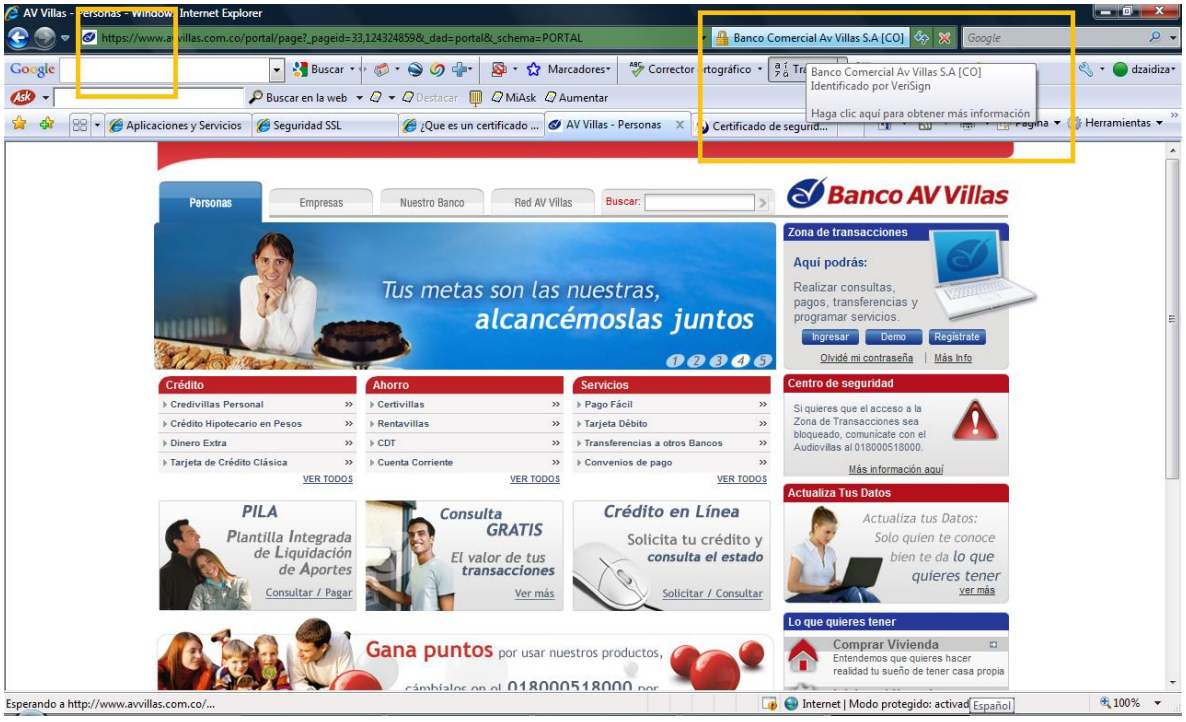


Figura. 20 Campos de indicación de certificado [AVVILLAS]

- Al realizar clic en el certificado se puede observar el certificado de la página y la empresa certificadora de este, para este ejemplo la empresa es VeriSing.

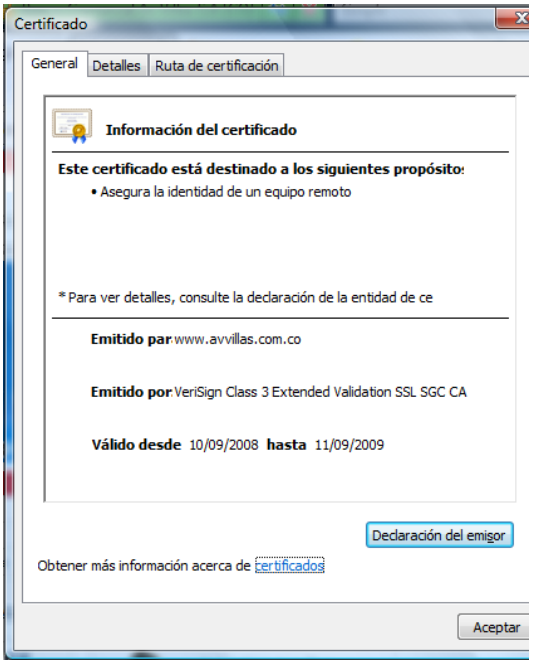


Figura. 21 Certificado en página de VeriSing SSL [AVVILLAS]

6.4.3 Comunicación WAP en aplicativos cliente - servidor [UDECCHI]

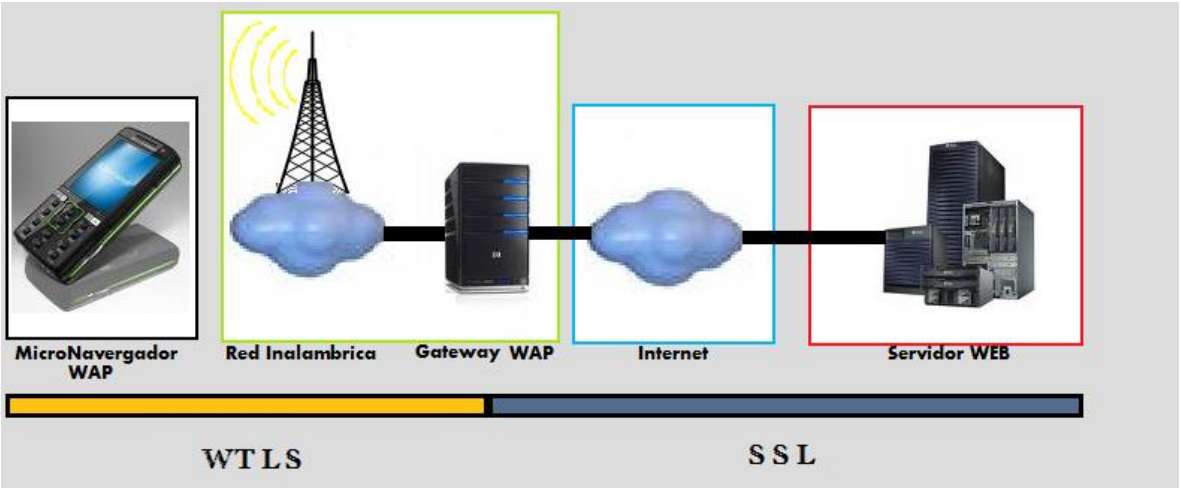


Figura. 22 Diagrama de comunicaciones WAP

El modelo consta de tres partes:

- El Gateway Wap utiliza SSL (Secure Sockets Layer) para realizar la comunicación de manera segura con el servidor WEB. Brindando privacidad, integridad y autenticación del servidor

- Para la comunicación entre el Gateway y el dispositivo móvil celular utiliza WTLS (Wireless Transport Layer Security).

- El puente entre los protocolos SSL y WTLS es el Gateway WAP.

Se realiza el cambio de SSL a WTLS dado que en las comunicaciones inalámbricas, se maneja un ancho de banda reducido. SSL como está diseñado con mayor ancho de banda, si se implementará SSL en dispositivos móviles celulares el costo de una terminal se elevaría de manera exponencial disminuyendo el mercado WAP.

WTLS está diseñado con un nivel de seguridad alto sin requerir de una gran capacidad de procesamiento. El tiempo estimado para pasar de SSL a WTLS oscila en el orden de los mili-segundos, para garantizar la seguridad el servidor WAP o Gateway WAP este no almacena el contenido decodificado.

El proceso de codificación y recodificación se realiza bajo parámetros de seguridad que han sido optimizados en velocidad, con el fin de que el contenido original que se almacena en la memoria volátil sea borrado tan pronto se realiza la conversión. Adicionalmente al proceso de seguridad se restringe el acceso físico a la consola Gateway, en la parte de configuración las restricciones son del orden administrativo.

WTLS, este protocolo es una adaptación del estándar TLS (Transport Layer Security), se toman las características TLS, la aplican a Datagramas que son paquetes pequeños que contienen solo encabezados que es la forma de comunicación WAP, optimizando los paquetes transmitidos y seleccionando algoritmos rápidos.

Soporte de Datagramas
Soporte de portadoras para ancho de banda variable
Soporte de retardos potencialmente largos
Terminales con capacidad de memoria pequeña

Tabla. 15 Definición de WTLS

Para el envió de mensajes seguros por WTLS se requieren servicios:

- Integridad de los datos

Asegura que los datos y intercambiados entres el servidor WAP y la terminal móvil no han sido modificados

- Confidencialidad de los datos

Se garantiza que los datos intercambiados entre el servidor WAP y la terminal no sean entendibles a terceras partes que puedas que puedan interceptar datos.

- Autenticación

Este protocolo contiene servicios que permiten autenticar la terminal y el servidor WAP.

- Protección por denegación de servicios

Se encarga de la protección de las capas superiores en el protocolo WAP, realiza protección contra ataques por denegación y envió de mensajes no comprobados.

6.4.4 Pasarela WAP

La pasarela WAP es la encargada de la conversión del protocolo WTLS a SSL actuando como un mediador entre el dispositivo móvil celular y el servidor WEB http.

El mensaje se cifra mediante TLS desde el servidor, se recibe en el Gateway WAP, donde se descifra el mensaje codificado en formato TLS, convirtiendolo en binario para luego cifrarlo mediante WTLS y enviarlo para ser recibido en el dispositivo móvil y luego ser descifrado.

Al cambiar de protocolo se corre un riesgo de que el mensaje pueda llegar a ser visto ya que hay se realiza un descifrado del mensaje, para esto se tienen barreras de seguridad en el servidor.

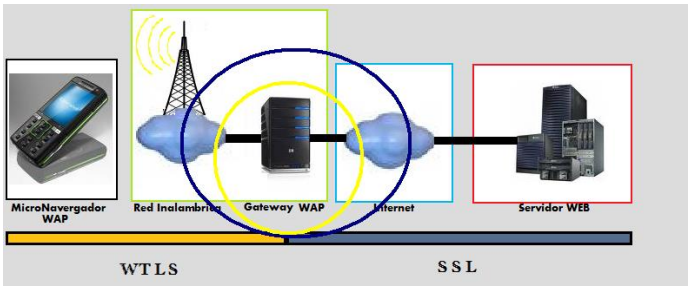


Figura. 23 Diagrama Servidor o Gateway WAP para realizar pasarela

6.4.5 Certificados SSL

Un certificado permite el cifrado de información confidencial durante transacciones en línea, cada certificado contiene información exclusiva y autenticada sobre el propietario del certificado. Una entidad de certificación verifica la identidad del propietario del certificado cuando se emite.

Se utiliza un certificado SSL cuando se tiene una tienda electrónica o se reciben tarjetas de crédito en línea, cuando se inicia la sesión de un sitio web, al procesar datos confidenciales como direcciones, identificaciones nombre. Un certificado establece un canal de comunicación privado permitiendo cifrar los datos durante la transmisión proporcionando la confidencialidad de los datos.

Los certificados SSL constan de una clave pública y una clave privada. La clave pública es utilizada para cifrar la información y la privada para descifrarla. Al conectarse a un dominio desde un navegador WEB una presentación SSL autentica al servidor WEB y al cliente o navegador WEB. Se establece una clave de sesión exclusiva con el fin de iniciar la conexión segura.

- Algunas entidades certificadoras son: [THAWTE] [VERISSL]
  - Thawte
  - VeriSing
  - Certicom
  - ipsCA

- La autenticación

Todos los certificados SSL según la compañía se han creado para un servidor en particular en un dominio específico. Cuando se produce la presentación SSL el navegador exige al servidor que se autentica. La autenticación de la organización se puede observar al hacer clic en el candado cerrado de la ventana.

Un certificado es otorgado por una entidad certificadora al igual como se otorga un permiso de conducir. La autoridad certificado emite Certificados SSL, muchas de estas autoridades solo verifican el nombre del dominio antes de emitir el certificado, otras de estas son más rigurosas al exigir los requisitos.

6.4.6 Criptografía de clave privada y pública

Para poder entender el contenido interno de un certificado a nivel de cifrado se descomponen los algoritmos en los cuales se basa el cifrado.

- DES (Data Encription Standard) algoritmo de cifrado DES
  Ver anexo 1
- Desarrollo del algoritmo de cifrado RSA Publico
  Ver anexo 2

6.5 Desarrollo del aplicativo móvil en Netbeans

En la siguiente imagen se puede encontrar el diagrama de flujo del aplicativo móvil.

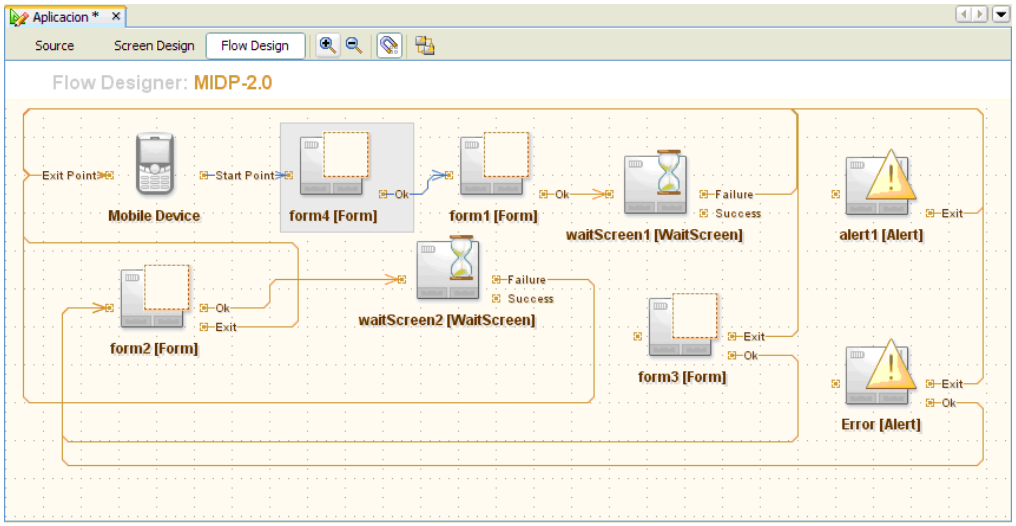


Figura. 24 Diagrama de flujo del aplicativo móvil

En la figura se muestra la creación del primer formulario, en este se da información de presentación del aplicativo.

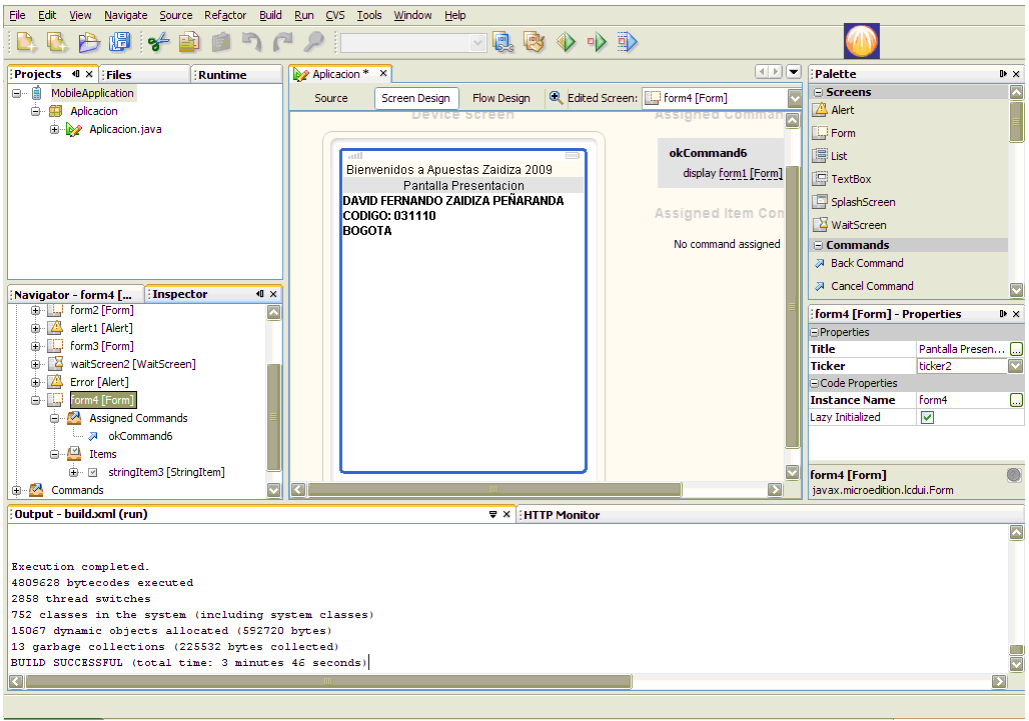


Figura. 25 Formulario de introducción

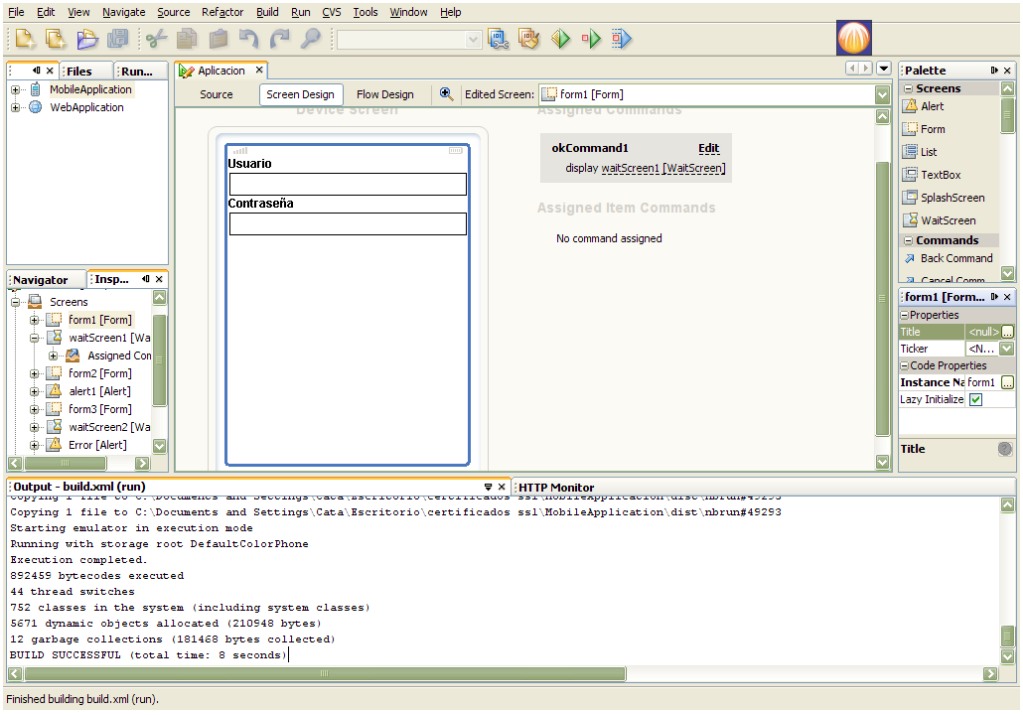


Figura. 26 Formulario de Usuario

En este formulario se realiza se toman el usuario y la contraseña que serán enviados hacia el servidor. Solo avanzara al a siguiente formulario cuando la autenticación haya sido correcta.

```
Conexion();

usuario = textField1.getString();
contrasena = textField2.getString();

byte[]
datos=("usuario="+usuario+"&contrasena="+contrasena+"&indicador=1").getBytes(
);

EnviarDatosLogin(datos);

LecturaLogin();

if(respuesta.indexOf("ConecteBD")!=-1)
    getDisplay().setCurrent(get_form2());
else
    getDisplay().setCurrent(get_alert1());

conn.close();
```

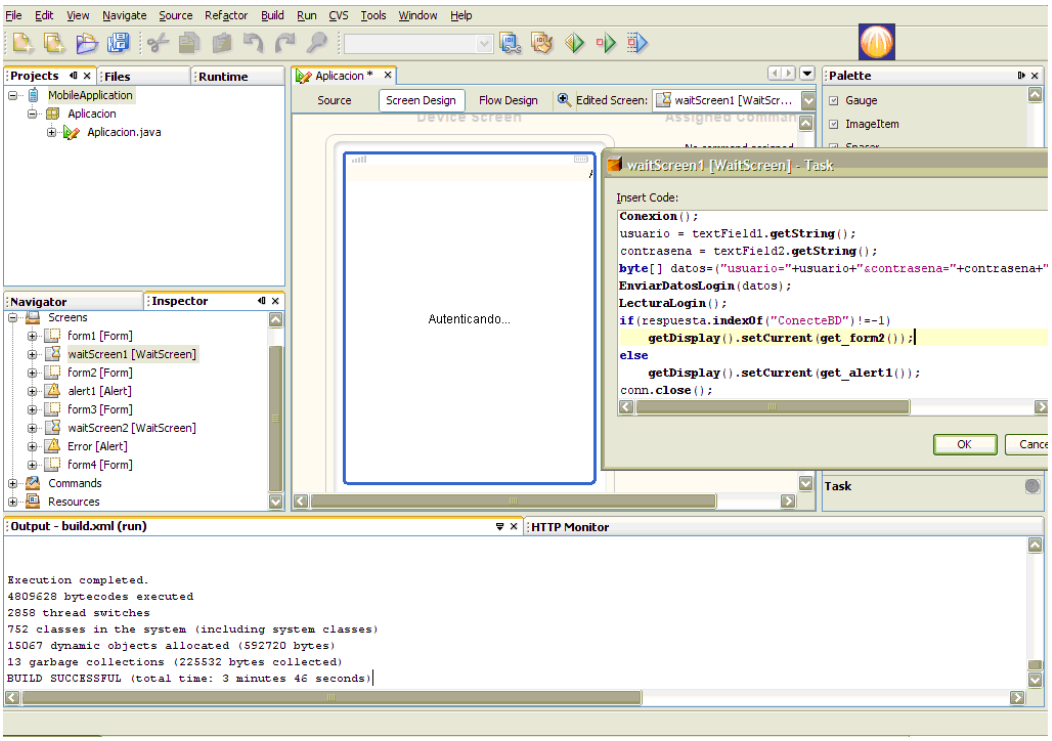


Figura. 27 Formulario de autenticación

Con el siguiente código se establece la conexión segura https hacia el servidor, la instrucción que se encarga de enviar la información segura es HttpsConnection.

```
public void Conexion(){
        String URL ="https://zaidiza.com/Conection.jsp";
        try {
            conn = (HttpsConnection)Connector.open(URL);
            //SecurityInfo si = conn.getSecurityInfo();
            //Certificate c = si.getServerCertificate();
            conn.setRequestMethod(HttpsConnection.POST);
            //conn.setRequestProperty("User-Agent","Profile/MIDP-2.0
Confirguration/CLDC-1.1");
            conn.setRequestProperty("Content-Type","application/x-www-form-
urlencoded");
            //System.out.println("conecte");
        } catch (IOException ex) {
            ex.printStackTrace();
            //System.out.println("No conecte");
        }
    }
```

En la siguiente figura se observa la dirección a la cual se establece la conexión segura https.

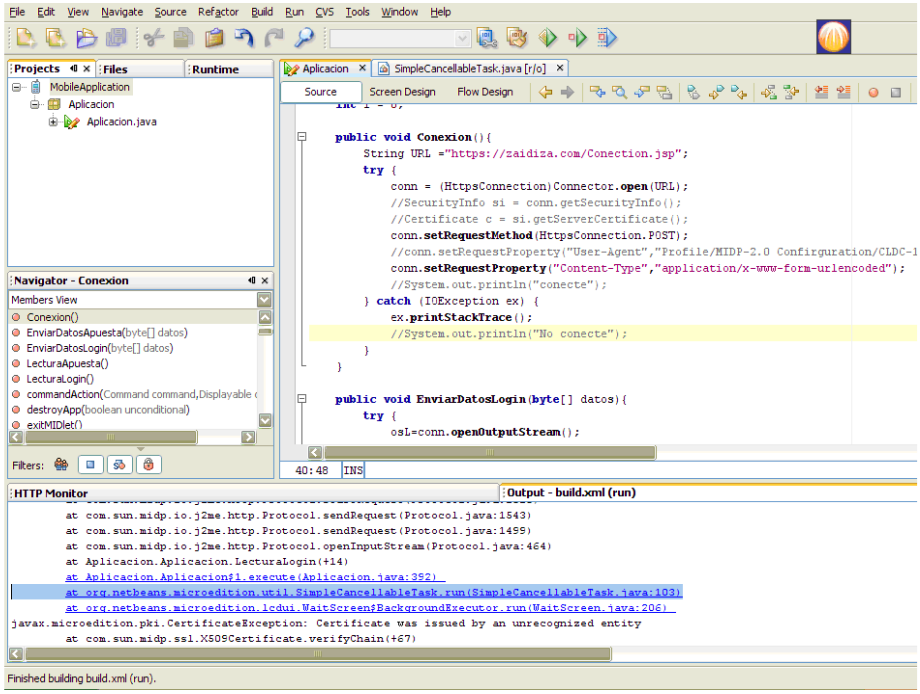


Figura. 28 Formulario de establecimiento de conexión

En la siguiente figura se observa el formulario de espera mientras se establece la conexión segura https.

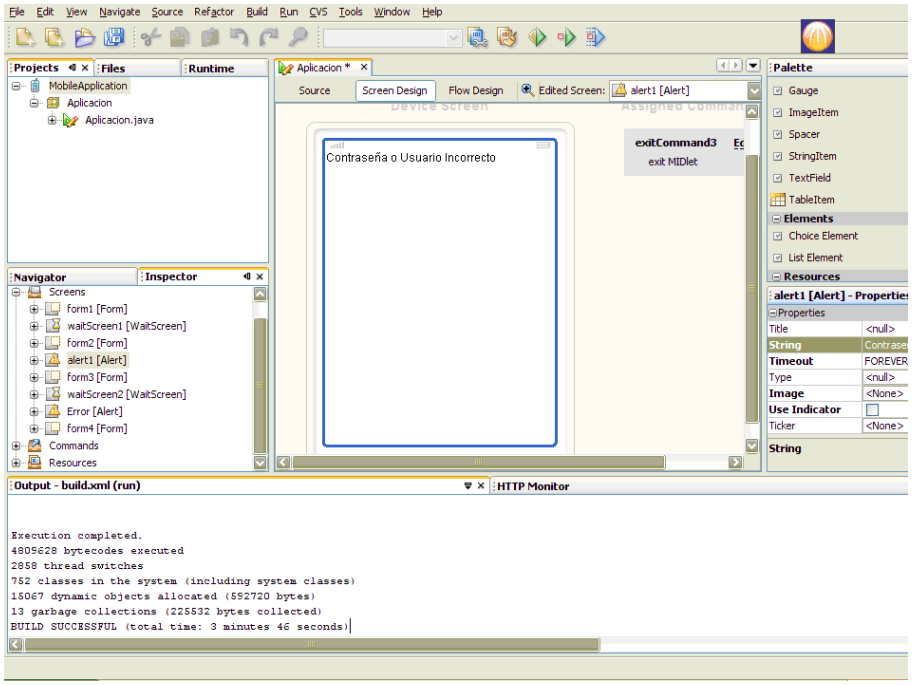


Figura. 29 Formulario de usuario o contraseña incorrecta

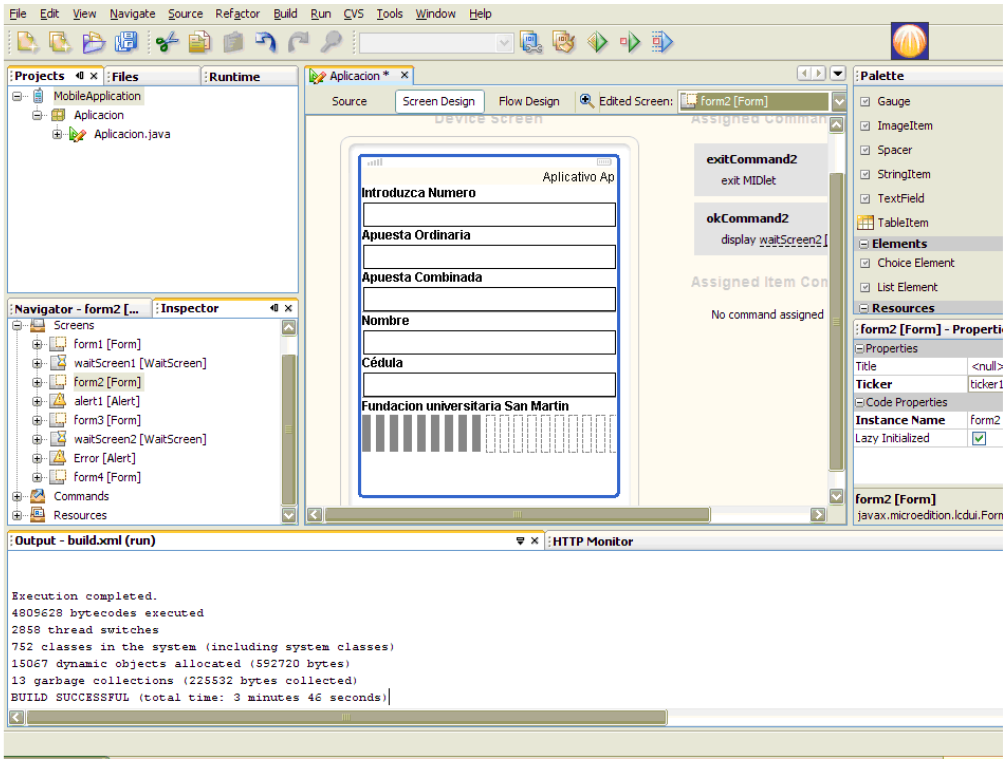


Figura. 30 Formulario al tener una autenticación correcta

Al diligenciar el siguiente el formulario de apuesta, se toman los datos por medio del siguiente código, esto permite realizar el envió de los datos al servidor. Solo hasta recibir el número de identificación desde el servidor, se mostrara el siguiente formulario.

```
Conexion();

Apuesta();

respuesta="";

LecturaApuesta();

int re = respuesta.indexOf("Identificacion=");

System.out.println("valor de re "+re+" lenght "+respuesta.length());

if(re!=-1){
   System.out.println("valor de subs "+respuesta.substring(l, re));
    getDisplay().setCurrent(get_form3());

    stringItem1.setText("");
    stringItem1.setText("valor de subs "+respuesta.substring(re));
```

```
    stringItem4.setText("");
    stringItem4.setText(textField6.getString());

    stringItem5.setText("");
    stringItem5.setText(textField7.getString());

    l = respuesta.length();

} else

   getDisplay().setCurrent(get_Error());

conn.close();
```

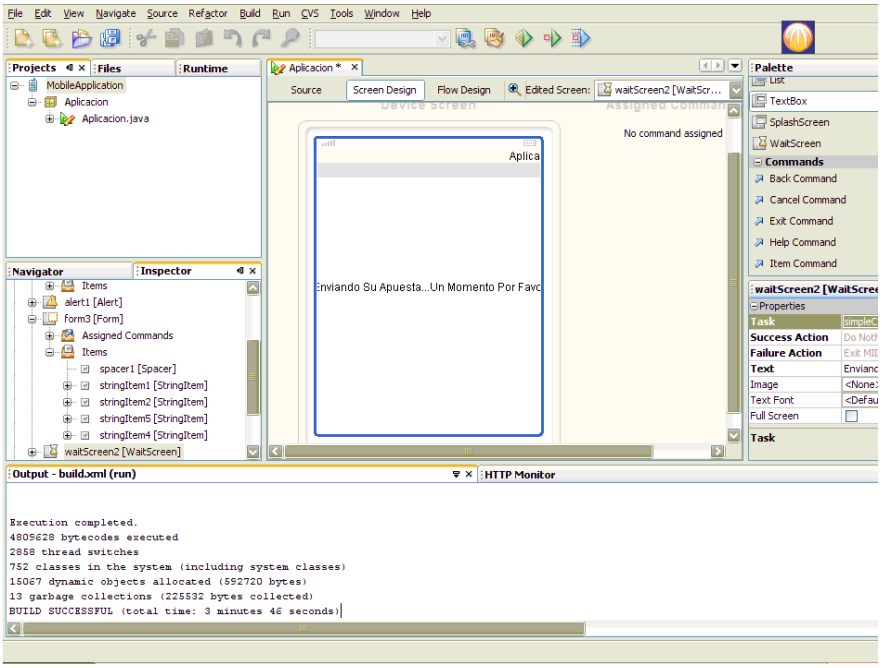


Figura. 31 Formulario al tener una autenticación correcta

6.6 Desarrollo de aplicación para el servidor

En el desarrollo de la conexión hacia el servidor desde Netbeans se deben de tener en cuenta, la configuración del PC donde se encuentra instalado el aplicativo de simulación.

Para poder establecer una conexión https desde el PC se debe configurar el certificado creado por zaidiza.com, con el fin de que no rechace la conexión. En la siguiente figura se valida el certificado para el PC, esta información arrojada al validarlo se puede observar en el servidor de la aplicación.

- Instrucción de configuración del certificado

Se debe introducir la instrucción en la consola de comando de Windows, dentro de la carpeta bin que está contenida dentro de la carpeta JDK.

Keytool –import –alias warsaw –file c:\certificado\zaidiza.cer

Seguido de esto pedirá la contraseña asignada al certificado cuando fue creado “123456”. Al realizar este proceso arrojara finalmente el contenido del certificado como se observa en la figura.

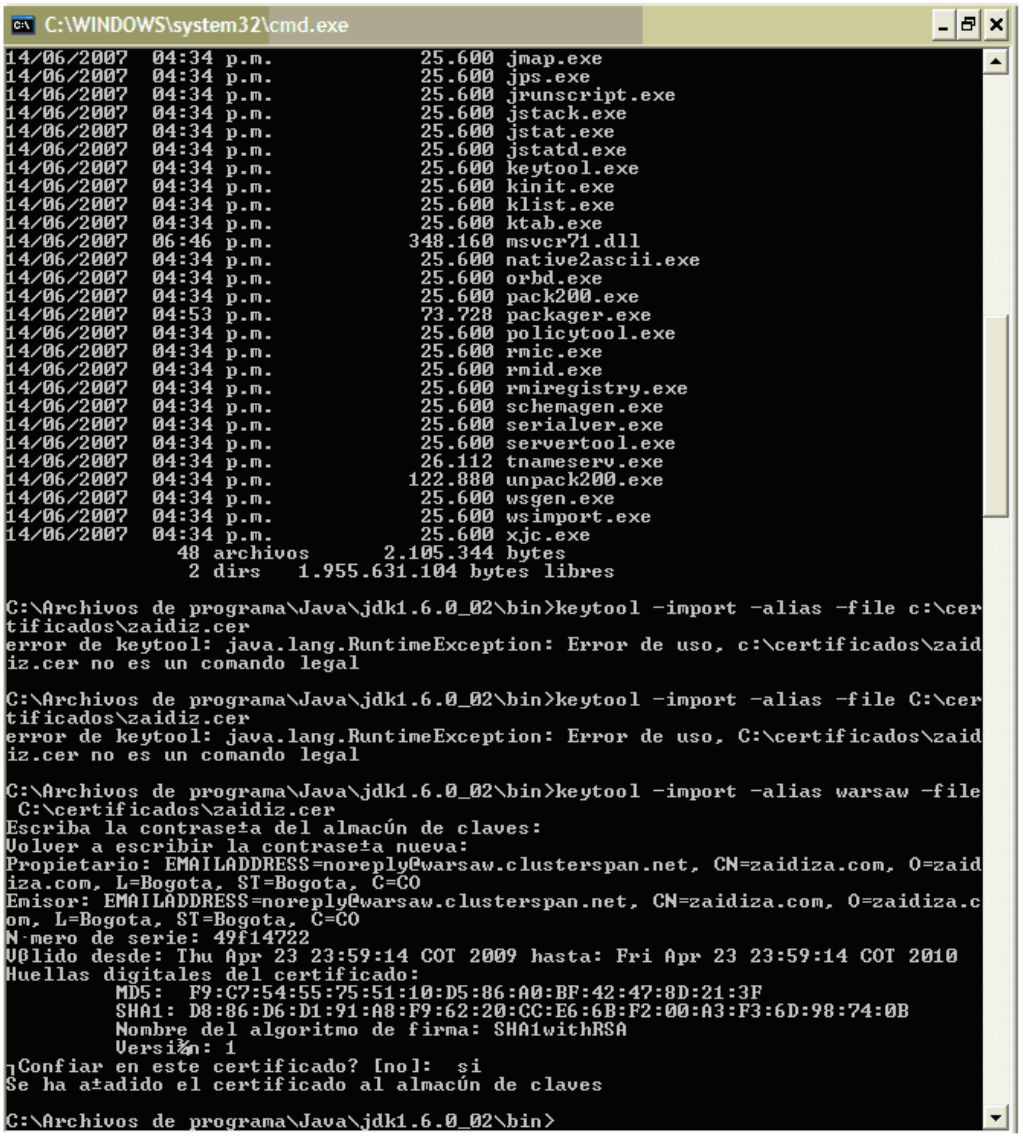


Figura. 32 Configuración de certificado en PC para poder establecer conexión https.

En el siguiente fragmento de código se observa como el servidor obtiene el usuario y la contraseña para poder realizar la validación y permitir establecer la conexión.

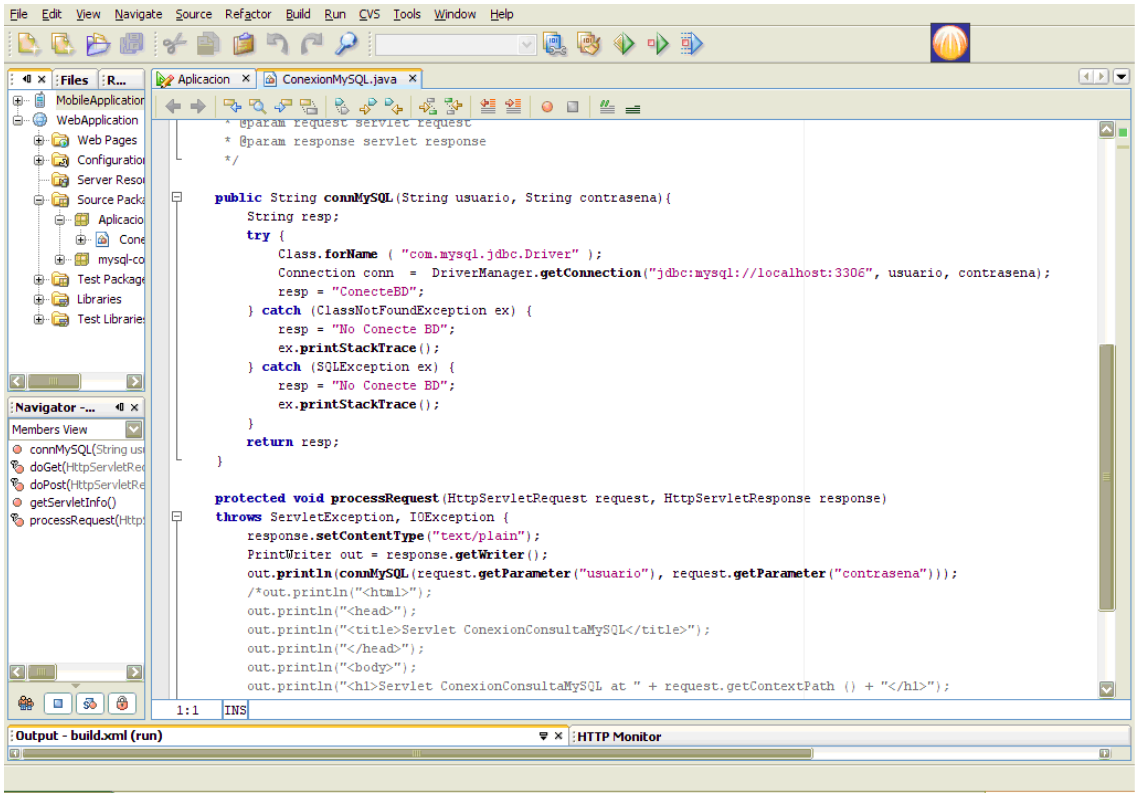


Figura. 33 Validación de datos en servidor

```
public String connMySQL(String usuario, String contrasena){
        String resp;
        try {
            Class.forName ( "com.mysql.jdbc.Driver" );
            Connection conn  =
DriverManager.getConnection("jdbc:mysql://localhost:3306", usuario, contrasena);
            resp = "ConecteBD";
        } catch (ClassNotFoundException ex) {
            resp = "No Conecte BD";
            ex.printStackTrace();
        } catch (SQLException ex) {
            resp = "No Conecte BD";
            ex.printStackTrace();
        }
```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

```
    throws ServletException, IOException {
        response.setContentType("text/plain");
        PrintWriter out = response.getWriter();
        out.println(connMySQL(request.getParameter("usuario"),
request.getParameter("contrasena")));
```

6.6.1 Implementación en el servidor

Inicial mente al ingresar al servidor se debe realizar una validación de datos como se indica en la figura.

Pagina de conexión
URL: https://warsaw.clusterspan.net:8443/login.php3

Usuario: zaidiza
Contraseña: fwHRCMUJVA

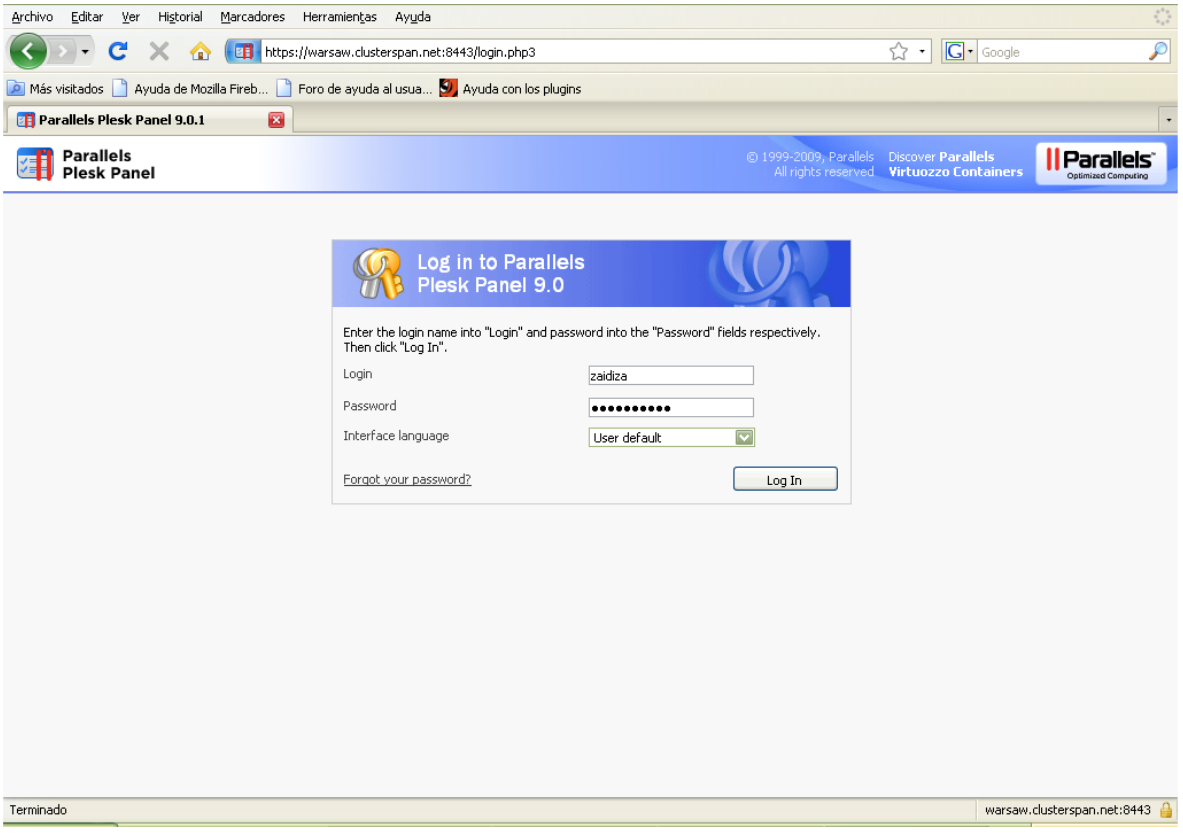


Figura. 34 Validación usuario y contraseña en el dominio

En la siguiente figura se pueden ver los usuarios creados. El formato de un usuario creado es ” zaid_**** ”.

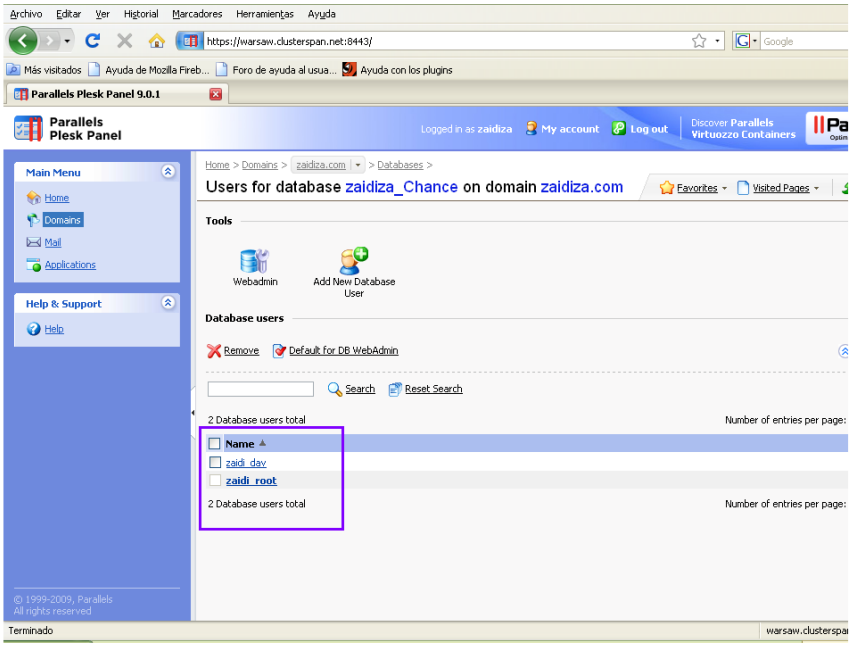


Figura. 35 Usuarios registrados en el servidor

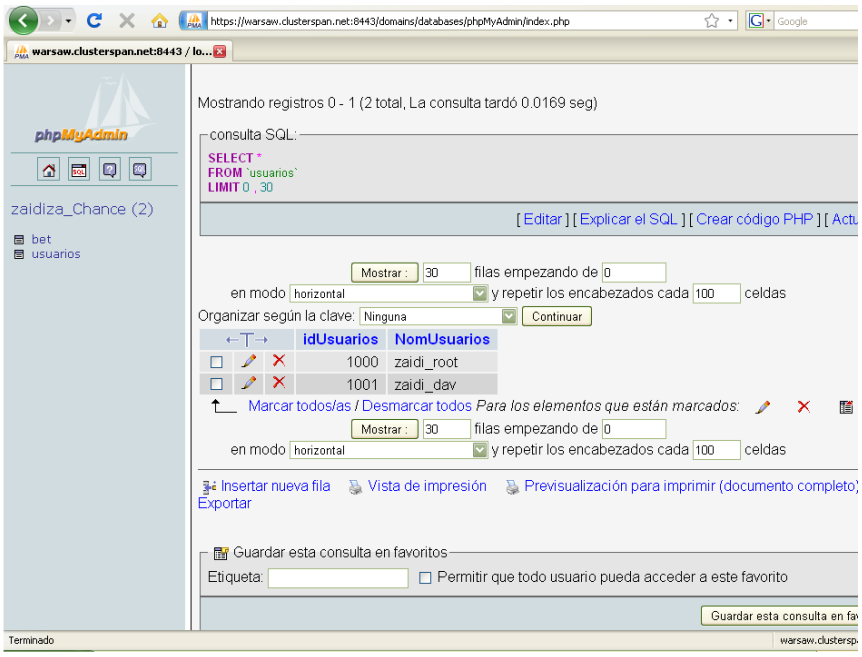


Figura. 36 Usuarios registrados en el servidor en la base de datos

En la siguiente figura se puede observar el historial de apuesta de un usuario registrado, además el número de confirmación correspondiente a cada una de las apuestas realizadas por el usuario.

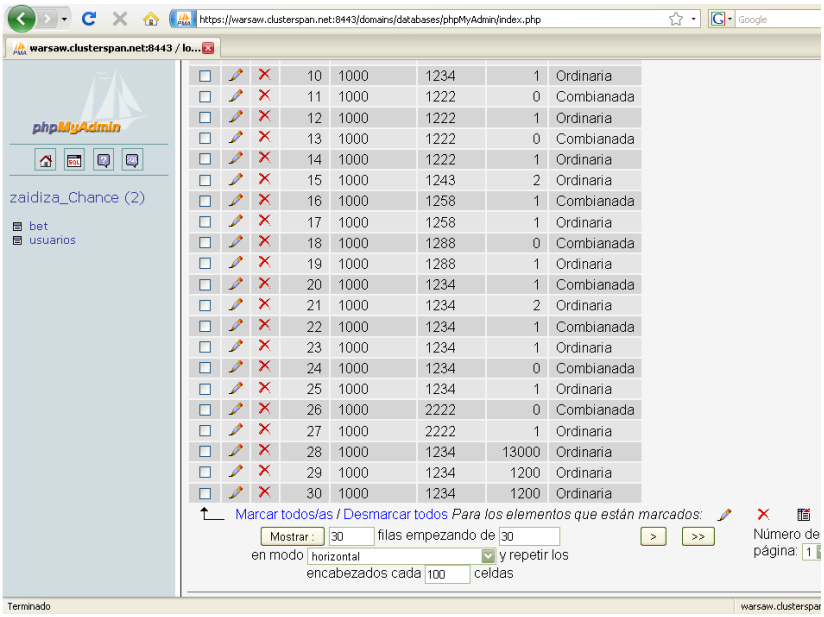


Figura. 37 Historial de apuestas del usuario zaidi_root

7. PRUEBAS Y RESULTADOS

7.1 Plan de pruebas

Para este plan se realizan una secuencia de pruebas para verificar si los aplicativos responden positivamente a lo esperado.

Las pruebas a realizar llevarán un nombre descriptivo, un objetivo que describe el porqué se hace esta prueba, los pasos que se deben seguir en el dispositivo móvil para que la prueba tenga éxito, los resultados que se esperan conseguir, como también los resultados obtenidos en la práctica, además se incluirán fotos de la practica y finalmente se obtendrá una conclusión por cada prueba.

7.1.1 Campo de usuario o de contraseña errónea.

En esta prueba de simulación se puede observar la primera prueba del aplicativo desde un simulador Netbeans, seguido de esto se muestra la implementación de dicho aplicativo en el dispositivo móvil escogido, el móvil K850i de Sony Ericsson.

La primera prueba trata de una conexión al servidor, donde se muestra que el usuario es incorrecto, esto se debe a dos razones la primera es que aun no está conectado al servidor como tal y para evitar que sea violado por prueba y error arroja el mensaje de contraseña incorrecta, como segunda medida que el autenticar no se valido al hacer la autenticación en el servidor.

El objetivo de esta prueba es ver el resultado que se obtiene al introducir un usuario o una contraseña incorrecta en la aplicación.

Los pasos a seguir en el dispositivo móvil son:

- Pulsar Menú
- Luego se busca y selecciona el icono Organizador
- Siguiendo con la ubicación e ingreso al icono Aplicación
- Inmediatamente se elige el icono MobileApplication
- Aparece la Pantalla de Presentación, en esta se oprime OK
- Se introduce una de las siguientes opciones; usuario erróneo contraseña correcta, usuario correcto contraseña erróneo, usuario erróneo contraseña erróneo, luego se oprime el icono Aceptar y enseguida se pulsa OK
- Se le da la opción SI en la siguiente pantalla
- Se espera el resultado de autenticación de usuario y contraseña

El resultado que se espera después de seguir los pasos anteriores es tener como respuesta de la autenticación “Contraseña o Usuario Incorrecto”.

Después de realizadas las pruebas prácticas, el resultado obtenido es un menaje con la siguiente información “Contraseña o Usuario Incorrecto”.

Figura. 38 Validación de usuario y contraseña errónea en dispositivo móvil

La presentación de la conexión inicialmente se simula en la herramienta de desarrollo Netbeans, en las siguientes capturas se puede observar la simulación de la prueba de conexión con usuario erróneo, demostrando que lo implementado fue desarrollado y probado antes de llevarse al dispositivo móvil celular.

Figura. 39 Validación de usuario y contraseña errónea en simulación Netbeans

Con este resultado se puede concluir que el aplicativo implementado detecta si los campos de Usuario o Contraseña están vacíos o contiene información errónea.

7.1.2 Campos para ingresar Números, Apuestas Ordinarias y Apuestas combinadas con información errónea.

El objetivo de esta prueba es ver el resultado que se obtiene en la aplicación al introducir información incorrecta o dejar los campos Introduzca Numero, Apuestas Ordinarias o Apuestas Combinadas incompletos.

Los pasos a seguir en el dispositivo móvil son:

- Pulsar Menú
- Luego se busca y selecciona el icono Organizador
- Siguiendo con la ubicación e ingreso al icono Aplicación
- Inmediatamente se elige el icono MobileApplication
- Aparece la Pantalla de Presentación, en esta se oprime OK
- Se edita usuario y contraseña correcta, luego se oprime el icono Aceptar y enseguida se pulsa OK
- Se le da la opción SI en la siguiente pantalla
- Se espera el resultado de autenticación de usuario y contraseña
- Se edita los campos de Introduzca Numero y Apuestas Ordinarias o Apuestas Combinadas con algún error.
- Luego se oprime el icono Aceptar y enseguida se pulsa OK

El resultado que se espera después de seguir los pasos anteriores es tener como respuesta un mensaje que diga “Error en el envío de datos… Por favor intente más tarde”.

Después de realizadas las pruebas practicas el resultado obtenido es un menaje con la siguiente información “Error en el envío de datos… Por favor intente más tarde”.

Figura. 40 Validación de datos al realizar una apuesta mal diligenciada en dispositivo móvil

Presentación de la conexión Simulada en Netbeans, al tratar de realizar una apuesta con los campos inválidos.

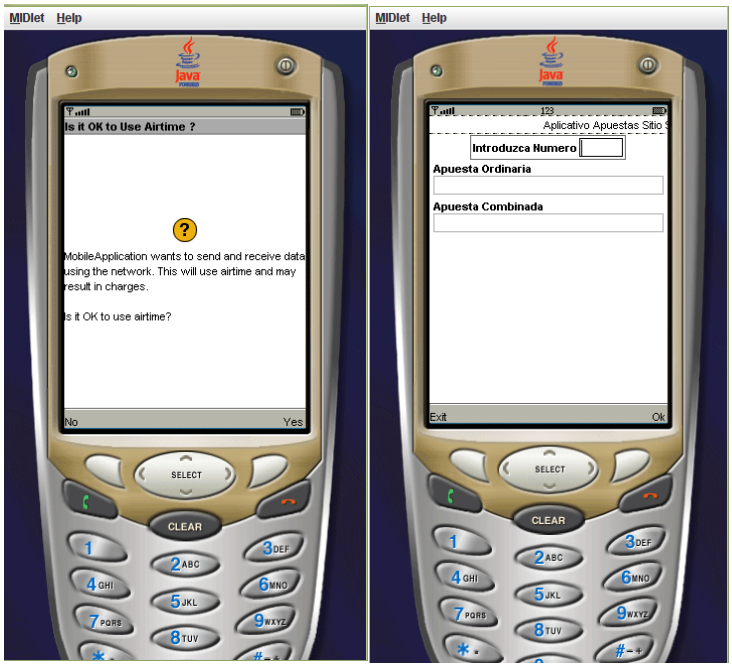


Figura. 41 Validación de datos al realizar una apuesta mal diligenciada en simulador Netbeans

Con estos resultados se puede concluir que el aplicativo implementado detecta si los campos Introduzca Numero y Apuestas Ordinarias o Apuestas Combinadas están vacíos o contiene información errónea y además se confirma al usuario con un número de identificación.

7.1.3 Procedimiento de apuesta correcto.

El objetivo de esta prueba es ver el resultado que se obtiene en la aplicación al introducir información correcta en todos los campos del aplicativo.

Los pasos a seguir en el dispositivo móvil son:

- Pulsar Menú
- Luego se busca y selecciona el icono Organizador
- Siguiendo con la ubicación e ingreso al icono Aplicación
- Inmediatamente se elige el icono MobileApplication
- Aparece la Pantalla de Presentación, en esta se oprime OK
- Se edita usuario y contraseña correcta, luego se oprime el icono Aceptar y enseguida se pulsa OK
- Se le da la opción SI en la siguiente pantalla
- Se espera el resultado de autenticación de usuario y contraseña
  - Usuario: zaidi_root
  - Contraseña: 12345

- Se edita los campos de “Introduzca Numero” y “Apuestas Ordinarias o Apuestas Combinadas” introduciendo los dígitos en el campo de apuesta.
- Luego se oprime el icono Aceptar y enseguida se pulsa OK
- Segundos más tarde se recibe un mensaje de verificación y advertencia “Advertencia no pierda su número de apuesta que con el cobra”.

El resultado que se espera después de seguir los pasos anteriores es tener como respuesta un mensaje que diga “Advertencia no pierda su número de apuesta que con el cobra”.

Después de realizadas las pruebas prácticas el resultado obtenido es un menaje con la siguiente información “Advertencia no pierda su número de apuesta que con el cobra”.

Figura. 42 Validación de datos al realizar una apuesta diligenciada correctamente en dispositivo móvil

Simulación en Netbeans al introducir información correcta en el aplicativo

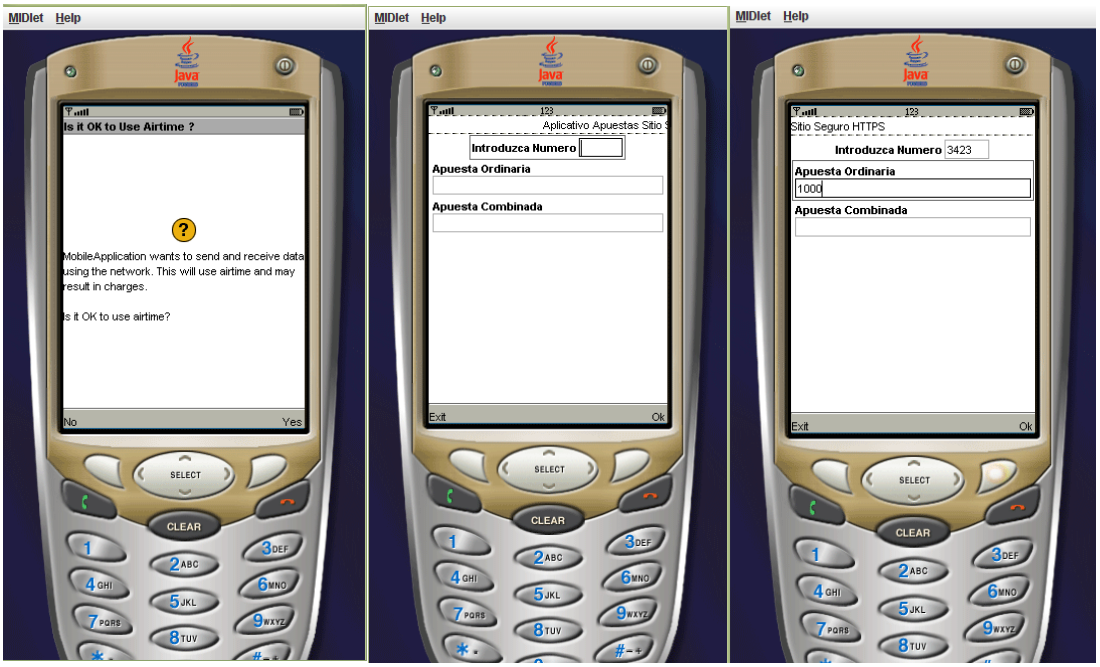


Figura. 43 Validación de datos al realizar una apuesta diligenciada correctamente en simulador Netbeans

Con estos resultados se puede concluir que el aplicativo implementado en el dispositivo móvil para realizar una apuesta por medio de este funciona sin ningún problema, haciendo el envío de datos.

7.1.4 Prueba del certificado SSL en sitio WEB

Al implementar el certificado SSL se requiere de una empresa certificadora, este servicio cuesta alrededor de $ 300000 pesos colombianos. En esta prueba la empresa certificadora es el mismo dominio zaidiza.com. Debido a que esta no es una empresa avalada el certificado aparecerá como no valido, informando que es riesgoso. Aun así el certificado implementado cumple con las funciones mencionadas de un certificado SSL.

En la siguiente figura se observa la advertencia que muestra que el certificado de seguridad no es válido por que está firmado por si mismo, para poder ver el certificado imprentado se debe añadir una excepción. Se deben realizar los pasos como se indica en cada figura, una vez realizados se obtendrán la venta donde se indica el certificado.

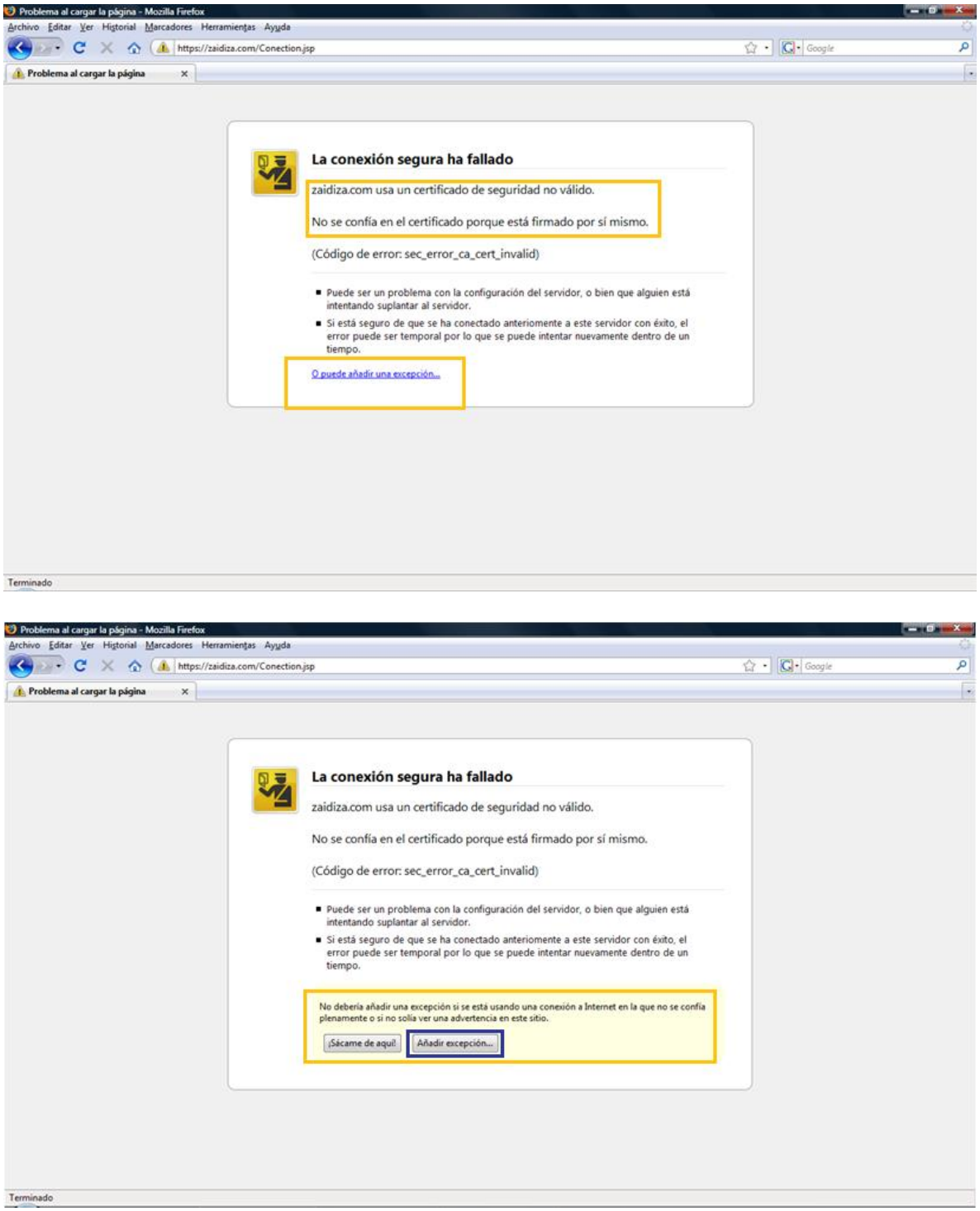


Figura. 44 Configuración para permitir visualización del certificado

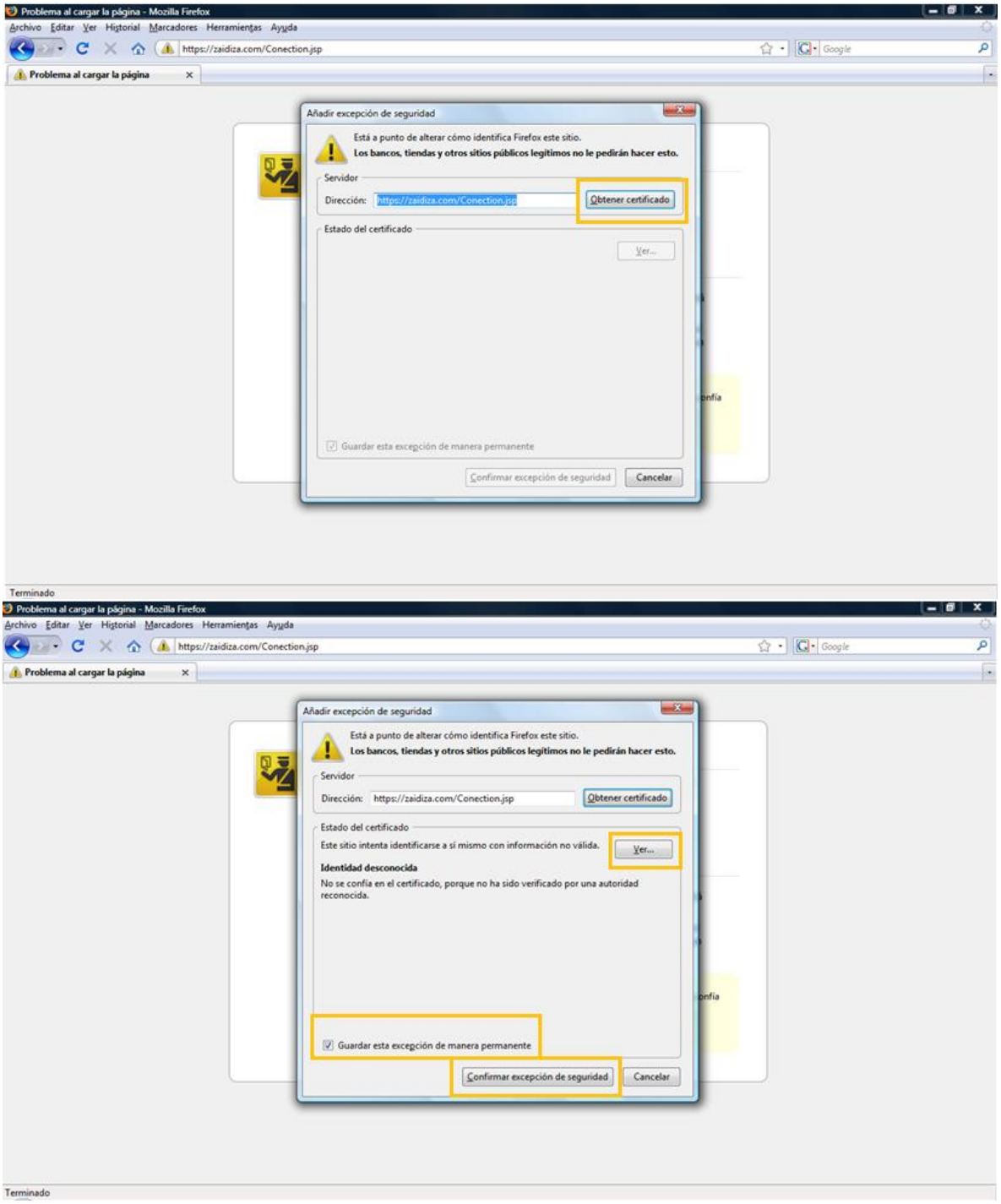


Figura. 45 Configuración para permitir visualización del certificado y recordar excepción

Al confirmar la excepción de seguridad nos muestra el visor del certificado, en este momento se observa el contenido de este.

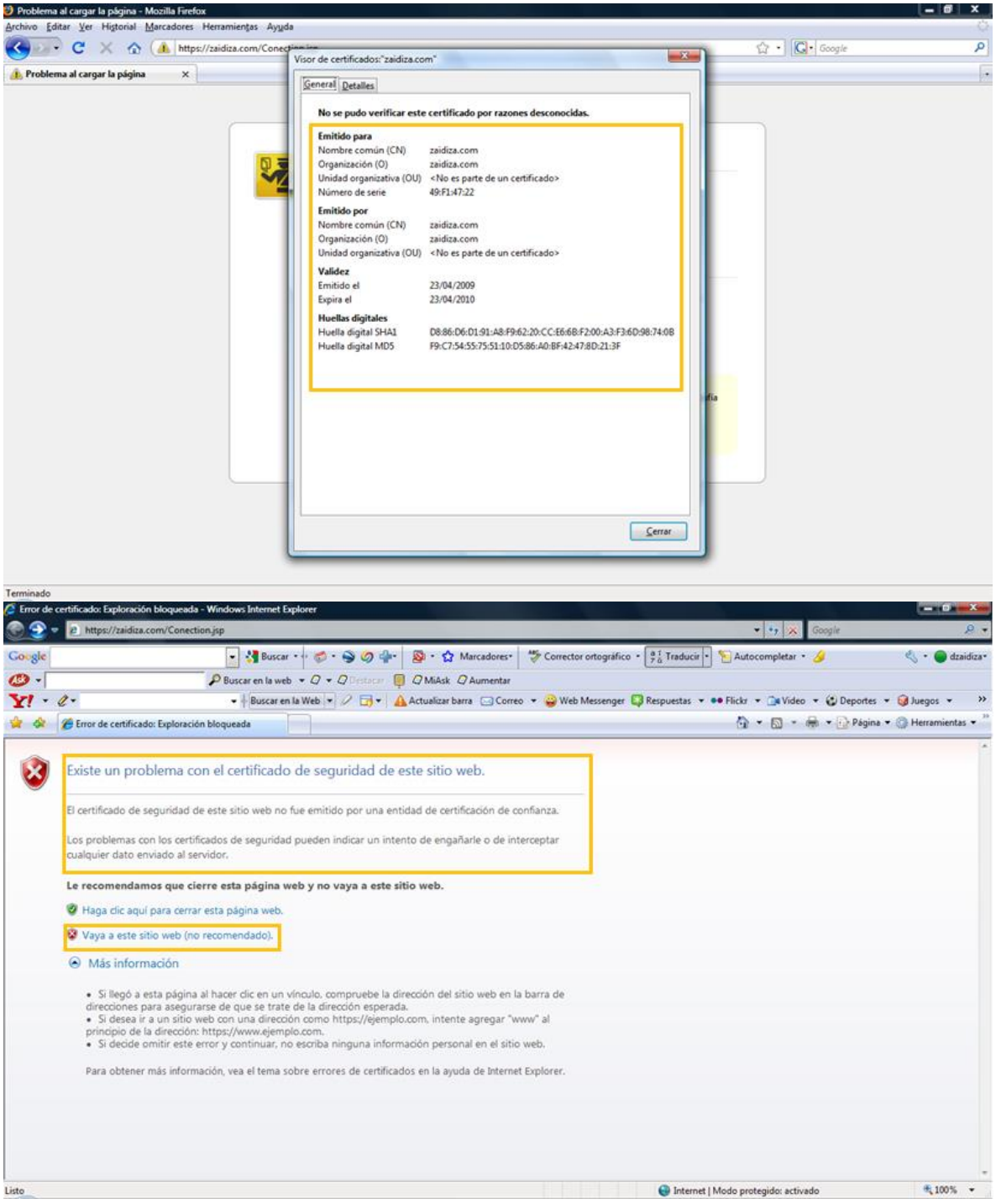


Figura. 46 Visualización del certificado

En la siguiente figura se observa el nivel de cifrado del certificado, al hacer clic en el vínculo ver certificado, muestra información acerca de este.

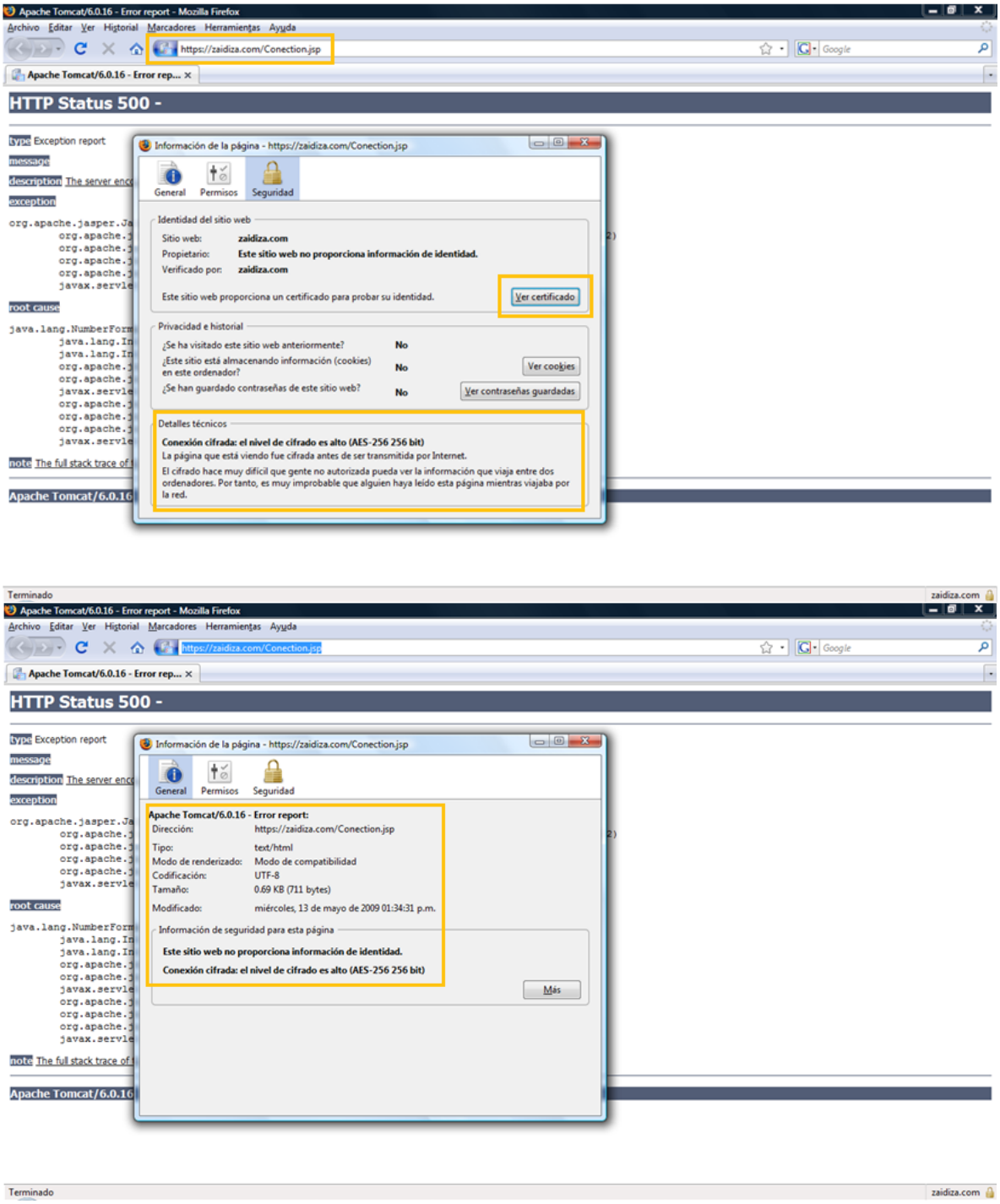


Figura. 47 Nivel de cifrado y contenido del certificado

En los detalles dentro del certificado se encuentra la versión de este, en este caso la número 1, el número de serie del certificado, el algoritmo de firmado, quien emite el certificado, la fecha de creación y expiración, el algoritmo de cifrado de clave pública, la clave publica utilizada, el algoritmo con el cual se realiza el firmado y finalmente el valor de la clave de firmado.

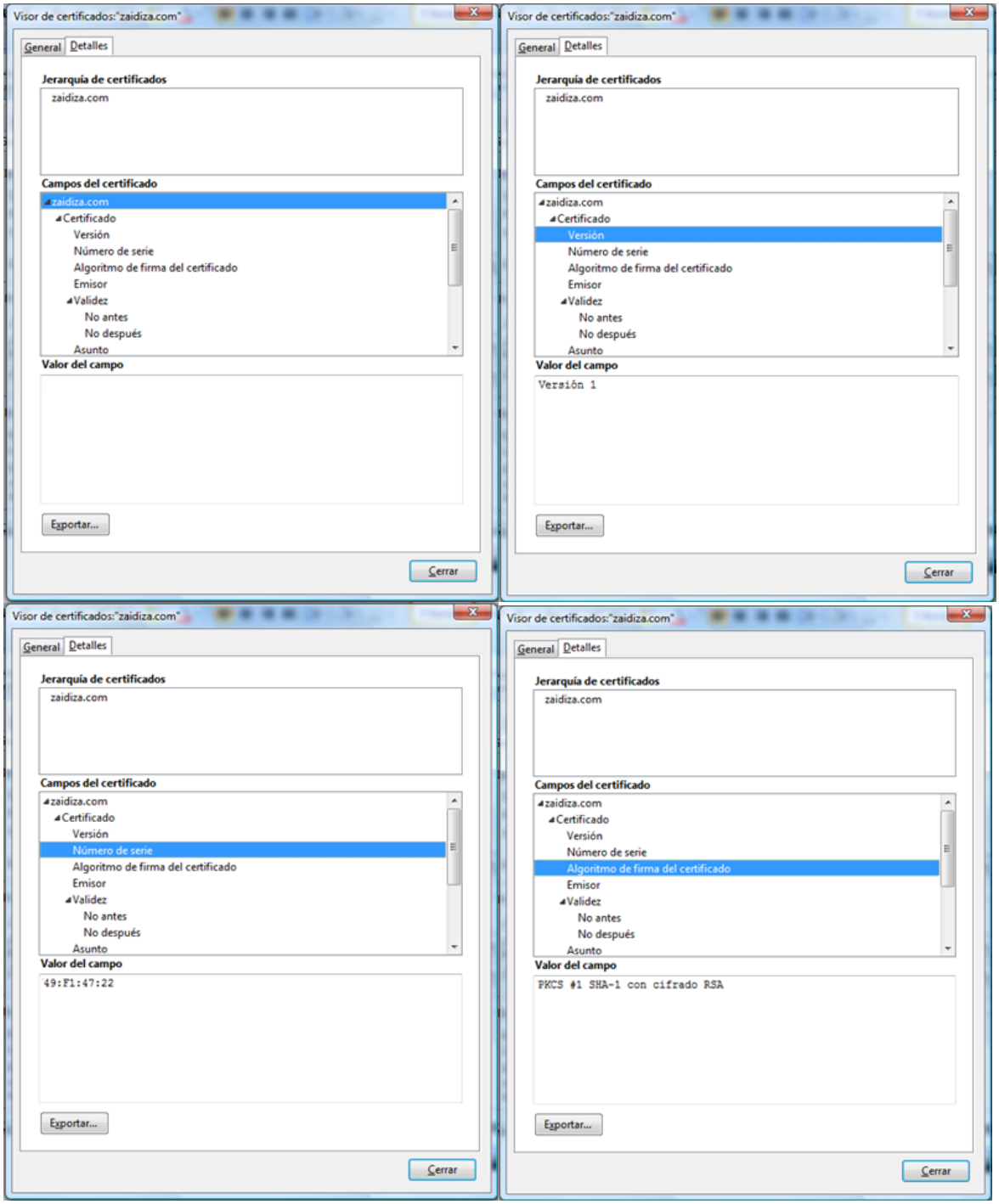


Figura. 48 Detalles del certificado y firmado

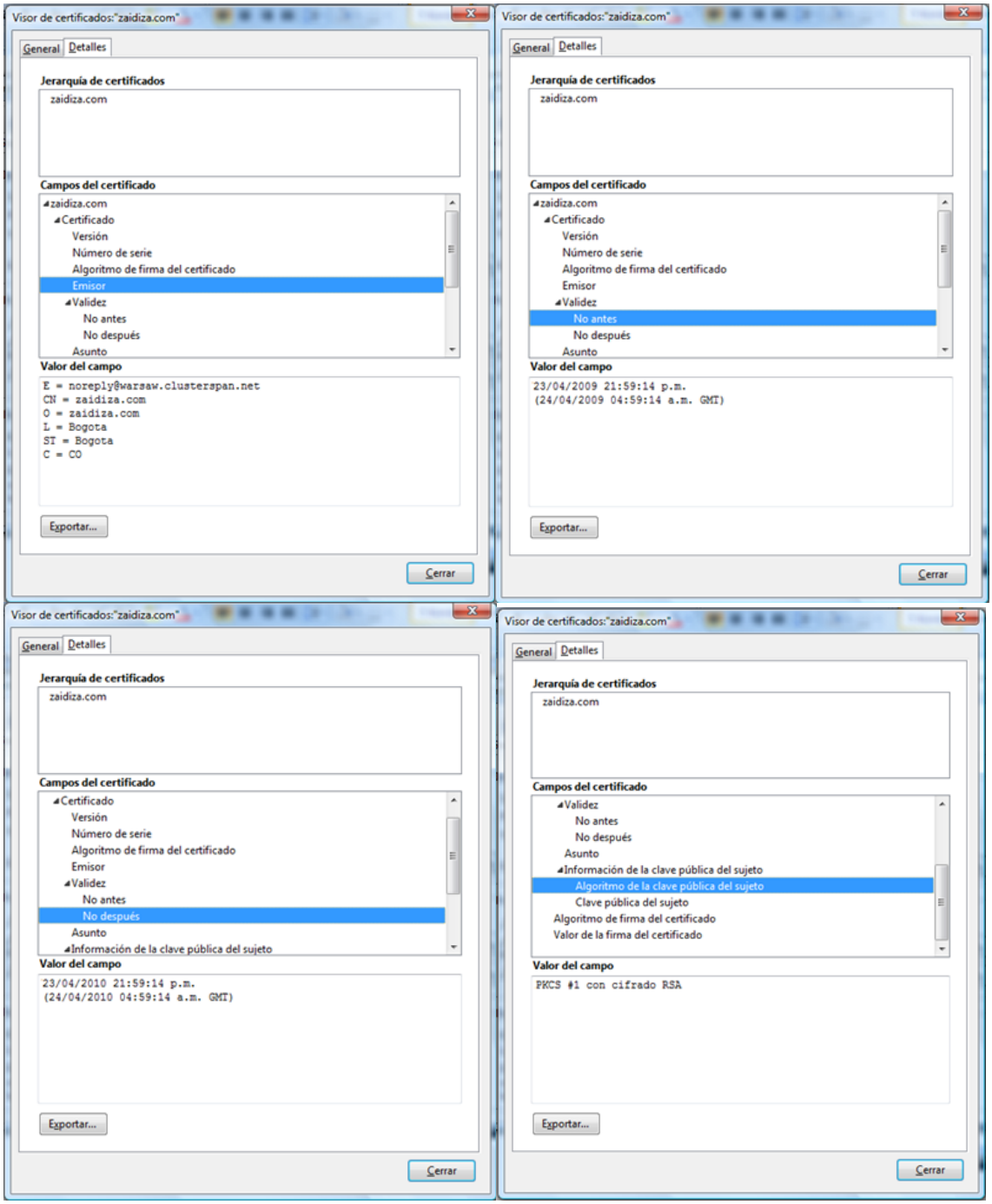


Figura. 49 Detalles del certificado y algoritmo de cifrado de clave publica

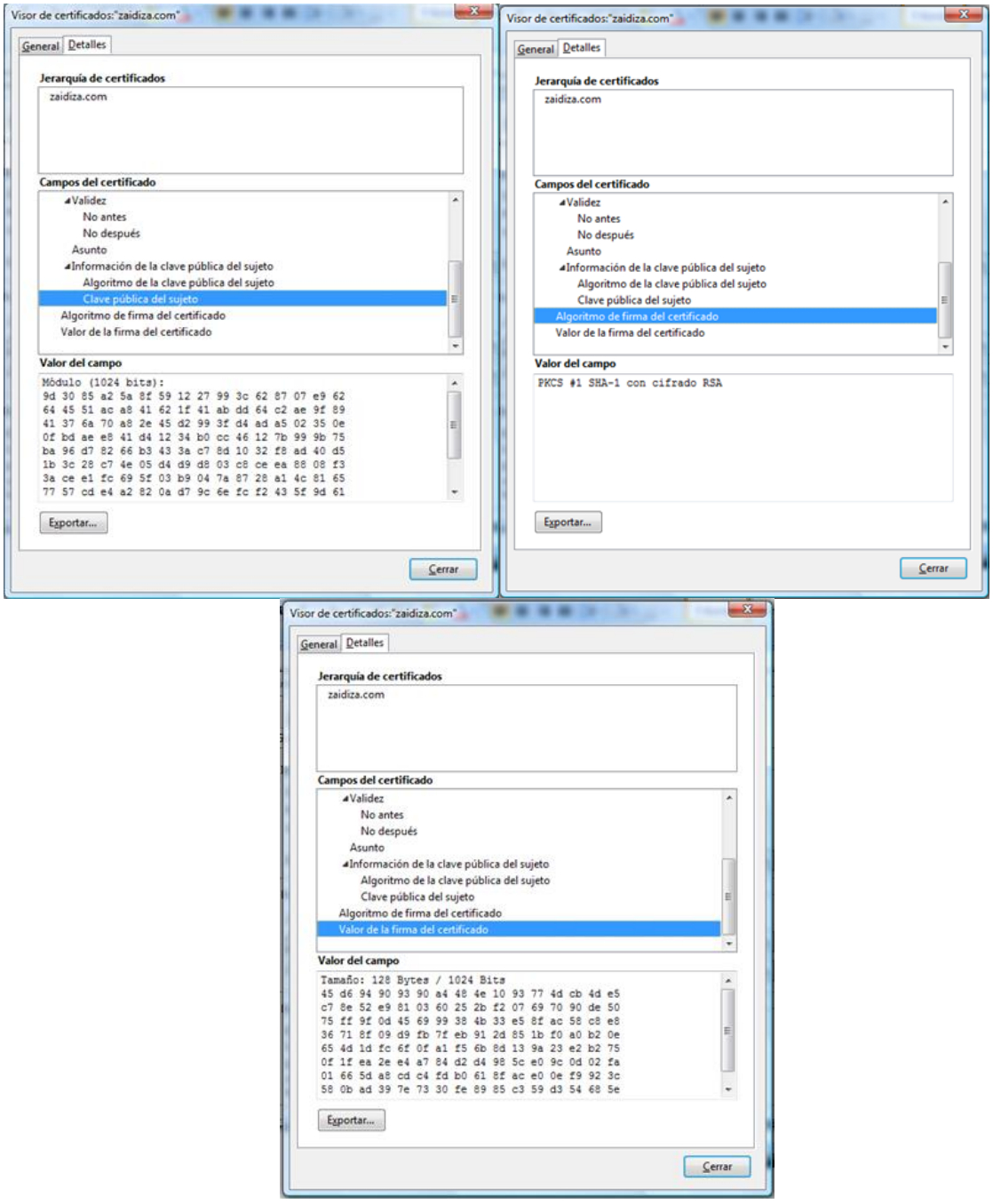


Figura. 50 Detalles del certificado y llaves implementadas

8. CONCLUSIONES

- Existen gran variedad de algoritmos de cifrados, pero no todos son factibles de implementar en dispositivos móviles. La implementación de los algoritmos depende del fabricante del sistema operativo, ya que este define las características del dispositivo; seguridad, adaptabilidad, los paquetes de desarrollo y hacia qué mercado está enfocado el dispositivo.

- La determinación del dispositivo para la implementación de una aplicación prototipo móvil, depende de las características y configuraciones que el sistema operativo brinde al desarrollador. Para el caso de este proyecto, el sistema operativo más óptimo es el Sony Ericsson Propietario.

- Al implementar un canal seguro no basta solo con cifrar la información, se requiere de una entidad que permita autenticar que la información que ha llegado al destinatario por medio de la red de datos celular.

- Al determinar la forma de transmisión segura se garantiza que los datos llegan de manera confiable, íntegra y autenticada. Esto se logra por medio del certificado SSL que garantiza tres aspectos, confidencialidad por medio de los algoritmos de cifrado, autenticación identificando el servidor e integridad de los datos no permitiendo que los datos sean modificados ni leídos por terceros.

- Teniendo en cuenta que existe gran variedad de dispositivos móviles en el mercado, se concluye que la unificación de un sistema operativo aún está lejos, las diferencias de diseño, perfil, requerimientos y características de cada compañía aún no permiten crear un perfil único.

- El desarrollo del aplicativo móvil demuestra que la transmisión de datos seguros por medio de dispositivos móviles en la red celular es posible.

9. RECOMENDACIONES

AL realizar una conexión segura se debe tener en cuenta como primer campo los diferentes sistemas operativos existentes en el mercado, ya que será uno de los parámetros para escoger el dispositivo donde se implementará el aplicativo. El sistema operativo define compatibilidad y adaptabilidad del mismo, según este se puede ver que tan viable es desarrollar una aplicación para un móvil.

Al escoger la forma de envió de datos se debe tener en cuenta que no solo se debe garantizar que los datos viajen seguros si no que adicionalmente se debe poder ofrecer datos confiables, no modificados por terceros y que provengan de una fuente segura y certificada.

Los servicios que ofrezcan el aplicativo se definen según el cliente, ya que el objetivo principal de este proyecto de grado es el envió de datos de manera segura por medio de un dispositivo móvil.

En una transmisión segura con certificados SSL se garantizan tres aspectos, confidencialidad, autenticación e integridad de los datos.

10. TRABAJO FUTURO

Al desarrollo de este proyecto de grado se puede plantear las proyecciones a futuro:

Desarrollo de aplicativos que permita tener funcionalidad en empresas según sus requerimientos. Garantizando así un canal ciento por ciento seguro, además de ser certificado.

Implementar la una combinación de algoritmos que permita garantizar el envío de mensajes entre emisor y receptor con un grado más alto de seguridad.

Permitir el desarrollo de nuevos servicios según el requerimiento del cliente, tal como una interfaz de estadística dinámica con datos obtenidos en línea, como apuestas, cantidad de jugadores, jugadas.

11. GLOSARIO

Wi-Fi: Wireless Fidelity es una marca de la *Wi-Fi Alliance* (anteriormente la *WECA: Wireless Ethernet Compatibility Alliance*), la organización comercial que adopta, prueba y certifica que los equipos cumplen los estándares 802.11. **Wi-Fi** no tiene ningún significado ni es acrónimo de nada. Es sólo una marca, un sello que sirve para certificar que un producto cumple con los estándares 802.11. [WIFI2008]

WiMAX: Las redes metropolitanas inalámbricas (por sus siglas en inglés WMAN) cubren una distancia mucho mayor que las WLAN, conectando edificios entre sí dentro de una amplia área geográfica. La emergente tecnología WiMAX (802.16d hoy día y 802.16e en un futuro próximo) permitirán mayor movilidad y reducirán la dependencia de las conexiones con cable. [INTEL2004]

3G: Redes amplias inalámbricas (por sus siglas en inglés WWAN) son las redes inalámbricas de mayor alcance, así como las más utilizadas hoy día en la infraestructura de telefonía móvil, aunque también disponen de la capacidad de transmitir datos. Los servicios de próxima generación de telefonía móvil basados en las diversas tecnologías 3G mejorarán significativamente las comunicaciones WWAN. [INTEL2004]

ARCN: (Attached Resource Computer Network) o Arcnet es una tecnología LAN muy utilizada que se basa en un esquema token – bus y es la más barata de instalar, permitiendo usar cables más largos sin pérdida de ancho de banda. [CORD07]

Ethernet: es la tecnología LAN más instalada y normalmente utiliza cable coaxial o tipos especiales de pares de cables trenzados para conseguir mayores velocidades de transmisión (el 10Base-T alcanza velocidades de hasta 10 Mbps, mientras que el Fast Ethernet o 100Base-T alcanza velocidades de hasta 100 Mbps). [CORD07]

FDDI: (Fiber Distributed Data Interface) es un estándar para la transmisión de datos a través de líneas de fibra óptica que pueden extenderse hasta una distancia de 124 millas. Basado en el protocolo Token Ring, el FDDI puede cubrir grandes áreas geográficas y tiene capacidad para miles de usuarios. [CORD07]

Routers: Es un ordenador de propósito especial (o un paquete de software) que maneja la conexión entre dos o más redes y examina las direcciones destinatarias de los paquetes que pasan a través de ellas, decidiendo por qué ruta enviarlos. [CORD07]

Servidores: Es un ordenador que comparte sus recursos, como impresoras y archivos, con otros ordenadores pertenecientes a una LAN. [CORD07]

Conmutador: Es un dispositivo de red que selecciona un camino o un circuito para enviar una unidad de información a su próximo destino. [CORD07]

Token Ring: Es un sistema utilizado cuando varios ordenadores están conectados a una red configurada en forma de anillo o de estrella, para evitar la colisión de los datos de dos ordenadores si estos envían sus mensajes a la red al mismo tiempo. Este es el segundo protocolo más utilizado para LANs. [CORD07]

FDMA: (Frequency Division Multiple Access) divide un espectro disponible en franjas no solapadas en la dimensión o dominio de la *frecuencia*. El FDMA es el modo más familiar de dividir un espectro y tradicionalmente ha sido utilizado por los sistemas analógicos. [CORD07]

TDMA: (Time Division Multiple Access) divide un espectro disponible en franjas no solapadas en la dimensión o dominio del *tiempo*. Los sistemas digitales son típicamente una combinación de FDMA y TDMA, donde la capacidad disponible se divide tanto en dimensiones de frecuencia como de tiempo, asignando a los usuarios canales de distintas frecuencias que utilizan en distintas franjas de tiempo. [CORD07]

GSM: (Global System for Mobile Communications) es un tipo de red digital inalámbrica TDMA con características de cifrado y usada ampliamente por toda Europa a 900 MHz. CDAM (Code Division Multiple Access) está basado en el concepto de espectro ensanchado, lo que significa que múltiples conversaciones comparten simultáneamente un espectro disponible y se distinguen entre sí mediante codificación en vez de usar canales de frecuencia o de tiempo [CORD07].

HomeRF: fue diseñado por el Grupo de Trabajo HomeRF (HRFWG), un consorcio que desarrolló una única especificación, el Protocolo de Acceso Inalámbrico Compartido (SWAP), para la gama de dispositivos de consumo que interactúan entre sí. Entre los más de 90 miembros que forman parte de la HRFWG se encuentran compañías líderes en el mercado de la industria de ordenadores, electrónica de consumo, periféricos, comunicaciones, software y semiconductores. Algunas de las empresas que forman parte de HRFWG son Toshiba, Compaq, Ericsson, Motorola, Hewlett- Packard, IBM, Intel, Microsoft, Philips, Proxim and Symbionics, Harris Semiconductor, National Semiconductor, Rockwell y Samsung. Probablemente, HomeRF representa el mayor reto para Bluetooth porque este consorcio está constituido por empresas muy importantes y representativas de las tecnologías de redes e inalámbricas. [CORD07]

HomeRF planea utilizar las mismas frecuencias que Bluetooth, así como velocidades de transferencia de datos de 1 Mbit/s. Se centra en la banda de 2.4

GHz y utiliza la tecnología de salto de frecuencia denominada SWAP (Protocolo de Acceso Inalámbrico Compartido). [CORD07]

SWAP: es una especificación industrial abierta que permite a los ordenadores, periféricos, teléfonos inalámbricos y otros dispositivos electrónicos compartir y comunicar voz y datos dentro y alrededor de la casa, sin las complicaciones y gastos de tener que tender nuevos cables. SWAP pretende "sentar las bases de una amplia gama de dispositivos de consumo que interactúan entre sí, estableciendo una especificación industrial abierta para comunicaciones inalámbricas digitales entre PCs y dispositivos electrónicos de consumo en cualquier lugar del hogar y sus alrededores" [CORD07].

HiperLAN: HiperLAN es un nuevo estándar europeo ratificado por primera vez en 1995 por el ETSI (Instituto Europeo de Telecomunicaciones). Originalmente HiperLAN usaba la misma banda de 2,4 GHz que usa Bluetooth para proporcionar servicios de red Ad-hoc tipo entre pares o la tradicional cliente - servidor a velocidades de transferencia de datos de 1 ó 2 Mbit/seg. [CORD07]

JSP: (Página de Servidor Java) Se refiere a un tipo especial de páginas HTML, en las cuales se insertan pequeños programas que corren sobre Internet (comúnmente denominados scripts), se procesan en línea para finalmente desplegar un resultado final al usuario en forma de HTML. Por lo general dichos programas hacen consultas a bases de datos y dependiendo del resultado que se despliegue será la información que se muestre a cada usuario de manera individual. Los archivos de este tipo llevan la extensión ".jsp". [ACM2007]

I-Mode: es el equivalente a WAP en Japón, tecnología utilizada por la operadora japonesa para prestar este servicio es más sencilla que la utilizada por las operadoras europeas para ofrecer servicios WAP.

I-Mode utiliza para su funcionamiento una variante del HTML (Hyper Text Markup Language, el lenguaje de marcos utilizado para crear páginas web), y una estructura basada en los siguientes elementos:

- **Terminales móviles capaces de soportar esta tecnología:** los terminales poseen capacidades superiores a las ofrecidas en otros mercados por los móviles destinados a servicios WAP, como pantallas de color, mayor capacidad de memoria, cámaras incorporadas o posibilidad de agregarlas como accesorio, etc.

- **Red de conexión:** comprime la información y la envía al terminal móvil, de modo que se puede enviar mayor número de datos. Se utiliza un sistema de empaquetado de la información que permite sacar el máximo partido a la conexión.

- **Servidor:** realizan varias funciones: controlan el acceso a la información por parte del usuario para poder facturar, realizan una segmentación de la información que aportan los proveedores de contenidos de modo que el usuario tenga mayor facilidad a la hora de acceder a la información que desee, y controlan a qué tipo de servicios está suscrito el usuario de modo que se le pueda enviar información de su interés.

- **Proveedores de contenidos:** son los encargados de poner a disposición del usuario aquella información que pueda serle de utilidad.

12. BIBLIOGRAFÍA

12.1 Referencias Bibliográficas

[J2ME2003] Agustín Froufe Quintanas - Patricia Jorge Cárdenas(2003), "J2ME Java 2 Micro Edition"

[Stall2001] Stallings William, (2001), "Sistemas operativos", Ed. Prentice hall

[TecMov2005] Zamora Abigail, (2005) "Tecnología móvil", Manual de capacitación de Nokia.

[SisOpe1993] Deitel Harvey M., 1993, "Sistemas operativos", Ed. Addison.

[SisOpeCon1993] Milenkovic Milan, (1993), "Sistemas Operativos Conceptos y diseño", Ed. Mc Grw Hill España.

[REAL2007] Real Academia Española © Todos los derechos reservados

[ETES2007] ETESA (Empresa Territorial para la Salud)

[SCHNEI96] Schneier, Bruce. Applied Cryptography: Protocols, Algorithms, and Source Code in C, 2nd. Edition. John Wiley & Sons, 1996.

[JAVAELE2003] Java a Tope: J2ME (Java 2 Micro edition) Edición Electrónica Sergio Gálvez Rojas – Lucas Ortega Díaz

[WAPANA] Dornan, Andy. *WAP.* Ediciones Anaya Multimedia. Glosario: Págs. 319-336.

[GUIWAP2002] Hougland, Damon. *Guía esencial WAP.* Prentice Hall, cop. 2002. Capitulo 6."Seguridad WAP". Págs.143-167

[COMOV] Huidobro Moya,Jose Manuel. *Comunicaciones Móviles.* Ed. Thomson-Paraninfo. Glosario: Págs. 419-431

12.2 Referencias de Internet

[MIDPCON]
URL: http://www.it.uc3m.es/celeste/docencia/j2me/docs/api/midp/

[MIDPJ2ME]
URL: http://www.it.uc3m.es/celeste/docencia/j2me/tutoriales/midp2_0/PracticaPKI/

[AESCSRC]
URL: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[DESCSRC]
URL: http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf

[MOZALG]
URL:http://www.mozilla.org/projects/security/pki/nss/nss-3.9/nss-3.9-algorithms.html

[JAVMIDP]
URL: http://Java.sun.com/products/midp

[JAVWIRE]
URL: http://wireless.Java.sun.com

[ENGA2009]
URL:http://es.engadget.com/2009/03/19/la-gran-comparacion-de-los-sistemas-operativos-moviles/

[VERISSL]
URL: http://www.verisign.es/ssl/index.html?sl=t13650207760000018&set=b033958

[VERISEC]
URL:http://www.verisign.es/ssl/ssl-information-center/how-ssl-security-works/index.html

[AVVILLAS]
URL:https://www.avvillas.com.co/portal/page?_pageid=33,124324859&_dad=portal&_schema=PORTAL

[SONYDEV]
URL:http://developer.sonyericsson.com/device/loadDevice.do?id=6b41eac4-0d8b-43b1-acb1-5129c2b3f8be

[THAWTE]
URL:https://www.thawte.com/ucgi/gothawte.cgi?a=t05470005477071007&set=b058152&gclid=CNHX4J6DuJoCFQa-sgodTSwtbw

[WAPFOR]
URL:http://www.wapforum.com

[UDECCHI]
URL:http://www2.udec.cl/~racuna/wap/introduccion.htm

[REAL2007]
URL: http://buscon.rae.es/draeI/SrvltConsulta?TIPO_BUS=3&LEMA=azar

[CORA 2008]
URL: http://www.corazones.org/diccionario/juegos_azar.htm

[MERC2006]
URL: http://www.mechantwarehouse.com/products/nuri-8000.shtml

[FDMA2008]
URL:http://www.overclockers.cl/foros/index.php?showtopic=164740

[PAGW2007]
URL: http://www.paginas-wap.com7index.asp

[JUEG2008]
URL: http://www.apuesta-en-intenet.com

[APUE2007]
URL: http://www.apuestaselgordodelasuerte.com/GaneFinal/index.php

[TECH 2003]
URL: http://www.tech-faq.com/lang/es/gsm.shtml

[UPVS2006]
URL: http://www.upv.es/satelite/trabajos/Grupo3_99.00/GlobalStar8.htm

[UPVC2006]
URL: http://www.upv.es/satelite/trabajos/pracGrupo3/ponce.htm

[TDMA2008]
URL: http://www.alegsa.com.ar/Dic/tdma.php

[TPVM2008]
URL: http://www.consoltic.com/modules/faq/RSTU/faq_0003.html

TIGO2008]
URL: http://www.tigo.com.co/paquetes_tigo_web.php

[WIFI2008]
URL: http://www.wi-fi.org/

[FDMA2008]
URL: http://www.overclockers.cl/foros/index.php?showtopic=164740

[FECE2008]
URL: http://www.feceazar.org/modulo.php?id=1

[INTEL2004]
URL: http://www.intel.com/cd/network/communications/emea/spa/179913.htm

[ELCO2008]
URL: http://www.elcolombiano.com

[UVES 1996]
URL: http://www.uv.es/sto/cursos/seguridad.Java/html/sJava-3.html

[GPHP2004]
URL: http://geneura.ugr.es/~maribel/php/Javascript.js

[SEGWAP06]
URL: seguridad WLM\Protocolo WAP1.htm

[CORD2007]
URL: http://www.cordobawireless.net/portal/descargas/Wireless_intro.pdf

[ACM2007]
URL: http://www.acm.org/crossroads/espanol/xrds8-2/servletsProgramming.html

[MICROTEC]
URL: http://www.microsoft.com/spain/empresas/tecnologia/hotspot.mspx

[JAVSERVL]
URL:http://translate.google.com.co/translate?hl=es&sl=en&u=http://Java.sun.com/products/servlet/&sa=X&oi=translate&resnum=5&ct=result&prev=/search%3Fq%3Dque%2Bes%2Bun%2Bservlet%2BJava%26hl%3Des%26pwst%3D1

[TECSERVL]
URL:http://www.tecnun.es/asignaturas/Informat1/ayudainf/aprendainf/JavaServlets/servlets.pdf

[UVTUECLIP]
URL: http://www.uv.es/~jgutierr/MySQL_Java/TutorialEclipse.pdf

***[*WMLCL]**
Pagoaga Fernández, Juan. *Seguridad en WAP II*. Enero, 2000.
URL: http://www.wmlclub.com/articulos/seguridad.htm

[SOSYMB]
Sistema operativo Symbian, (2007)
URL: http://www.symbian.com

[ELTIEMP]
URL:http://www.eltiempo.com/colombia/bogota/2008-09-22/IMAGEN/IMAGEN-4543755-1.jpg

[NetMov]
URL: http://leo.ugr.es/J2ME/APPS/GuionTetris/guionNetbeansMoviles.pdf

13. ANEXOS

13.1 Anexo 1

DES (Data Encription Standard) algoritmo de cifrado DES [DESCSRC]

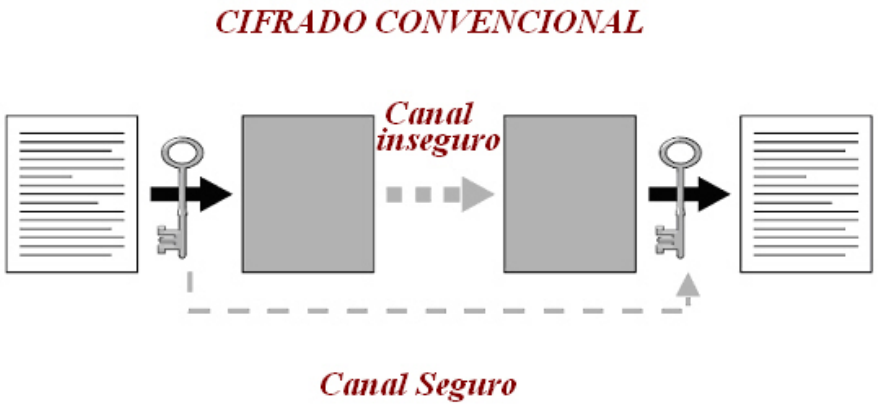
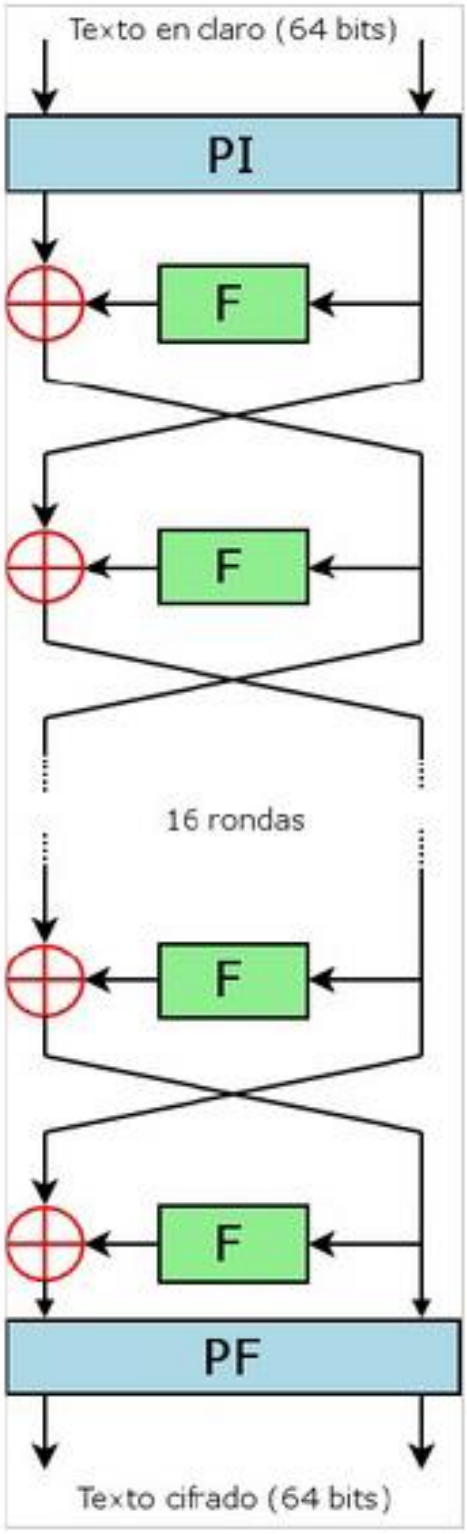


Figura. 51 Esquema de cifrado convencional del DES

El estándar de cifrado de datos es considerado uno de los algoritmos prototipos de cifrado por bloques, el tamaño de los bloques generados por DES son de 64 bits, utiliza también una clave criptográfica para cifrar, de modo que el descifrado solo puede hacerse por aquellos que conozcan la clave secreta de cifrado.

La clave mide 64 bits, aun que solo se utilizan 56 de ellos, los ocho bits restantes se utilizan para comprobar paridad, y después estos son descartados.

En el diagrama DES hay 16 fases idénticas llamadas rondas, también hay una permutación inicial y final llamadas PI y PF respectivamente. La Permutación Inicial PI deshace la acción de la Permutación Final PF y viceversa. El bloque de 64 bits es dividido en dos de 32 y son procesada de manera alternativa, a esto se le conoce como esquema Feistel. El Feistel asegura que el cifrado y el descifrado sean procesos muy similares la diferencia es que las sub claves se aplican en orden inverso cuando se descifra, y el resto del algoritmo es idéntico.



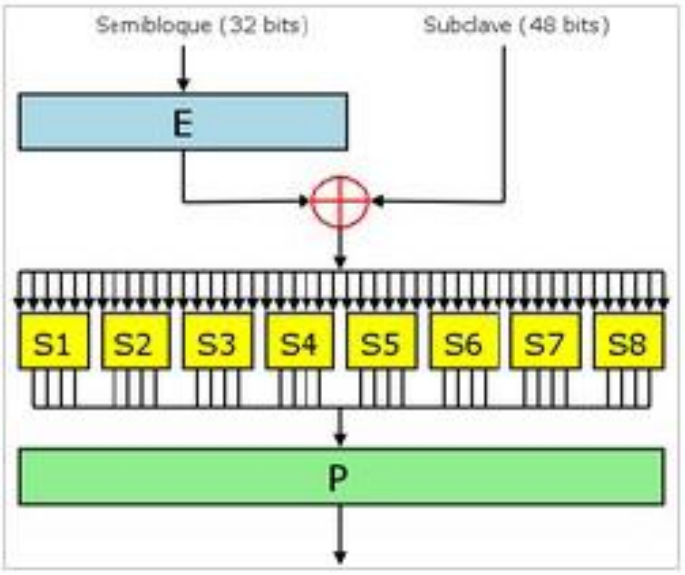


Figura. 52 Diagrama de flujo DES para el mensaje

En el diagrama de bloques de la izquierda se representan las 16 permutaciones que se deben realiza en el algoritmo. El diagrama de la derecha representa el proceso para cada ronda del algoritmo.

- Cifrado: $C = E_{k_1}(D_{k_2}(E_{k_1}(P)))$
- Descifrado: $P = D_{k_1}(E_{k_2}(D_{k_1}(C)))$

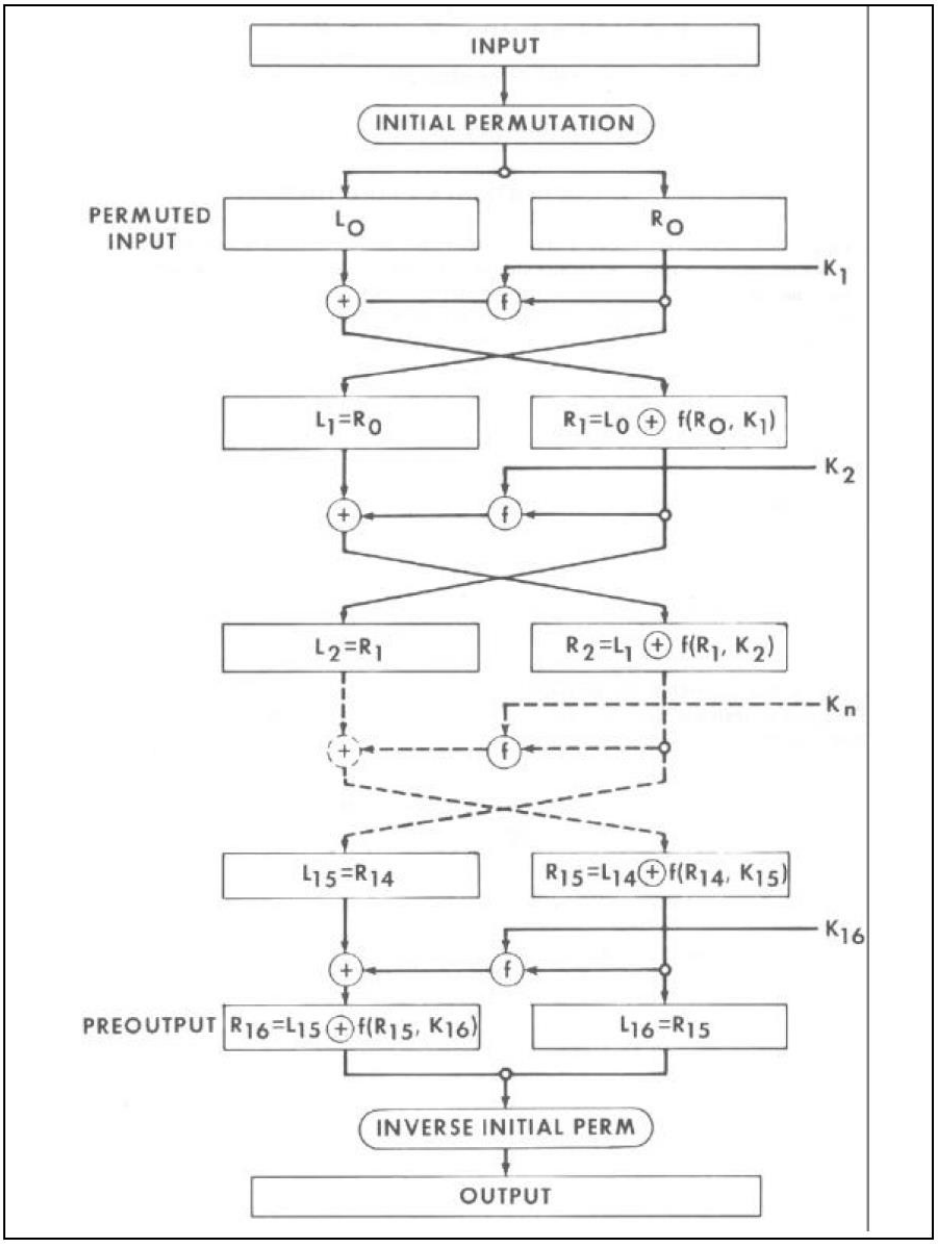


Figura. 53 Diagrama de especifico DES

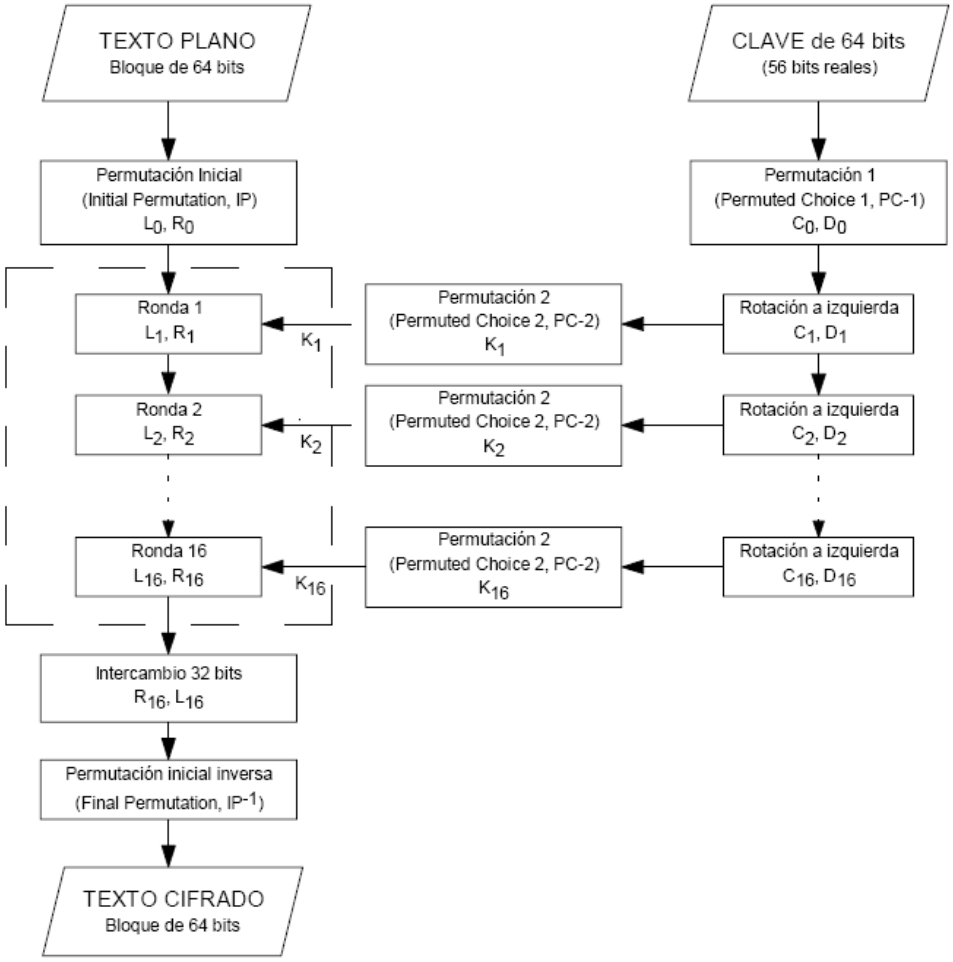


Figura. 54 Esquema General del Algoritmo DES

13.1.1 Desarrollo de las permutaciones

Se realiza el proceso para desarrollar el bloque 64 bits de la permutación inicial seleccionando la segunda, cuarta, sexta y octava columna.

Tabla antes la Permutación							
1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

Tabla antes la Permutación							
1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

Tabla. 16 Tablas antes de la permutación

Se separan el mensaje original de 64 bits en bloques de 8 bits, previo a la permutación, la tabla *P1* muestra el resultado de la permutación,

Tabla después Permutación (P1)							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Tabla después Permutación (P1)							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Tabla. 17 Tablas después de la permutación P1 (Permutación Inicial)

En la parte superior se encuentran los números pares que corresponden a las columnas 2, 4, 6 y 8, y en la parte inferior se encuentran los números impares que corresponden a las columnas 1, 3,5 y 7.

Otra forma de representar las tablas del DES

Permutación Inicial (P1)															
58	50	42	34	26	18	10	2	60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6	64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1	59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5	63	55	47	39	31	23	15	7

Tabla. 18 Tabla de permutación inicial P1

Estas representaciones muestran la ubicación de los bits del mensaje, es decir que en la permutación P1 el bit 58 del mensaje queda ubicado al inicio de la tabla en la posición 1.

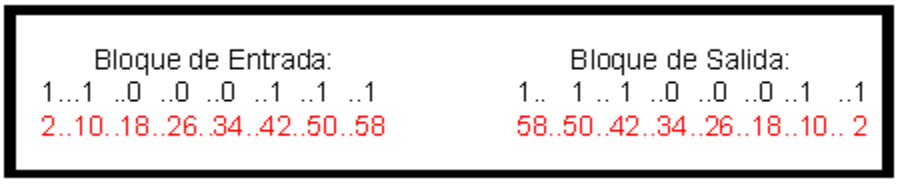


Figura. 55 Ejemplo de permutación P1

Al realizar la permutación de los 64 bits, se hace una nueva división en dos sub-bloques izquierdo y derecho L_i (Left) y R_i (Rigth) de 32 bits cada uno, correspondientemente. Los primeros 32 conforman el bloque Li, los restantes conforman el bloque Ri.

.

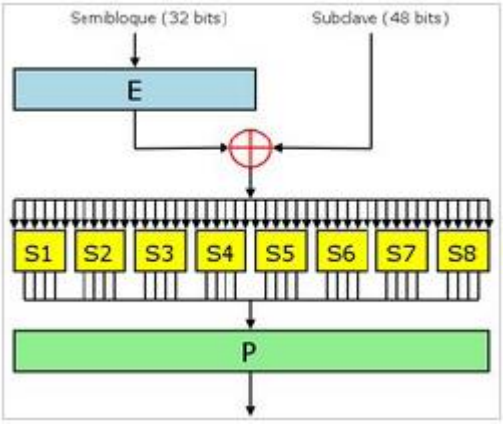


Figura. 56 Proceso para realizar cifrado DES

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus f\,(R_{i-1}, K_i)$$

Li es el sub bloque de 32 bit al cual se le aplica la permutación E con el fin de obtener un bloque de 48 bits. Por otro lado se encuentra el bloque con el cual se genera la llave que debe ser de 48 bits con el fin de realizar la or exclusiva.

Siguiendo con el desarrollo se propone un ejemplo para analizar el procedimiento necesario para la implementación de algoritmo DES.

13.1.2 Desarrollo de Ejemplo de Cifrado DES

Se toma los ASCII correspondientes a cada letra, se forma un bloque de 8 x 8 con 64 posiciones, a las cuales se hará la permutación inicial.

Mensaje a Cifrar = Denytamo

Decimal	Carácter	Binario
97	a	01100001
68	D	01000100
101	e	01100101
109	m	01101101
110	n	01101110
111	o	01101111
116	t	01110100
121	y	01111001

D	01000100
e	01100101
n	01101110
y	01111001
t	01110100
a	01100001
m	01101101
o	01101111

Tabla. 19 Tablas de mensaje a cifrar

Tabla antes la Permutación							
1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

Tabla Permutación (P1)							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Tabla. 20 Tablas del antes y el después de permutación P1

Se aplica la tabla de la permutación obtenida a la tabla de datos de entrada, la tabla antes de la permutación.

Tabla antes la Permutación							
0	1	0	0	0	1	0	0
0	1	1	0	0	1	0	1
0	1	1	0	1	1	1	0
0	1	1	1	1	0	0	1
0	1	1	1	0	1	0	0
0	1	1	0	0	0	0	1
0	1	1	0	1	1	0	1
0	1	1	0	1	1	1	1

Permutación Inicial (P1)							
1	1	1	1	1	1	1	1
0	0	0	1	1	0	0	0
1	1	0	1	0	1	1	1
1	1	1	0	1	0	1	0
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	0
1	1	0	0	1	1	0	0
1	0	0	0	0	1	0	0

Tabla. 21 Tablas del antes y el después de permutación P1 con el bloque datos

Los bits resaltados en la tabla de permutación inicial P1, conforman el sub-bloque Lo, los bits restantes conforman el bloque Ro.

Sub-bloques iniciales

- L_0 = 11111111 00011000 11010111 11101010
- R_0 = 00000000 11111110 11001100 10000100

Para la salida Ro se utiliza la permutación E, con el fin de obtener 48 bits y asi poder hacer la OR exclusiva con la clave ki.

- R_0 = 0000 0000 1111 1110 1100 1100 1000 0100

Se rotan las columnas de los extremos una vez hacia arriba y se colocan en el extremo contrario, dejando en la mitad la tabla de los 32 bits, con el fin de tener la tabla resultante de 48 bits (permutación E).

Bits duplicados			
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32

→

32 Bits			
0	0	0	0
0	0	0	0
1	1	1	1
1	1	1	0
1	1	0	0
1	1	0	0
1	0	0	0
0	1	0	0

Permutación E					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	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	1

→

Permutación E					
0	0	0	0	0	0
0	0	0	0	0	1
0	1	1	1	1	1
1	1	1	1	0	1
0	1	1	0	0	1
0	1	1	0	0	1
0	1	0	0	0	0
0	0	1	0	0	0

Tabla. 22 Tablas de permutación E y aplicación de la permutación E al ejemplo

Al hacer la permutación E se obtienen 48 bit, en una matriz de 6 x 8, los bits resultantes son los siguientes:

Resultado de la permutación E

$E(R_0)$ = 000000 000001 011111 111101 011001 011001 010000 001000

13.1.3 Generación de la sub clave Ki

La clave K_i tiene un valor inicial de 64 bits, su longitud es fija, esta se debe llevar a un bloque de 48 para hacer la or con el bloque de texto de 48 bits.

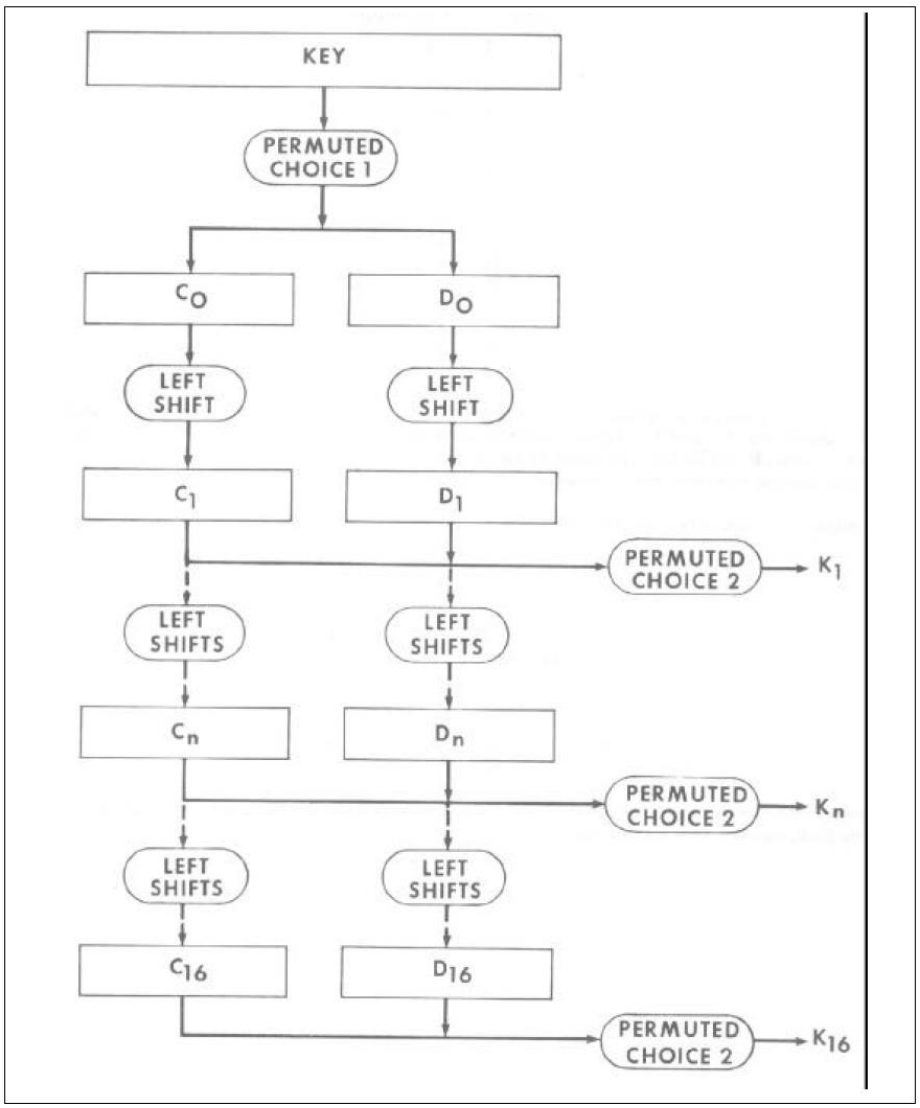


Figura. 57 Diagrama de bloques para la generación de la llave

Para conformar el bloque de 56 bits de la llave se le aplica la permutación pc1, consiste en eliminar los bits de paridad del bloque y reorganizar la matriz de la siguiente manera:

Tabla de 64 bits Inicial							
1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

Tabla de 56 bits						
1	2	3	4	5	6	7
9	10	11	12	13	14	15
17	18	19	20	21	22	23
25	26	27	28	29	30	31
33	34	35	36	37	38	39
41	42	43	44	45	46	47
49	50	51	52	53	54	55
57	58	59	60	61	62	63

Tabla. 23 Tablas para el bloque de 56 bits que conforma la llave

Esta se utiliza para la generación de la permutación inicial en de la sub claves Ki, cada ronda. Esto conforman 2 bloque llamados Ci y Di cada uno con un tamaño de 28 bits.

Permutación PC1						
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

Tabla. 24 Matriz resultante de permutación PC1

La clave secreta se encuentra definida por las siguientes ecuaciones:

$$C_i = LS\ (C_{i-1}) \quad D_i = LS(D_{i-1})$$

$$K_i = PC2\ (C_i\ ,\ D_i)$$

LS indica el desplazamiento que se debe hacer a la izquierda de manera circular como se indican en la siguiente tabla:

Vuelta	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
No.Bits desplazados. Izda.	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Se toma una palabra de ocho letras para representar la clave, esta constituirá el bloque de 64 bits.

- Clave K: Santiago

Decimal	Carácter	Binario
97	a	01100001
103	g	01100111
105	i	01101001
110	n	01101110
111	o	01101111
83	S	01010011
116	t	01110100

S	01010011
a	01100001
n	01101110
t	01110100
i	01101001
a	01100001
g	01100111
o	01101111

Tabla. 25 Matriz para constituir el bloque de 64 bits

Se aplica la permutación PC1 al bloque de entrada conformado por la clave Santiago.

Tabla de 64 bits de K_i Inicial							
0	1	0	1	0	0	1	1
0	1	1	0	0	0	0	1
0	1	1	0	1	1	1	0
0	1	1	1	0	1	0	0
0	1	1	0	1	0	0	1
0	1	1	0	0	0	0	1
0	1	1	0	0	1	1	1
0	1	1	0	1	1	1	1

Tabla de 56 bits eliminando paridad						
0	1	0	1	0	0	1
0	1	1	0	0	0	0
0	1	1	0	1	1	1
0	1	1	1	0	1	0
0	1	1	0	1	0	0
0	1	1	0	0	0	0
0	1	1	0	0	1	1
0	1	1	0	1	1	1

Permutación PC1						
0	0	0	0	0	0	0
0	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	0	0	0	0	0
1	1	0	0	0	1	0
1	1	1	0	0	1	1
0	0	1	0	0	1	0
1	0	0	1	0	0	1

Tabla. 26 Clave Santiago después de permutación PC1

En la tabla de la permutación PC1 se observa que no están el bit de paridad de cada byte que conforma la clave de 64 bits. Después de la permutación se obtienen los sub bloques C0 y D0 al dividir la matriz en dos partes, los 16 primeros bits conforman el primer bloque y los restantes 16 conforman el segundo respectivamente.

- Sub-bloque C_0 = 0000000 0111111 1111111 1100000
- Sub-bloque D_0 = 1100010 1110011 0010010 1001001

Sub-bloques iniciales después de la primera rotación LS

Sub-bloque C_1 = 0000000 1111111 1111111 1000000

Sub-bloque D_1 = 1000101 1100110 0100101 0010011

Seguido de esto se realiza la permutación PC2, esta es conocida como permutación de compresión, dado que en esta se comprimirán 56 bits a 48 bits y así poder obtener la clave Ki. Para el orden de concatenar C_i y D_i, es utilizando primero los 28 bits de C_i y posteriormente los 28 bits de D_i, la tabla PC2 es una tabla de 8 x 6, dando como resultado 8 bloques de 6 bits.

Tabla (C_i, D_i)						
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	32	33	34	35
36	37	38	39	40	41	42
43	44	45	46	47	48	49
50	51	52	53	54	55	56

Tabla de 8 x 6 bits					
1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36
37	38	39	40	41	42
43	44	45	46	47	48

Tabla. 27 Eliminación de bits de paridad de la tabla 56 bita para obtener 48bits

Se eliminan los bits de paridad de la matriz de 56 bits dando como resultado una matriz de 48 bits a la cual se le aplicara la permutación PC2.

Tabla de 8 x 6 bits					
1	2	3	4	5	6
7	8	10	11	12	13
14	15	16	17	19	20
21	23	24	26	27	28
29	30	31	32	33	34
36	37	39	40	41	42
44	45	46	47	48	49
50	51	52	53	55	56

Permutación PC2					
14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
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. 28 Tablas de permutación PC2

En la Permutación PC2 se obtiene la clave K_i.al concatenar Ci y Di siendo i igual a uno, al unirlos obtenemos una cadena de 48 bits.

Concatenando C_i, D_i; i=1

0000000 1111111 1111111 1000000 1000101 1100110 0100101 0010011

Tabla (C_i, D_i) 56 bits						
0	0	0	0	0	0	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	0	0	0	0	0	0
1	0	0	0	1	0	1
1	1	0	0	1	1	0
0	1	0	0	1	0	1
0	0	1	0	0	1	1

Permutación PC2 48 bits					
1	1	1	0	0	0
0	0	1	0	1	1
0	1	1	0	0	1
1	0	0	1	1	0
1	1	0	1	1	1
0	1	0	0	1	0
1	1	0	1	0	0
0	0	0	1	1	0

Tabla. 29 Permutación PC2 aplicando la trama de bits del mensaje

La primera tabla corresponde a los datos obtenidos de la primera permutación PC1, la segunda tabla es el resultante de la permutación PC2 siendo así la primera llave K1.

$$K_1 = PC2\ (C_1, D_1) =$$

K1 = 111000 001011 011001 100110 110111 010010 110100 000110

Las operaciones *LS* (Rotaciones a la Izquierda) y PC2, se repiten 15 veces para así obtener las 15 sub claves de cifrado restantes.

Al tener la clave K_i y la expansión de (R_0), el siguiente paso es la función $f(R_{i-1}, K_i)$, la cual consta de tres procesos (Suma OR exclusiva, ocho funciones no lineales, Permutación P).

13.1.4 Suma OR exclusiva

Con la clave K_i y $E(R_{i-1})$, se procede a realizar la suma OR exclusiva, al realizar la operación se tiene un 50% de certeza que el siguiente bit sea 1, con lo cual aumenta la dificultad de poder descifrar el mensaje. La operación OR exclusiva se ejemplifica en la siguiente tabla:

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Tabla. 30 Tablas de verdad de la OR exclusiva

La clave K_i y R_{i-1} se encuentra formado por 8 bloques de 6 bits cada un, con lo cual es posible realizar la suma OR exclusiva.

Suma OR exclusiva

Retomando los bits de la expansión del sub-bloque R_0 y la sub clave K_1, se procede a realizar la suma OR exclusiva. El resultado obtenidode la suma OR exclusiva $E(R_0) \oplus K_1$ es el siguiente:

$$\begin{array}{rl} \mathbf{E(R_0) =} & \mathbf{000000\ \ 000001\ \ 011111\ \ 111101\ \ 011001\ \ 011001\ \ 010000\ \ 001000} \\ & \oplus \\ \mathbf{K_1 =} & \mathbf{111000\ \ 001011\ \ 011001\ \ 100110\ \ 110111\ \ 010010\ \ 110100\ \ 000110} \\ \hline & \mathbf{111000\ \ 001010\ \ 000110\ \ 011011\ \ 101110\ \ 001011\ \ 100100\ \ 001110} \end{array}$$

Operación de la Or exclusiva entre $E(R_0) \oplus K_1$

La cadena de bits obtenida de la suma OR exclusiva, se subdivide en 8 bloques de 6 bits, como se puede apreciar en el ejemplo anterior, siendo cada bloque ($b_6b_5b_4b_3b_2b_1$), el resultado de la operación es un numero con valor entre 0 a 15, representado con cuatro bits, concatenados forman una cadena de 32 bits, la cual será la entrada para la permutación P. La posición de los bits dentro de cada caja se encuentra definida por la fila b_6b_1 y la columna $b_5b_4b_3b_2$ de la caja.

Función no lineal (cajas)

$$\mathbf{E(R_0) \oplus K_1 =} \ \mathbf{111000\ \ 001010\ \ 000110\ \ 011011\ \ 101110\ \ 001011\ \ 100100\ \ 001110}$$

En la Caja S_1 se utiliza el primer bloque **111000**, en la Caja S_2 se utiliza el segundo bloque y así sucesivamente.

Ver tablas en el anexo1

13.1.5 Cajas-S para DES

La salida del bloque 1 es: 3 (0011), el procedimiento se repite en las siguientes 7 cajas obteniendo como resultado:

Salida después de pasar por cajas S

$$\begin{array}{rl} \mathbf{E(R_0) \oplus K_1 =} & \mathbf{0011\ \ 1011\ \ 1110\ \ 1010\ \ 1000\ \ 1100\ \ 1011\ \ 0001} \\ & \mathbf{3\ \ \ \ 11\ \ \ \ 14\ \ \ \ 10\ \ \ \ 8\ \ \ \ 12\ \ \ \ 11\ \ \ \ 1} \end{array}$$

13.1.6 Permutación P

El ultimo paso de la función *f*(R_{i-1} , K_i), es una permutación P, cuyo resultado se sumara con la salida del sub-bloque L_i, dando origen a la entrada del sub-bloque R_i

Tabla antes de la permutación P			
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32

Tabla después de la Permutación P			
16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

Tabla. 31 Equivalente a tablas de permutación P

Se realiza el ultimo paso de la *f*(R_{i-1} , K_i), recordando que dicho proceso se repite 15 veces.

Tabla			
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32

Tabla			
0	0	1	1
1	0	1	1
1	1	1	0
1	0	1	0
1	0	0	0
1	1	0	0
1	0	1	1
0	0	0	1

Tabla. 32 Tablas antes de la permutación P

Permutación P			
16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

Permutación P			
0	1	0	1
0	0	1	1
0	1	0	0
1	0	0	1
0	1	0	0
1	1	1	1
0	1	0	0
1	1	1	1

Tabla. 33 Tablas después de la permutación P

Finalmente se obtiene $f(R_{i-1}, K_i)$, que son los bits resultantes después de la permutación P:

$$f(R_{i-1}, K_i) = \mathbf{0101\ 0011\ 0100\ 1001\ 0100\ 1111\ 0100\ 1111}$$

13.1.7 Suma $L_i \oplus R_i$

El registro de bits obtenido por la función $f(R_{i-1}, K_i)$, es sumado con un OR exclusiva con el registro de bits de L_0, dando como resultado la entrada para el siguiente sub-bloque de R_i.

El registro de bits realizado en la permutación inicial P1, que origino al sub-bloque R_0, es la entrada para el siguiente sub-bloque L_i, el proceso se repite 15 veces, siendo el ultimo proceso la permutación $P1^{-1}$.

Suma $L_i \oplus f(R_{i-1}, K_i)$

$$\begin{array}{lr} \mathbf{f(R_{i-1}, K_i) =} & \mathbf{0101\ 0011\ 0100\ 1001\ 0100\ 1111\ 0100\ 1111} \\ & \oplus \\ \mathbf{L_0 =} & \mathbf{1111\ 1111\ 0001\ 1000\ 1101\ 0111\ 1110\ 1010} \\ \hline & \mathbf{1010\ 1100\ 0101\ 0001\ 1001\ 1000\ 1010\ 0101} \end{array}$$

Resultado de la suma OR exclusiva $L_0 \oplus f(R_0, K_1)$

Obteniendo los registros de 32 bits para el siguiente sub-bloque L_i y R_i como se muestra a continuación:

- L_1 = 0000 0000 1111 1110 1100 1100 1000 0100
- R_1 = 1010 1100 0101 0001 1001 1000 1010 0101

13.1.8 Verificación de las ecuaciones

Por propiedad se debe tener que $L_1 = R_0$

$$L_1 = R_0$$

00000000111111101100110010000100 = 00000000111111101100110010000100

Por propiedad se cumple que $R_1 = L_0 \oplus f(R_0 , K_1)$:

10101100010101011001100010100101 =
11111111000110001101011111101010⊕01010011010011010100111101001111

13.1.9 Proceso inverso para descifrado del algoritmo DES

Para llevar a cabo este proceso se debe tener en cuenta la generación de la tabla de descifrado del mensaje. El primer paso es realizar la permutación $P1^{-1}$. Se elimina los últimos bits de paridad para obtener un bloque de 56 bits con el cual se obtienes el bloque de la permutación inicial inversa (IP^{-1}).

Tabla antes la Permutación							
1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

Permutación Final ($P1^{-1}$)							
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	56	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

Tabla. 34 Tablas de la permutación inversa $P1^{-1}$

Si el mensaje es mayor de 64 bits, se utilizan los bloques necesarios para dividir el mensaje original, si un bloque formado por un mensaje, no tiene la longitud de 64 bits, se rellena utilizando 0.

Se toman los valores de L1y R1 suponiendo que estos son los valores de L16 y R16 para realizar el procedimiento.

L_{16} = **10101100 01010001 10011000 10100101**

R_{16} = **00000000 11111110 11001100 10000100**

Concatenando L_{16} , R_{16}

10101100 01010001 10011000 10100101

00000000 11111110 11001100 10000100

Se desarrolla la permutación inversa con los Bits R16 y L16, el resultado se ve en la siguiente tabla:

Tabla antes la Permutación							
1	0	1	0	1	1	0	0
0	1	0	1	0	0	0	1
1	0	0	1	1	0	0	0
1	0	1	0	0	1	0	1
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	0
1	1	0	0	1	1	0	0
1	0	0	0	0	1	0	0

→

Permutación Inversa $P1^{-1}$							
0	0	0	1	0	0	0	1
0	0	1	0	0	0	0	0
0	1	1	0	1	0	1	1
0	1	1	0	1	1	0	0
0	0	1	1	0	1	0	0
0	1	1	0	0	0	0	1
0	0	1	1	1	0	0	0
0	1	1	0	1	1	1	1

Tabla. 35 Tablas de permutación inversa $P1^{-1}$

El mensaje cifrado arroja los siguientes caracteres, ◄ seguido de un espacio una llave de abierta, una l minúscula, un numero cuatro (4), un carácter a, un número ocho (8) y una o minúscula.

◄ {l4a8o

Figura. 58 Mensaje cifrado DES

Decimal	Carácter	Binario
17	◄	**00010001**
32	(space)	**00100000**
123	{	**01111011**
108	L	**01101100**
52	4	**00110100**
97	A	**01100001**
56	8	**00111000**
111	O	**01101111**

Tabla. 36 Tablas de valores ASCII

Se llena la tabla con los valores recibidos L16 y R16 obtenidos al haber realizado la permutación Inversa $P1^{-1}$, esto con el fin de poder realizar la la permutación P1, con esto se verifica que los procesos de permutación inversa si se cumplen.

En las siguientes tablas se describe el proceso de la permutación Inicial.

Tabla antes la Permutación							
1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

→

Tabla antes la Permutación							
0	0	0	1	0	0	0	1
0	0	1	0	0	0	0	0
0	1	1	0	1	0	1	1
0	1	1	0	1	1	0	0
0	0	1	1	0	1	0	0
0	1	1	0	0	0	0	1
0	0	1	1	1	0	0	0
0	1	1	0	1	1	1	1

Tabla. 37 Tablas de antes de permutación P1 en la recepción

Tabla Permutación (P1)							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

→

Permutación Inicial (P1)							
1	0	1	0	1	1	0	0
0	1	0	1	0	0	0	1
1	0	0	1	1	0	0	0
1	0	1	0	0	1	0	1
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	0
1	1	0	0	1	1	0	0
1	0	0	0	0	1	0	0

Tabla. 38 Tablas de permutación P1 en la recepción

Permutación Inicial (P1)							
1	0	1	0	1	1	0	0
0	1	0	1	0	0	0	1
1	0	0	1	1	0	0	0
1	0	1	0	0	1	0	1
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	0
1	1	0	0	1	1	0	0
1	0	0	0	0	1	0	0

Tabla. 39 Tabla de división en bloques L16 y R16

L_{16} = 10101100 01010001 10011000 10100101

R_{16} = 00000000 11111110 11001100 10000100

Al tener la trama de 32 bits R16, se procede a aplicar la permutación E con el fin de adicionar los bits restantes para formar 48 bits.

R_{16} = 0000 0000 1111 1110 1100 1100 1000 0100

Ubicación de bits del Bloque R16 para aplicar proceso de permutación E

Bits duplicados			
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32

→

32 Bits			
0	0	0	0
0	0	0	0
1	1	1	1
1	1	1	0
1	1	0	0
1	1	0	0
1	0	0	0
0	1	0	0

Permutación E					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	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	1

→

Permutación E					
0	0	0	0	0	0
0	0	0	0	0	1
0	1	1	1	1	1
1	1	1	1	0	1
0	1	1	0	0	1
0	1	1	0	0	1
0	1	0	0	0	0
0	0	1	0	0	0

Tabla. 40 Tablas de permutación E en descifrado

Se obtiene los 32 bits después de realizar la permutación E

$E(R_{16})$ = 000000 000001 011111 111101 011001 011001 010000 001000

13.1.10 Suma OR exclusiva Descifrado

La suma OR exclusiva se realiza utilizando la última clave generada (K_{16}), en nuestro ejemplo utilizaremos la clave K_1, como la clave K_{16}.

$E(R_{16})$ = 000000 000001 011111 111101 011001 011001 010000 001000

K_{16} = 111000 001011 011001 100110 110111 010010 110100 000110

111000 001010 000110 011011 101110 001011 100100 001110

Resultado de la suma OR exclusiva $E(R_{16}) \oplus K_{16}$

$E(R_0) \oplus K_1$ = 111000 001010 000110 011011 101110 001011 100100 001110

Como resultados de las operaciones no lineales (cajas – S) tenemos los siguientes:

$E(R_0) \oplus K_1$ = 0011 1011 1110 1010 1000 1100 1011 0001
3 11 14 10 8 12 11 1

Para el descifrado es necesario aplicarla permutación P para obtener $f(R_{i-1}, K_i)$, ya que este proceso se realiza 15 veces.

Tabla			
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32

→

Tabla			
0	0	1	1
1	0	1	1
1	1	1	0
1	0	1	0
1	0	0	0
1	1	0	0
1	0	1	1
0	0	0	1

Permutación P			
16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

Permutación P			
0	1	0	1
0	0	1	1
0	1	0	0
1	0	0	1
0	1	0	0
1	1	1	1
0	1	0	0
1	1	1	1

Tabla. 41 Tablas de permutación P en descifrado

Resultado de la suma OR exclusiva $L_0 \oplus f(R_{i-1}, K_i)$

$$\begin{array}{lr} \boldsymbol{f(R_{i-1}, K_i) =} & \mathbf{0101\ 0011\ 0100\ 1001\ 0100\ 1111\ 0100\ 1111} \\ & \oplus \\ \boldsymbol{L_{16} =} & \mathbf{1010\ 1100\ 0101\ 0001\ 1001\ 1000\ 1010\ 0101} \\ \hline \boldsymbol{R_{15} =} & \mathbf{1111\ 1111\ 0001\ 1000\ 1101\ 0111\ 1110\ 1010} \end{array}$$

Por propiedad L_{15} es igual a R_{16} por lo tanto el resultado es:

- L_{15} = **0000 0000 1111 1110 1100 1100 1000 0100**
- R_{15} = **1111 1111 0001 1000 1101 0111 1110 1010**

13.1.11 Verificación de ecuaciones

$$L_{15} = R_{16}$$

00000000111111101100110010000100 = 00000000111111101100110010000100

$$R_1 = L_{16} \oplus f(R_{16}, K_{16})$$

11111111000110001101011111101010 =
10101100010100011001100010100101⊕01010011010010010100111101001111

Nuevamente se aplica la permutación inversa $P1^{-1}$

- L_1 = **1111 1111 0001 1000 1101 0111 1110 1010**
- R_1 = **0000 0000 1111 1110 1100 1100 1000 0100**

Concatenando L_1 y R_1, da como resultado obtenemos el bloque de 64 bits al cual se le aplicara la permutación inversa para obtener el mensaje enviado:

11111111 00011000 11010111 11101010
00000000 11111110 11001100 10000100

El resultado de la permutación se puede ver en las siguientes tablas, adicionalmente la comparación del mensaje obtenido con el original:

Tabla antes la Permutación							
1	1	1	1	1	1	1	1
0	0	0	1	1	0	0	0
1	1	0	1	0	1	1	1
1	1	1	0	1	0	1	0
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	0
1	1	0	0	1	1	0	0
1	0	0	0	0	1	0	0

→

Permutación Inversa $P1^{-1}$							
0	1	0	0	0	1	0	0
0	1	1	0	0	1	0	1
0	1	1	0	1	1	1	0
0	1	1	1	1	0	0	1
0	1	1	1	0	1	0	0
0	1	1	0	0	0	0	1
0	1	1	0	1	1	0	1
0	1	1	0	1	1	1	1

Tabla. 42 Tablas de permutación $P1^{-1}$ descifrado

D	01000100
e	01100101
n	01101110
y	01111001
T	01110100
A	01100001
M	01101101
O	01101111

→

Permutación Inversa $P1^{-1}$							
0	1	0	0	0	1	0	0
0	1	1	0	0	1	0	1
0	1	1	0	1	1	1	0
0	1	1	1	1	0	0	1
0	1	1	1	0	1	0	0
0	1	1	0	0	0	0	1
0	1	1	0	1	1	0	1
0	1	1	0	1	1	1	1

Tabla. 43 Tabla de comparación de mensaje

Tabla Inicial							
0	1	0	0	0	1	0	0
0	1	1	0	0	1	0	1
0	1	1	0	1	1	1	0
0	1	1	1	1	0	0	1
0	1	1	1	0	1	0	0
0	1	1	0	0	0	0	1
0	1	1	0	1	1	0	1
0	1	1	0	1	1	1	1

Tabla Final							
0	1	0	0	0	1	0	0
0	1	1	0	0	1	0	1
0	1	1	0	1	1	1	0
0	1	1	1	1	0	0	1
0	1	1	1	0	1	0	0
0	1	1	0	0	0	0	1
0	1	1	0	1	1	0	1
0	1	1	0	1	1	1	1

Tabla. 44 Tabla Inicial Vs Tabla Final

13.2 Anexo 2

Algoritmo de cifrado RSA(Rivest, Shamir y Adelman)

13.2.1 Definición de números primos

Se considera número primo a aquel número entero positivo distinto de 1 y 0, que sea divisible solo por 1 y por si mismo, que su resultado sea exacto en los dos casos.

- Divisores de 5= {1, 5} ***por lo tanto es primo***
- D(7)={1, 7} ***por lo tanto es primo***
- D(8)={1, 2, 4,8} ***no es primo, es divisible por 2 además de 1 ,4 y 8***

13.2.2 Números Gemelos Primos

Se considera número gemelos a los números primos cuya resta es 2.

5 ***es primo*** y 7 ***es primo***, y 7 - 5 = 2; 31 – 29 = 2;

13.2.3 Números Mersenne Primos

Se consideran números primos mersenne a aquellos que cumplan la expresión N=(2^n)-1 donde ***n*** puede ser cualquier numero y ***N*** es el primo de Mersenne, se considera hasta ahora solo se han descubiertos 45 primos de mersenne.

13.2.4 Números Primos relativos

Se le conocen como números primos entre si. Se le considera primos relativos a aquellos números que no son divisibles entre si, es decir que cuando el máximo común divisor es 1. El hecho de que los números sean pares e impares no influye.

- 38 ***Divido entre*** 13 ***son primos entre si***
- 38 ***Divido entre*** 14 ***son primos entre si***
- 17 ***Divido entre*** 13 ***son primos entre si***

13.2.5 Números Casi Primos

Se conoce como numero casi primos aquellos que números que solo tengan dos divisores primos.

21 ***Puede ser divido entre*** *3* ***y****7*

13.2.6 Sistema de cifrado RSA

El sistema RSA es un método desarrollado a partir de los números casi primos. El método consiste en seleccionar dos números primos, en este caso p y q; estos tienen que ser de gran tamaño.

Esto se hace con el fin de obtener el producto de los dos ***n = pq.*** De esta manera se obtiene la clave pública conocida como ***n*** con características de un casi primo, haciendo difícil encontrar los factores p y q.

Criterios para la elección de los factores p y q:

1- Se calcula el mcd (mínimo común divisor) de p - 1 y q – 1, para el ejemplo se toman números pequeños.

p = 59, q = 43

p – 1 = 58 q – 1 = 42 **MCD = 2**

58	2	42	2
29	29	21	3
1		7	7
		1	

2- La descomposición en factores primos de p - 1 y q - 1 debe tener algún factor primo grande p' y q'.

58	2	42	2
29	29	21	3
1		7	7
		1	

3- La descomposición en factores primos de p' - 1 y q' - 1 deben tener factores primos grandes.

p' = 29 q' = 7

p' – 1 = 28 q' – 1 = 6 **MCD = 2**

28	2	6	2
14	2	3	3
7	7	1	
1			

4- La descomposición en factores primos de p' + 1 y q' + 1 deben tener factores primos grandes.

p' = 29 q' = 7

p' + 1 = 30 q' + 1 = 8 **MCD = 2**

30	2	8	2
15	3	4	2
5	5	2	2
1		1	

Si p y q cumplen las cuatro condiciones se les pueden llamar ***números primos fuertes.***

Para la clave privada se tiene otro numero ***e*** que es primo y debe cumplir la condición ***mcd (e,(p-1).(q-1)) = 1*** que es equivalente a ***mcd(e ,Φ(n))=1,*** donde estos números tienen que ser primos relativos.

Es un sistema de clave pública que garantiza autenticidad de la información y del remitente.

Teorema para cifrar
$encrypt(m) = m^e \pmod n$

Teorema para descifrar
$decrypt(c) = c^d \pmod n$

$(m^e)^d = m \bmod n$

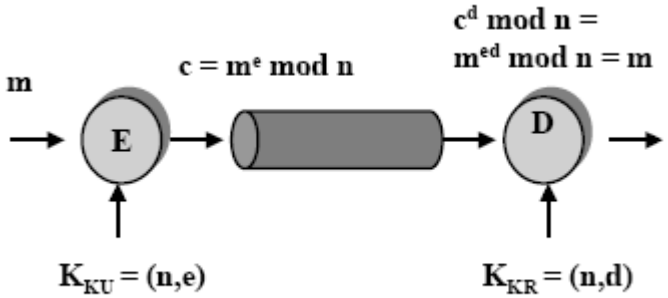


Figura. 59 Esquema de cifrado RSA

La notación de la clave publica es ***(n,e)*** de manera que n es el producto de dos números primos grandes ***n = pq.*** ***e*** es el exponente de cifrado, este varía entre $1<e<\Phi(n)$, siendo ***Φ(n)= (p-1)(q-1).*** La clave privada maneja la notación ***(n,d),*** es el inverso de ***e*** modulo ***Φ(n): 1<d<Φ(n)***.

El exponente ***e*** varía de acuerdo a lo seguro y rápido que se quiere que sea el sistema, entre mas pequeño es más rápido, pero puede no ser tan seguro. Los exponentes de cifrado que se suelen recomendar son dos; ***e = 3 y e =(2exp16) + 1= 65537*** esta corta expresión hace que el sistema sea rápido.

Por propiedad ***e*** y ***d*** corresponde a ***e*d = 1 mod Φ(n)*** dado que ***d*** es inverso multiplicativo de ***e*** así se puede calcular ***d*** a partir de la formula dada ***d = 1 + k Φ(n) / e <Φ(n),*** siendo ***k = Φ(n) mod e*** de modo que ***k*** tiene que ser menor que ***e*** y se debe cumplir la expresión ***k < e – 1/ Φ(n) < e.***

Dentro de las garantías para el cifrado se debe tener en cuenta el criptoanálisis de Winer, en este se debe garantizar que $d < 1/3\ n^{1/4}$ esto con el fin de que ***d*** no se pueda determinar en un tiempo calculado.

Para cifrar un bloque (P) se toman los caracteres y se convierten en sus equivalentes numéricos, para llevar a cabo el cifrado y descifrado del bloque se dispone de los teoremas:

$$Bloque\ Cifrado\ (m) = m^e \pmod n$$

Bloque Descifrado (c) = $c^d \pmod n$

Desarrollo de ejemplo para el cifrado RSA

13.2.7 Frase a cifrar

PUBLIC KEY CRIPTOGRAPHY

PU	BL	IC	KE	YC	RI	PT	OG	RA	PH	Y**X**
1520	0111	0802	1004	2402	1708	1519	1406	1700	1507	2423

Tabla. 45 Frase a cifrar con RSA

13.2.8 Equivalente Numérico

A	B	C	D	E	F	G	H	I	J	K	L	M
00	01	02	03	04	05	06	07	08	09	10	11	12
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Tabla. 46 Equivalente en bits de cada letra del alfabeto

Ya teniendo los valores numéricos separados en bloques de a cuatro, se procese a seleccionar los números primos ***p*** y ***q.*** Se toman los valores obtenidos del ejemplo de criterios para la elección de los factores p y q:

p = 59, q = 43

Se calcula ***p- 1*** y ***q – 1***

p – 1 = 58 q – 1 = 42 **MCD = 2**

Seguido de esto se procede al cálculo de la variable ***n,*** estas ese el producto de ***n = p*q.*** Para este proceso se utilizan número primos mucho más grandes.

p = 59, q = 43

n = p*q

n = 43*59

n = 2537

Se calcula ***Φ(n)***

Φ(n)= (p-1)(q-1)

p = 59, q = 43

Φ(n)= (p-1)(q-1)

Φ(n)= (59-1)(43-1)

Φ(n)= 2436

Se toma el exponente ***e*** con el con el siguiente criterio ***1<e<Φ(n),*** se toma ***e*** en menor tamaño para poder hacer los cálculos.

e = 13

1<e<Φ(n)

1<13<2436

Para saber si el exponente es valido se mira que se cumpla el criterio del máximo común divisor, que sean primos relativos

mcd (e ,Φ(n))=1 o mcd (e,(p-1).(q-1)) = 1.

mcd (13 ,2436)=1

Se arma la ecuación para el bloque de cifrado

Bloque Cifrado (*m*) = m^e(mod *n*)

Bloque Cifrado (*m*) = m^{13} (mod *2537*)

Siendo *m* cada uno de los bloques de 4*:*

PU	BL	IC	KE	YC	RI	PT	OG	RA	PH	Y**X**
1520	0111	0802	1004	2402	1708	1519	1406	1700	1507	2423

Tabla. 47 Frase a cifrar con RSA

13.2.9 Proceso para encontrar el primer equivalente numérico

PU
1520

Bloque Cifrado (*m*) = 1520^{13} (mod *2537*)

Se encuentra el múltiplo equivalente 1520^{13}

1520^1 =1520*1 = 1520 = 1520 (mod *2537*)

1520^2 =1520*1520 = 2310400 = 2310400(mod *2537*) = 1730(mod *2537*)

1520^3 =1520*1730 = 2629600 = 2629600(mod *2537*) = 1268(mod *2537*)

1520^4 =1520*1268 = 1927360 = 1927360(mod *2537*) = 1777(mod *2537*)

1520^5 =1520*1777 = 2701040 = 2701040(mod *2537*) = 1672(mod *2537*)

1520^6 =1520*1672 = 2541440 = 2541440(mod *2537*) = 1903(mod *2537*)

1520^7 =1520*1903 = 2892560 = 2892560(mod *2537*) = 380(mod *2537*)

1520^8 =1520*380 = 577600 = 577600(mod *2537*) = 1701(mod *2537*)

1520^9 =1520*1701 = 2585520 = 2585520(mod *2537*) = 317(mod *2537*)

1520^10 =1520*317 = 481840 = 481840 (mod *2537*) = 2347(mod *2537*)

1520^11 =1520*2347 = 3567440 = 3567440(mod *2537*) = 418(mod *2537*)

1520^12 =1520*418 = 635360 = 635360(mod *2537*) = 1110(mod *2537*)

1520^13 =1520*1110 = 1687200 = 1687200(mod *2537*) = 95(mod *2537*)

Bloque Cifrado (*m*) = 95 (mod *2537*) = *Bloque Cifrado* (*m*) = 1520^{13} (mod *2537*)

Bloque Cifrado (*m*) = 95

PU	BL	IC	KE	YC	RI	PT	OG	RA	PH	Y**X**
0095	1648	1410	1299	0811	2333	2132	370	1185	1457	1084

Tabla. 48 Tabla con múltiplo obtenido de PU

13.2.10 Recuperación del texto original

Bloque Descifrado (*c*) = c^d(mod *n*)

Se calcula ***d*** por medio de la siguiente expresión

e*d = 1 mod Φ(n)

d = 1 + k Φ(n) / e <Φ(n).----- k = Φ(n) mod e

k = 2436 mod 13 ----- d = 1 + (5) (2436) / 13

d = 937

Bloque Descifrado (*c*) = c^{937}(mod *2537*)

Se remplaza por el bloque cifrado que llega

Bloque Descifrado (*c*) = 95^{937}(mod *2537*)

Bloque Descifrado (*c*) = 1520(mod *2537*)

Bloque Descifrado (*c*) = 1520

Así se obtienen todos los equivalentes al descifrar.

PU	BL	IC	KE	YC	RI	PT	OG	RA	PH	Y**X**
0095	1648	1410	1299	0811	2333	2132	370	1185	1457	1084
1520	0111	0802	1004	2402	1708	1519	1406	1700	1507	2423

Tabla. 49 Tabla comparativa entre el mensaje original y los múltiplos obtenidos

b_6b_1	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	**1100**	1101	1110	1111	
00	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7	
01	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8	S_1
10	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0	
11	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13	

Caja S1

b_6b_1	0000	0001	0010	0011	0100	**0101**	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
00	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10	
01	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5	S_2
10	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15	
11	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9	

Caja S2

b_6b_1	0000	0001	0010	**0011**	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
00	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8	
01	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1	S_3
10	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7	
11	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12	

Caja S3

b_6b_1	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	**1101**	1110	1111	
00	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15	
01	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9	S_4
10	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4	
11	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14	

Caja S4

b_6b_1	0000	0001	0010	0011	0100	0101	0110	**0111**	1000	1001	1010	1011	1100	1101	1110	1111	
00	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9	
01	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6	S_5
10	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14	
11	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3	

Caja S5

b_6b_1	0000	0001	0010	0011	0100	**0101**	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
00	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11	
01	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8	S_6
10	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6	
11	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13	

Caja S6

b_6b_1	0000	0001	**0010**	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
00	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1	
01	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6	S_7
10	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2	
11	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12	

Caja S7

b_6b_1	0000	0001	0010	0011	0100	0101	0110	**0111**	1000	1001	1010	1011	1100	1101	1110	1111	
00	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7	
01	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2	S_8
10	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8	
11	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11	

Caja S8