CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS
AVANZADOS DEL INSTITUTO POLITÉCNICO
NACIONAL

**DEPARTAMENTO DE INGENIERÍA ELÉCTRICA**
**SECCIÓN DE COMPUTACIÓN**

# Seguridad en redes inalámbricas para sistemas multimedia de tiempo real

Tesis que presenta

**Luis de Jesús González Noriega**

Para obtener el grado de

**Maestro en Ciencias**

en la Especialidad de

**Ingeniería Eléctrica**

opción

**Computación**

Director de la tesis:

**Dr. Pedro Mejía Alvarez**

México, D.F. Julio 2005

# Abstract

The constantly growth of wireless local area networks (WLAN's) has generated that mobile communication equipment, such as personal digital assistants (PDA's), that before were simple devices of personal organization, now are considered as mobile computing equipment. Nowadays, PDA's make operations which are comparable to those made by personal computers (PC's). PDA's are able to execute applications in mobility environments where they can have access to different data types (i.e., multimedia data ,voice, video, images). However, they are small devices with limited battery power and memory restrictions. This brings some disadvantages, such as low levels of security and quality of service (QoS).

This thesis introduces the design and implementation of a middleware software system, whose objective is to increase the security levels in the data transfer of real-time multimedia applications on a WLAN. This system is capable of providing a flexible management of QoS and security levels.

In the introduction of security to the system, we developed cryptographic symmetrical and asymmetric algorithms. The introduction of this type of security algorithms provides flexibility to the system. Also, we developed and implemented a handshake protocol for the Wireless Transport Layer Security (WTLS) which was part of the layer of the Wireless Application Protocol (WAP); whose objective was to establish the parameters of security required in the system and to obtain a safe connection.

The middleware system was tested on an Client-Server application based on the 802.11b WLAN. The application consisted of a real-time system video transmission system (server side), and in a video player (client side), both dependent of the QoS and security offered by the server. The results obtained allowed us to verify our goals of designing a middleware system with a flexible and efficient balance between the QoS and the security in a real-time multimedia application.

# Resumen

El desarrollo que han tenido las redes de área local inalámbricas ha generado que equipos de comunicaciones móviles, como los asistentes personales digitales (PDA's), que antes eran simples dispositivos de organización personal, ahora sean vistos como equipos de cómputo móviles. Hoy en día las PDA's son capaces de realizar operaciones comparables a las que realizan las computadoras personales. Permiten ejecutar aplicaciones en un ambiente de movilidad en el cual se puede tener acceso a diferentes tipos de datos, por ejemplo, datos multimedia (voz, video e imágenes). Sin embargo, puesto que son dispositivos pequeños tienen limitaciones en cuanto a poder de cómputo, energia, capacidad de almacenamiento y memoria. Estas limitaciones resultan en desventajas, entre las cuales destacan los bajos niveles de seguridad y calidad de servicio (CS) que son requeridas en la mayoria de aplicaciones multimedia.

Esta tesis presenta un sistema basado en el diseño e implementación de un middleware, cuyo objetivo es incrementar los niveles de seguridad en la transferencia de datos en tiempo real sobre una red de área local inalámbrica. Este sistema permite ajustar de manera flexible los niveles de calidad de servicio y los niveles de seguridad, de tal manera que se establezca un equilibrio entre la calidad de servicio y la seguridad.

En el desarrollo del sistema se utilizaron algoritmos criptográficos de tipo simétrico y asimétrico; la unión de este tipo de algoritmos resulta en el desarrollo de sistemas de seguridad flexibles y ajustables. Así mismo, se realizó una implementación del protocolo de negociación de WTLS (Wireless Transport Layer Security) que es una capa del protocolo WAP (Wireless Application Protocol) cuyo objetivo es la de establecer los parámetros de seguridad requeridos en el sistema y obtener una conexión segura.

El sistema fue probado en una aplicación Cliente-Servidor montado sobre una red de área local inalámbrica 802.11b. La aplicación consistió, del lado del

servidor en un sistema de transmisión de video en tiempo real y del lado del cliente en una aplicación de reproducción de video, dependiente de la calidad de servicio y seguridad proveída por el servidor. Los resultados obtenidos de las pruebas realizadas al sistema muestran un buen balance entre la calidad de servicio y la seguridad.

A mis Padres.

# Agradecimientos

A mis padres: Luis González Villatoro y Carmen Noriega Trujillo †, por su esfuerzo, apoyo y confianza, gracias a ustedes he alcanzado esta meta.

A mi hermana Josefa del Carmen González Noriega por su apoyo en todo momento.

A toda mi familia por brindarme su apoyo y sobre todo por confiar en mi.

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por la beca otorgada para la realización de estos estudios.

Al Centro de Investigación y de Estudios Avanzados del IPN (CINVESTAV-IPN) por darme la oportunidad de realizar mis estudios de maestría en sus instalaciones.

A la biblioteca del departamento de ingeniería eléctrica por su apoyo bibliográfico necesario para el desarrollo de este trabajo.

Al Dr. Pedro Mejía Alvarez por la confianza otorgada al dejarme participar en este trabajo de tesis.

A mis revisores el Dr. Jorge Buenabad Chávez y la Dra. Xiaoou Li por sus criticas y sugerencias a este trabajo.

A los profesores de la sección de computación por su dedicación en cada uno de los cursos que imparten.

A cada uno de mis compañeros y amigos del CINVESTAV-IPN con los que pasamos gratos momentos.

Por último no por ello menos importante, le doy las gracias a todos mis amigos por su confianza al brindarme su amistad.

# Índice general

# Índice de figuras

# Índice de tablas

# Capítulo 1

# Introducción

En este capitulo, se realiza una introducción a nuestro trabajo de tesis, presentando los motivos que dieron origen al problema para el cual esta tesis fue planteada. Además, muestra un panorama general del resto de los capítulos de esta tesis.

## 1.1. Motivación

En los últimos años las redes de área local inalámbricas (WLAN, en ingles Wireless Local Area Network) han ganando mucha popularidad, que se ha visto acrecentada conforme sus prestaciones aumentan y se descubren nuevas aplicaciones. Las WLANs permiten a sus usuarios acceder a información y recursos sin necesidad de estar conectados físicamente a un determinado lugar. Una WLAN por sí misma es móvil, elimina la necesidad de usar cables y establece nuevas aplicaciones añadiendo flexibilidad a la red. Un usuario dentro de una red de área local inalámbrica puede transmitir y recibir datos (como son, voz, vídeo, imágenes, entre otros) dentro de edificios, entre edificios e inclusive en áreas metropolitanas, a velocidades que exceden los de 10 Mbps. Esto hace que la tendencia actual en el desarrollo tecnológico pretenda crear

de manera continuamente dispositivos más pequeños y más portables; por lo tanto, muchos de los fabricantes de computadoras han creado equipos de comunicaciones móviles, como por ejemplo, los asistentes personales digitales (PDA's, Personal Digital Asistent), los que incluyen dentro de sus principales beneficios hardware de comunicación inalámbrica, permitiendo con ello que una gran variedad de aplicaciones de comercio electrónico y multimedia, se ejecuten sobre las PDA's. Sin embargo, como en todo desarrollo tecnológico nuevo, el objetivo es lograr la funcionalidad aunque eso implique no tomar en cuenta otros requerimientos que ayuden a tener un optimo rendimiento, como por ejemplo, el poder de computo, capacidad de almacenamiento y memoria entre otras limitaciones de estos dispositivos. Esto hace que en la mayoria de los casos no exista una compatibilidad y que ademas resulte en vulnerabilidades a la seguridad del sistema.Por otro lado la seguridad podría formar parte del desarrollo de una aplicación, pero por naturaleza los conceptos de seguridad son complejos por si mismos, lo que resulta en una tarea difícil de entender. Además los servicios de seguridad podrían degradar el rendimiento, bajando con ello los niveles de calidad de servicio (QoS) de la aplicación multimedia. Es por eso que casi todos los desarrolladores de aplicaciones multimedia ignoran los mínimos requerimientos de seguridad para garantizar confiabilidad e integridad de los datos.

El integrar la QoS y la seguridad en un sistema multimedia de tiempo real, puede perjudicar al cumplimiento de los requerimientos temporales de la aplicación, ya que a mayor QoS o mayor seguridad, se demandaría un mayor tiempo de cómputo de las aplicaciones. Por lo tanto, la meta es proporcionar un equilibrio entre la QoS, seguridad y el funcionamiento de la aplicación de tal manera que se logre maximizar la QoS y la seguridad sin que se pierdan los plazos de respuesta de la aplicación.

Aunque actualmente existen posibles soluciones para adecuar seguridad, en la práctica encontramos que se necesita la mayoría de las veces de hardware dedicado. Por ejemplo, en la capa enlace de datos, la seguridad se puede

proporcionar por el protocolo WEP(Wired Equivalent Privacy) del estandar IEEE 802.11. En la capa de aplicación, se encuentra la capa de seguridad de sockets (SSL, Secure Sockets Layer) que se utiliza comúnmente. Alternativamente, existen mecanismos de seguridad que se utilizan para proporcionar la autenticación del usuario. Sin embargo, estas soluciones no son suficientes porque el protocolo WEP solo se puede utilizar para las redes inalámbricas y además que tiene defectos en su diseño que ponen en riesgo su seguridad [34]. Además, aunque el SSL se considera seguro, es complicado y difícil de implementar y puede por ello no ser factible en los dispositivos móviles con capacidades de procesamiento y memoria limitados.

La seguridad proporcionada a través de hardware es claramente inflexible pues requiere tener preestablecido los mecanismos de seguridad. Otra cosa importante es que estas soluciones no proporcionan niveles ajustables de seguridad para diversas preferencias del usuario y de la aplicación.

Por otro lado las soluciones actuales no consideran la cantidad de recursos disponibles en los diversos dispositivos en el ambiente. Por lo tanto, estas posibles soluciones son claramente inadecuadas para las aplicaciones multimedia que requieren mayor control sobre los recursos para mantener un nivel aceptable de QoS.

## 1.2. Propuesta

En esta tesis, se describe un mecanismo en el cual se separa entre el desarrollo de aplicaciones y los servicios de seguridad, pero permite la integración de servicios de seguridad ajustables a una determinada aplicación como se observa en la figura 1.1. Los mecanismos de seguridad fueron diseñados para ofrecer diferentes niveles de seguridad que sean ajustables en tiempo de ejecución con lo cual se obtiene flexibilidad dentro de la aplicación. Esta facilidad de ajustarse a las necesidades de seguridad en un determinado momento hace que en un sistema multimedia de tiempo real los parámetros de seguridad no

perjudiquen el funcionamiento del sistema.

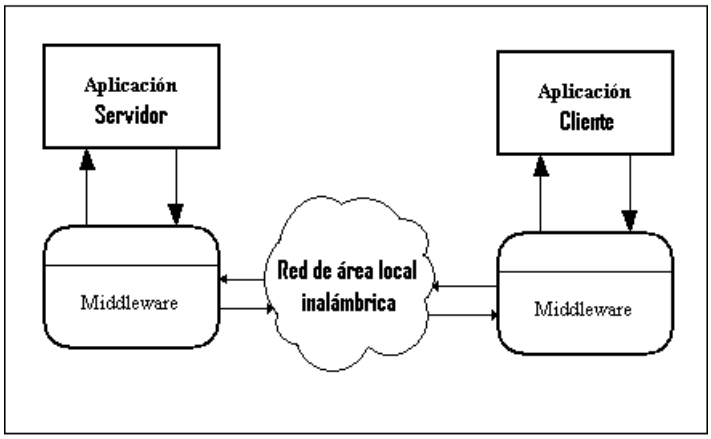


Figura 1.1: Arquitectura general del sistema.

Para separar entre la aplicación y la seguridad se implemento un middleware el cual se encuentra entre las capas de aplicación y transporte del protocolo de comunicaciones TCP /IP. Este middleware es el que se encarga de ofrecer los diferentes niveles de seguridad ajustables, de los cuales podrá disponer una aplicación que se encuentre integrada a este. En la capa del middleware dentro de TCP / IP, los componentes de la seguridad actúan recíprocamente con los encargados de los recursos y las entidades de la QoS para proporcionar soluciones óptimas de seguridad-funcionamiento. Esto reduce de manera considerable la carga, permitiendo que los usuarios de la aplicación puedan tener servicios de seguridad correctamente y eficientemente. Esta solución proporciona servicios de seguridad como son: autenticación y confidencialidad. Para implementar estos servicios de seguridad se usaron algoritmos criptográficos simétricos (DES, TDES y AES) y asimétricos (CCE), además de usar el protocolo de negociación de WTLS(Wireless Transport Layer Secure) en el cual se autentica y se establecen los parámetros de seguridad en una comunicación. Además, permite una solución más flexible y configurable de la seguridad que no requiera de hardware dedica-

do. Los niveles de seguridad se pueden modificar para requisitos particulares para los diversos dispositivos y aplicaciones en tiempo real. Comparado a las soluciones existentes, esta propuesta tiene la ventaja que no depende de los dispositivos de hardware dedicados, es simple y puede trabajar además sobre redes cableadas e inalámbricas. Por ultimo el sistema propuesto tiene la facilidad de extenderse según se vayan desarrollando nuevos algoritmos criptográficos que mejoren aun mas los índices de seguridad existentes hasta este momento.

## 1.3. Contribuciones

A continuación se presentan las principales contribuciones de esta tesis:

1. Desarrollo de un sistema que provee seguridad en la transferencia de datos multimedia en tiempo real en una WLAN y que es capaz de proveer un equilibrio entre la calidad de servicio y la seguridad.

2. Desarrollo de un middleware de seguridad encargado de ajustar los niveles de seguridad para toda aplicación que implemente el paradigma cliente-servidor.

3. Desarrollo de un framework el cual permite crear aplicaciones seguras dentro de una red cableada e inalámbrica.

## 1.4. Organización de la tesis

El resto de este documento está organizado como sigue: en el capítulo 2 se describen los conceptos que tienen relación y que son el fundamento teórico para este trabajo de tesis. Como son los tipos de aplicaciones multimedia, la calidad de servicio de las aplicaciones multimedia, las redes de área local inalámbricas y conceptos de seguridad de la información. En el capítulo 3 se revisan los elementos del diseño e implementación del middleware propuesto

en esta tesis. En el capítulo 4 se incluye la evaluación y análisis de los resultados obtenidos en las pruebas hechas al sistema. Por ultimo en el capítulo 5 se presentan las conclusiones obtenidas para este trabajo.

# Capítulo 2

# Marco Teórico

En este capítulo se revisará la teoría y las tecnologías que son el soporte para el desarrollo de esta tesis. Primeramente, se analiza que es un sistema multimedia, sus características, la calidad de servicio (QoS) que requiere este tipo de sistemas y los tipos de sistemas multimedia que existen. Posteriormente, puesto que este trabajo esta enfocado a sistemas multimedia distribuidos, se estudia la tecnología de las redes de área local inalámbricas (WLAN, por sus siglas en ingles), de las cuales se describen sus principales características de comunicación y seguridad. Por ultimo, se hace referencia a los términos de seguridad en la información que se pueden proporcionar a nivel de aplicación, para este caso, en este capítulo se tratan temas de criptografía.

## 2.1. Sistemas Multimedia

A pesar del paso del tiempo las aplicaciones multimedia son requeridas y utilizadas con una mayor frecuencia. Estas aplicaciones deben ser cooperativas (involucran varios usuarios) y sincronizadas (requieren que las tareas de los usuarios estén coordinadas). Un ejemplo de las aplicaciones multimedia es una simple videoconferencia la cual involucra dos o más usuarios, cada uno

equipado con una cámara de vídeo digital, un micrófono y salida de audio y vídeo. A continuación se describirá que es un sistema multimedia y sus características.

### 2.1.1. Definición multimedia

Etimológicamente, multimedia se traduce como "múltiples medios", por lo tanto se puede definir como "múltiples medios por los cuales la información se almacena, se transmite, se presenta o es percibida". Una Definición mas formal dentro del ámbito computacional, describe al termino multimedia como una tecnología interdisciplinaria orientada al desarrollo de aplicaciones, que aprovechan la capacidad multisensorial del ser humano y la habilidad de las computadoras de procesar la información, no numérica como video, gráficos y audio [31]. Es decir, multimedia es, en esencia, una forma de mezclar diferentes tecnologías de difusión de información, afectando a varios sentidos a la vez para lograr un efecto mayor en la comprensión del mensaje. Es por ello que los productos basados en sistemas multimedia ofrecen combinaciones de texto, audio y vídeo, en un mismo documento que son coordinadas (producidas, controladas y mostradas) por una computadora. Suponen una combinación de estas tecnologías optimizadas a fin de dar un producto atractivo y eficiente para los usuarios. La tecnología multimedia provee un acceso amplio a la información.

### 2.1.2. Recursos de los sistemas multimedia

Los dos recursos sobre los que se basan los sistemas multimedia, son:

- El **Audio**, integrado por sonidos, música, palabras, ruidos u otro tipo de efectos sonoros. Se pueden definir 3 facetas del mensaje de audio: la palabra, es el máximo de inteligibilidad, da fuerza, claridad conceptual, rigor formal, concreción; la música, da ritmo y movimiento; los

efectos sonoros y los silencios, dan matices expresivos que refuerzan los mensajes.

- El **Video**, integrado a su vez por el grupo de gráficos (texto, ilustraciones, animaciones, diagramas o imágenes virtuales 3D) o por un grupo de películas. La imagen provoca emoción, da ambientación y representación creativa.

### 2.1.3. Caraterísticas de los sistemas multimedia

Los sistemas multimedia demandan la entrega a tiempo de flujos de datos a los usuarios. Estos datos pueden ser de tipo audio o vídeo, que son generados y consumidos en tiempo real dentro del sistema. La entrega a tiempo de los elementos individuales es esencial para la integridad de la aplicación. La naturaleza de las tareas ejecutadas por los sistemas de tiempo real existentes, como el control del tráfico aéreo, el control de proceso de fabricación, entre otras aplicaciones, difiere de aquellas ejecutadas en los sistemas multimedia. Las primeras tratan con cantidades de datos pequeños y en algunos casos tienen tiempos límites estrictos, pero el fallo en el cumplimiento de cualquiera de sus tiempos límites de entrega puede producir consecuencias serias e incluso desastrosas (Sistemas de Tiempo Real Críticos "Hard Real Time Systems").

Las consecuencias de un fallo en el cumplimiento en los tiempos límites en los sistemas multimedia pueden causar problemas (especialmente en entornos comerciales como servicios de vídeo bajo demanda, aplicaciones de conferencias de negocios y medicina remota), pero eso no implica consecuencias serias o catastróficas (Sistemas de Tiempo Real Acríticos "Soft Real Time Systems"):

En resumen, los sistemas multimedia son sistemas de tiempo real, que deben ejecutar tareas y entregar sus resultados de acuerdo de una planificación que es determinada externamente. El grado en el que esto se consigue

por el sistema es conocido como la calidad de servicio (Quality of Service, QoS) de que disfruta una aplicación.

### 2.1.4. Tipos de sistemas multimedia

Las investigaciones y desarrollos en el área de la multimedia se puede dividir en dos grandes grupos:

- El primero centrado en el área de estaciones de trabajo independientes, con el software y las herramientas relacionadas, tal como composición musical, enseñanza asistida por computadora, video interactivo, etc.
- El segundo grupo centrado en el intercambio de información multimedia entre esas estaciones de trabajo a través de redes, combinando así los sistemas distribuidos con la multimedia.

### 2.1.5. Sistemas multimedia distribuidos

Un sistema multimedia distribuido, se puede definir como un sistema en donde los datos de medios continuos en el cual la transmision de datos es constante y discretos en el cual los datos son enviados en bloques en un tiempo determinado, los cuales deben ser transportados a través de redes de computadoras [4]. Por ejemplo, se puede mencionar a las aplicaciones de telefonía sobre Internet y videoconferencia, por nombrar algunas de ellas. Dichas aplicaciones son viables con las actuales redes y sistemas de propósito general, a pesar de que a menudo la calidad del audio y del vídeo resultante esta lejos de ser satisfactoria. Las aplicaciones multimedia generan y consumen flujos de datos continuos en tiempo real. Éstos contienen grandes cantidades de audio, vídeo y otros elementos de datos dependientes del tiempo con lo que resulta esencial el procesamiento, la entrega a tiempo de los elementos individuales de datos.

La especificación de un flujo multimedia se expresa en términos de valores aceptables para la tasa a la que los datos pasan desde la fuente al destino

(ancho de banda), el retardo en la entrega de cada elemento (latencia) y la tasa a la que se pierden o se desechan los elementos.

- La latencia es particularmente importante en aplicaciones interactivas.

- En las aplicaciones multimedia a menudo resulta aceptable una pequeña perdida de datos de los flujos multimedia ya que las aplicaciones pueden volver a sincronizarse con los elementos que siguen a aquellos perdidos.

- Las aplicaciones multimedia, son a menudo, altamente distribuidas y operan sobre entornos de computación distribuida de propósito general. Compiten, por lo tanto, con otras aplicaciones distribuidas por el ancho de banda de la red y por los recursos de las estaciones de trabajo de los usuarios y servidores.

- Los requisitos de los recursos de las aplicaciones multimedia son dinámicos. Una videoconferencia puede necesitar más o menos ancho de banda dependiendo del aumento o de la disminución del número de conferencistas. El uso de los recursos de cómputo en cada computadora varía, ya que cambia, por ejemplo, el número de flujo de vídeo que debe mostrar. Las aplicaciones multimedia pueden suponer otras cargas variables o intermitentes. Por ejemplo, la ejecución de una clase multimedia puede incluir una actividad de simulación con un uso intensivo del procesador.

- A menudo los usuarios desean equilibrar los costos en recursos de las aplicaciones multimedia con otras actividades. Podrian reducir sus peticiones de ancho de banda para el vídeo en una aplicación de conferencia para permitir que se realice una conversación de audio separado, o pueden querer seguir programando o editando textos mientras están participando en la conferencia.

La reserva y la planificación de los recursos antes mencionados son diseñados para satisfacer las necesidades tanto de las aplicaciones multimedia como las otras; a lo cual se le denomina gestión de la calidad de servicio.

### Gestores de calidad de servicio multimedia

Los sistemas de gestión de QoS están diseñados para responder a todas las necesidades de recursos para las aplicaciones multimedia, gestionando de forma dinámica los recursos disponibles y variando las reservas en respuestas a los cambios en la demanda y las prioridades de los usuarios. Un sistema de gestión QoS debe gestionar todos los recursos de cómputo y de comunicación necesarias para adquirir, procesar y transmitir flujos de datos multimedia, especialmente cuando los recursos son compartidos entre diferentes aplicaciones.

Los recursos requeridos para la gestión de QoS incluyen ancho de banda de la red, ciclos de procesador y capacidad de memoria. También consideran la capacidad del disco duro en el servidor.

En un sistema distribuido abierto, las aplicaciones multimedia pueden ser iniciales y utilizadas sin anuncio previo. Pueden coexistir varias aplicaciones en la misma red e incluso en la misma estación de trabajo. Por lo tanto, la necesidad de la gestión de la QoS surge independientemente de la cantidad total de ancho de banda de los recursos o de capacidad de memoria de un sistema. Es necesario gestionar la QoS para garantizar que las aplicaciones sean capaces de obtener la cantidad de recursos necesarios en los momentos requeridos, incluso cuando otras aplicaciones estén compitiendo por esos recursos.

Asi mismo es importante tomar en cuenta que medio de transmisión se va a utilizar para conocer sus características y poder administrar de manera satisfactoria sus recursos en beneficio de las aplicaciones multimedia.

## 2.2. Redes de área local inalámbrica

Una red de área local inalámbrica (WLAN, Wireless Local Area Network, por sus siglas en ingles) es un sistema flexible de comunicación de datos implementado como extensión, o como alternativa, a una red de área local cableada. Las redes WLAN trasmiten y reciben datos por el aire mediante la tecnología de radio frecuencia, minimizando la necesidad de disponer de conexiones cableadas lo que, a su vez, combina la conectividad de datos con movilidad de usuario. Las redes WLAN proporcionan toda la funcionalidad de las redes de área local, sin las correspondientes restricciones físicas. Además de ofrecer movilidad al usuario final dentro de un entorno de conexión en red, las redes WLAN permiten una portabilidad de la red física, lo que permite mover a las redes de área local con los usuarios que las emplean. Las redes de área local inalámbricas se diferencian de las convencionales principalmente en la Capa Física y la Capa de Enlace de Datos, según el modelo de referencia OSI. La capa física indica como son enviados los bits de una computadora a otra. La capa de Enlace de Datos (denominada MAC), se encarga de describir como se crean los paquetes de datos y verifican los bits de modo que no tengan errores. Las demás capas forman los protocolos o utilizan puentes, ruteadores o compuertas para conectarse. Los dos métodos para remplazar la capa física en una red inalámbrica son la transmisión de Radio Frecuencia y la Luz Infrarroja.

### 2.2.1. Medios de transmisión inalámbrica

Las redes WLAN utilizan ondas electromagnéticas dentro del espectro de radio frecuencia (RF) e infrarrojo(IR) para transferir datos desde un punto a otro. Los organismos encargados de regular la concesión de licencias sobre el espectro de radio frecuencia han dispuesto una serie de frecuencias para el uso comercial sin licencia. Estas bandas ISM ( bandas de uso industrial, científico y medico) incluyen las bandas de 900 MHz, 2.4 GHz y 5 GHz utilizados por

muchos dispositivos comerciales de comunicación inalámbrica. La mayoría de los dispositivos WLAN que están apareciendo están diseñados para operar en la banda 2.4 GHz debido a su disponibilidad global y a las menores interferencias que en ella se generan. Hay varios medios de transmisión capaces de transferir datos mediante ondas electromagnéticas. Al igual que sucede con casi todas las tecnologías, cada uno de esos medios tiene sus propias ventajas y limitaciones.

### Sistemas infrarojos (IR)

Los sistemas infrarrojos no permiten construir soluciones practicas para la implementación de WLANs corporativas y por lo tanto no se utiliza ampliamente. Estos son capaces de transmitir datos aprovechando las frecuencias ubicadas en las cercanías del espectro electromagnético de la luz visible, aunque por debajo de las frecuencias de esta. Estas bandas tienen las mismas limitaciones de la luz visible ya que no pueden penetrar objetos no transparentes como paredes, suelos y techos. Como resultado, las WLANs que transmiten mediante rayos infrarrojos están restringidas a operar, como mucho, dentro de la misma habitación, estando, además, limitadas a comunicaciones de corto alcance, sin oclusiones de la línea de visión.

### Sistemas de radio de banda estrecha

Los sistemas de radio de banda estrecha transmiten y reciben datos en una frecuencia de radio especifica. Los diferentes usuarios se comunican en frecuencias alternativas o canales, para garantizar un cierto nivel de intimidad y evitar las interferencias. Los receptores de radio se construyen para ponerse en escucha solo en su frecuencia designada, filtrando todas las restantes. La limitación natural de este sistema resulta clara: si otro transceptor está operando a la misma frecuencia y dentro del rango de cobertura, se producirá interferencia y los datos se perderán o corromperán. Otra desventaja de implementar las tecnologías de banda estrecha es que, en muchos países, es preciso obtener

una licencia de las autoridades correspondientes para cada ubicación donde se quiera implementar cualquiera de estos sistemas de transmisión.

### Sistemas de radio de banda ancha(Expansión de espectro)

Los sistemas de radio de banda ancha en lugar de utilizar una única frecuencia, la tecnología de expansión de espectro, como su nombre sugiere, recorre la banda de frecuencias disponibles para transmitir los datos de manera fiable. Originalmente empleada por los militares, la tecnología de expansión de espectro distribuye la señal sobre un amplio rango de frecuencias de manera uniforme, consumiendo así un mayor ancho de banda a cambio de conseguir una mayor fiabilidad, integridad y seguridad de las comunicaciones. Esta forma de transmisión de banda ancha permite a los dispositivos evitar las interferencias y los ruidos provocados por otras señales. Existen dos tecnologías de espectro expandido: la tecnología de expansión de espectro por secuencia directa (DSSS, Direct Seequence Spread Spectrum, por sus siglas en ingles) y la tecnología de expansión de espectro por salto de frecuencia (FHSS, Frequency Hopping Spread Spectrum, por sus siglas en ingles).

- **La secuencia directa (DSSS):** En este método, el flujo de bits de entrada se multiplica por una señal de frecuencia mayor (señal portadora), basada en una función de propagación determinada. El flujo de datos original puede ser entonces recobrado en el extremo receptor relacionándolo con la función de propagación conocida. Este método requiere un procesador de señal digital (DSP) para relacionar la señal de entrada.

- **El salto de frecuencia (FHSS):** Este método es una técnica en la cual los dispositivos receptores y emisores se mueven de forma síncrona en un patrón determinado de una frecuencia a otra, brincando ambos al mismo tiempo y en la misma frecuencia predeterminada. Como en el método de secuencia directa, los datos deben ser reconstruidos en

base del patrón de salto de frecuencia. Este método es viable para las redes inalámbricas, pero la asignación actual de las bandas ISM no es adecuada, debido a la competencia con otros dispositivos, como por ejemplo las bandas de 2.4 y 5.8 MHz que son utilizadas por hornos de Microondas.

### 2.2.2. Configuraciones de redes de área local inalámbricas

Cada computadora en una WLAN es llamada estación o nodo. Una WLAN puede ser configurada como una red punto a punto (llamada red ad-hoc) donde dos o mas estaciones de trabajo directamente intercambian información de una a otra, o bien puede ser en modo infraestructura donde un punto de acceso (AP, Access Point, por sus siglas en ingles) central encierra todas las comunicaciones entre las estaciones de trabajo que se encuentran alrededor de él.

#### Modo ad-hoc

Cuando dos o mas nodos están lo suficientemente cerca como para comunicarse uno con otro, se forma un conjunto de servicio básico (BSS, Basic Service Set, por sus siglas en ingles). El mínimo BSS consiste en dos estaciones. Un BSS que se encuentra solo y que no esta conectado a un AP es llamado un conjunto de servicio básico independiente (IBSS, Independent Basic Service Set) o una red ad-hoc (ver la figura 2.1). Un conjunto de servicio extendido (ESS, Extended Service Set, por sus siglas en ingles) es formado cuando dos o más BSSs operan dentro de la misma red. Una red ad-hoc es una red cuando las estaciones se comunican solo a través de la configuración peer-to-peer. No hay APs, y no se necesita permiso para la comunicación. La mayoría de estas redes son espontáneas y se pueden configurar de manera rápida.

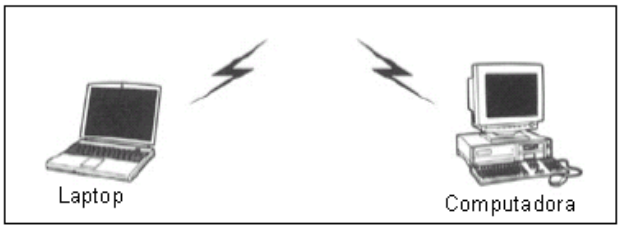


Figura 2.1: Red ad-hoc.

### Modo infraestructura

Se dice que una WLAN esta siendo operada en un modo infraestructura (ver figura 2.2) cuando dos o mas BSSs son interconectados usando un punto de acceso. Un punto de acceso actúa como un concentrador para los nodos de la red inalámbrica. Un AP encamina el trafico entre los BSSs. Algunas veces los puntos de acceso están conectados a las redes comunes (es decir las redes cableadas) para proporcionar o compartir los recursos de dicha red, a los nodos de la red inalámbrica.


Figura 2.2: Red inalámbrica en modo Infraestructura

Las redes inalámbricas en modo infraestructura son la configuración mas común en redes grandes.

### 2.2.3. Estándares de redes inalámbricas

#### Bluetooth

Bluetooth representa una alianza entre las comunicaciones móviles y las compañías de computo móvil. La alianza se formo en 1998 promovida por las manufactureras como Ericsson, Nokia, IBM, Intel y Toshiba. Una de las razones para el desarrollo de Bluetooth fue que entre la gran variedad de opciones de conectividad no se tenia una compatibilidad entre una y otra [5]. La pila de protocolos de Bluetooth no esta representada por las siete capas clásicas del modelo de referencia OSI. Esto es porque Bluetooth es un intento por ínteroperar con módems, teléfonos y otros dispositivos. Bluetooth esta catalogada como una red de área personal (PAN, Personal Area Network, por sus siglas en ingles). Es decir es una red de modo ad-hoc y su principal ventaja es su mínimo consumo de energía.

#### HiperLAN

HiperLAN es un estándar de telecomunicaciones desarrollado en Europa por el ETSI (European Telecommunications Standards Institute). Es considerado como el estándar mas parecido al estándar 802.11b de IEEE [15]. HiperLAN tiene dos estándares:

- **HiperLAN/1:** En 1991 el ETSI formo el comité sub-técnico RES10 para desarrollar HiperLAN. El resultado fue el estándar de HiperLAN/1 (aprobado en 1996) que define la capa física (PYH) y la capa de acceso al medio (MAC) especificaciones para las redes inalámbricas de alta velocidad de comunicaciones. HiperLAN/1 usa GMSK (gaussian minimum shift keying) y especifica la tasa de transferencia por arriba de 20 Mbps entre dispositivos portátiles. Una ventaja de HiperLAN/1 es que trabaja en una ancho de banda dedicado (5.1 a 5.3 GHz, el cual solo esta disponible en Europa) también no usa la tecnología de Spread Spectrum en orden para coexistir con otra frecuencia de radio como es

el caso de el rango ISM 2.4GHz. También, el protocolo usa una variante de CSMA/CA y opcionalmente incluye cifrado de la información. Otra característica de HiperLAN/1 es un encaminamiento ad-hoc. Por ejemplo, si su destino esta fuera de rango, los nodos intermedios automáticamente lo reenvían encontrando la mejor ruta disponible dentro de la red HiperLAN/1 (las rutas son recalculadas automáticamente de manera regular).

- **HiperLAN/2:** En 1997, el ETSI formo el grupo BRAN (Broadband Radio Area Network) para trabajar sobre HiperLAN/2, el cual fue aprobado en febrero de 2000. HiperLAN/2 es un rediseño de HiperLAN/1 y fue el primer estándar en usar OFDM (). HiperLAN/2 y IEEE 802.11a son muy similares en su uso de la banda 5GHz y OFDM para obtener tasas de transferencias superiores a 54Mbps. La principal diferencia entre los dos estándares esta en la parte de acceso al medio de los sistemas. HiperLAN/2 usa TDM (Time Division Multiplexing), por otro lado 802.11a/g usa CSMA/CD.

### HomeRF

HomeRF es una etiqueta para un grupo industrial que se unió en 1998, para desarrollar un estándar para conectividad inalámbrica entre computadoras personales y dispositivos electrónicos. El estándar que resultó es el Shared Wireless Access Protocol (SWAP), el cual permite transmitir voz y datos con una tasa de transferencia por arriba de 1.6 Mbps.

### IEEE 802.11

En el ámbito de las redes WLANs el estándar que más ha destacado es la especificación de la IEEE: 802.11. Liberada en 1997, hoy es la especificación más utilizada ya que brinda a sus usuarios flexibilidad, simplicidad de uso y efectividad de costos. Este estándar especifica los parámetros de dos capas

del modelo OSI: la capa física (PHY) y la capa de control de acceso al medio (MAC). La capa MAC tiene tres funciones principales: controlar el canal de acceso, mantener la calidad de servicio (QoS) y proveer seguridad. La capa MAC del IEEE 802.11 soporta servicios de seguridad para las aplicaciones de las capas superiores tales como la autenticación y la privacidad, pero la especificación IEEE 802.11 sólo da un método débil de autenticación y para asegurar la privacidad cuenta con una opción llamada Wired Equivalent Privacy (WEP) que no ha cumplido con su propósito. Al inicio, el 802.11 especificaba un bajo índice de transferencia real, hasta de 2Mbps. El estándar ha sido mejorado en dos diferentes especificaciones: el estándar 802.11b conocido como Wi-Fi, que permite, en teoría, una funcionalidad inalámbrica comparable con Ethernet con un índice de transferencia real de hasta 11Mbps en la banda comercial y el estándar 802.11a que permite hasta 54Mbps en la banda industrial, científica y médica. La arquitectura de una WLAN IEEE 802.11 consiste, generalmente, de un conjunto de servicios básicos (BSS) que se interconectan a un sistema de distribución (DS) para formar un conjunto de servicios extendidos (ESS) como se muestra en la figura 3.1. Cada estación puede transmitir directamente a cualquier otra estación en el mismo BSS (modo ad-hoc). Por otro lado, para transmitir a estaciones pertenecientes a diferentes BSS, las estaciones pasan a través de un punto de acceso (AP) que es una unidad de enlace que implementa ambos protocolos MAC, el de la IEEE 802.11 y el del DS (modo de infraestructura).

### 2.2.4. Seguridad en redes inalámbricas

El acceso tan flexible y rápido sin necesidad de usar cables, es el motivo por el cual las redes inalámbricas han tenido tanto desarrollo, pero en consecuencia resulta en un problema grande en lo que a seguridad se refiere. Existen diferentes métodos para garantizar seguridad dentro de una comunicación inalámbrica como: filtrado de direcciones MAC, el uso del protocolo WEP(Wired Equivalent Privacy, por sus siglas en ingles), conexión a través

de CNAC (Closed Network Access Control, por sus siglas en ingles).

Estos métodos aunque intentan garantizar niveles de seguridad dentro de la red a diferentes niveles del modelo OSI, en la mayoría de los casos requieren implementar otros mecanismos para elevar los niveles de seguridad de la red, dentro de los cuales podemos mencionar: VPNs, implementar seguridad a nivel de la capa de aplicación usando seguridad basada en criptografía, entre otros.

### Filtrado de direcciones MAC

Este método consiste en la creación de una tabla de acceso en cada uno de los puntos de acceso de la red de área local inalámbrica. Dicha tabla contiene las direcciones MAC (Media Access Control, por sus siglas en ingles) de las tarjetas de red inalámbricas que se pueden conectar a una determinada red de área local inalámbrica. Como toda tarjeta de red esta posee una dirección MAC única, que permite autenticar el equipo. Aunque, este método tiene como ventaja su sencillez, por lo cual se puede usar para redes relativamente pequeñas. Sin embargo, posee muchas desventajas que lo hacen impráctico para uso en redes medianas o grandes:

- No es un sistema escalable y flexible, porque cada vez que se desee autorizar o dar de baja un equipo, es necesario editar las tablas de direcciones de todos los puntos de acceso. Después de cierto número de equipos o de puntos de acceso, la situación se torna inmanejable.

- El formato de una dirección MAC no es amigable (esta representado en 48 bits en hexadecimal), lo que puede llevar a cometer errores en la manipulación de las listas.

- Las direcciones MAC viajan sin cifrar por el aire. Un intruso podría obtener direcciones MAC de tarjetas autorizadas en la red empleando un sniffer y luego asignarle una de estas direcciones a su computado-

ra, empleando programas como: AirJack6 o WellenReiter, entre otros, haciendo se pasar por un usuario valido.

### Protocolo WEP

Este protocolo forma parte de la especificación 802.11 [21, 34] y se creo con la finalidad de proteger los datos que se transmiten en una comunicación inalámbrica mediante el uso de mecanismos de cifrado. WEP opera a nivel 2 del modelo OSI y es soportado por la mayoría de los dispositivos inalámbricos.

WEP cifra de la siguiente manera (ver figura 2.3):

- A la trama en claro se le calcula un código de integridad (Integrity Check Value, ICV) mediante el algoritmo CRC-32. Dicho ICV se concatena con la trama y es empleado más tarde por el receptor para comprobar si la trama ha sido alterada durante el transporte.

- Se escoge una clave secreta compartida entre emisor y receptor. Esta clave puede poseer 40 ó 128 bits.

- Si se empleara siempre la misma clave secreta para cifrar todas las tramas, dos tramas en claro iguales producirían tramas cifradas similares. Para evitar esta vulnerabilidad, se concatena la clave secreta con un número aleatorio llamado vector de inicialización (IV) de 24 bits. El IV cambia con cada trama.

- La concatenación de la clave secreta y el IV (conocida como semilla) se emplea como entrada de un generador RC4 de números seudoaleatorios. El generador RC4 es capaz de generar una secuencia seudoaleatoria (o cifra de flujo) tan larga como se desee a partir de la semilla.

- El generador RC4 genera una cifra de flujo, del mismo tamaño de la trama a cifrar más 32 bits (para cubrir la longitud de la trama y el ICV).

- Se hace un XOR bit por bit de la trama con la secuencia de clave, obteniéndose como resultado la trama cifrada.
- El IV y la trama se transmiten juntos.

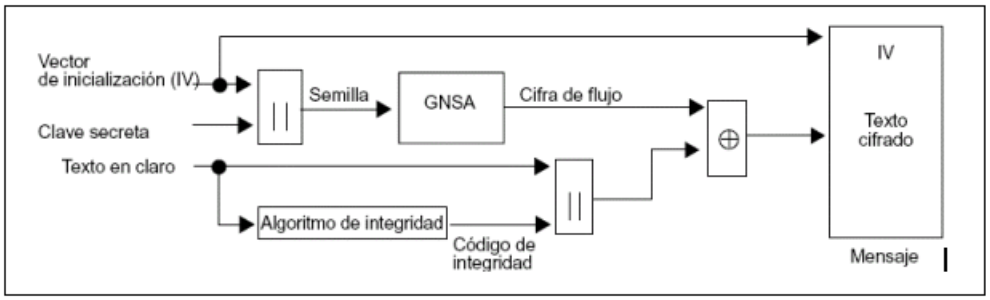


Figura 2.3: Proceso de cifrado WEP.

En el receptor se lleva a cabo el proceso de descifrado (figura 2.4):

- Se emplean el IV recibido y la clave secreta compartida para generar la semilla que se utilizó en el transmisor.
- Un generador RC4 produce la cifra de flujo a partir de la semilla. Si la semilla coincide con la empleada en la transmisión, la cifra de flujo también será idéntica a la usada en la transmisión. Se efectúa un XOR bit por bit de la cifra de flujo y la trama cifrada, obteniéndose de esta manera la trama en claro y el ICV.
- A la trama en claro se le aplica el algoritmo CRC-32 para obtener un segundo ICV, que se compara con el recibido.
- Si los dos ICV son iguales, la trama se acepta; en caso contrario se rechaza.

WEP resuelve aparentemente el problema de la confidencialidad en la comunicación inalámbrica. Sin embargo, existen tres situaciones que hacen que WEP no sea seguro:

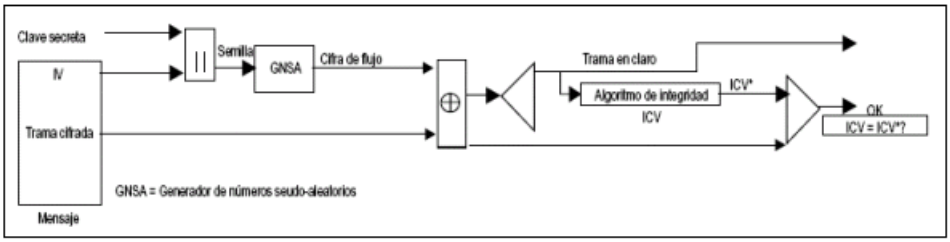


Figura 2.4: Proceso de descifrado de WEP.

- En casi todas las redes de área local inalámbricas establecidas se emplea a WEP con claves de cifrado estáticas. Esto hace posible que un atacante acumule grandes cantidades de tramas cifradas con la misma clave y pueda intentar un ataque por fuerza bruta.
- El IV que se utiliza es de longitud insuficiente (24 bits). Dado que cada trama se cifra con un IV diferente, solamente es cuestión de tiempo para que se agote el espacio de 224 IV distintos. Esto no es problemático en una red casera con bajo tráfico, pero en una red que posea alto tráfico se puede agotar el espacio de los IV en más o menos 5 horas. Si el atacante logra conseguir dos tramas con IV idéntico, puede efectuar un XOR entre ellas y obtener las tramas en claro mediante un ataque estadístico. Con la trama en claro y su respectivo cifrado se puede obtener la cifra de flujo; conociendo el funcionamiento del algoritmo RC4 es posible entonces obtener la clave secreta y descifrar toda la conversación.
- WEP no ofrece servicio de autenticación. El cliente no puede autenticar a la red, ni al contrario; basta con que el equipo móvil y el punto de acceso compartan la clave WEP para que la comunicación pueda llevarse a cabo.

Existen herramientas gratuitas que facilitan el trabajo de romper la clave secreta de enlaces protegidos con WEP como son : WEPCrack, que consiste

en una serie de scripts escritos en lenguaje Perl diseñados para analizar un archivo de captura de paquetes de un sniffer o bien AirSnort hace lo mismo, pero integra las funciones de sniffer y analizador de claves y por lo tanto es más fácil de usar. AirSnort captura paquetes pasivamente y obtiene la clave WEP cuando ha capturado suficientes datos.

#### CNAC (Closed Network Access Control)

Consiste en usar el identificador de la red de área local inalámbrica (SSID, Server Set ID) como contraseña para acceder a la red, tratando de ser un mecanismo de autenticación. Este identificador de red es muy fácil de conseguir ya que es enviado por los clientes al asociarse o autenticarse en el punto de acceso. Por lo que tampoco garantiza que usuarios no autorizados tengan acceso a la red.

## 2.3. Seguridad en la información

El concepto de seguridad en la información es mucho mas amplio que la simple protección de los datos a nivel lógico. Para proporcionar una seguridad real se deben de tener en cuenta múltiples factores, tanto internos como externos. En primer lugar hay que caracterizar el sistema que va albergar la información para poder identificar las amenazas y en este sentido se puede hacer la siguiente subdivisión:

- **Sistemas aislados:** Son los que no están conectados a ningún tipo de red. De unos años a esta fecha se han convertido en minoría, debido al auge que ha experimentado Internet.
- **Sistemas interconectados:** Hoy en día casi cualquier computadora pertenece a alguna red, enviando y recibiendo información del exterior casi constantemente. Esto hace que las redes de computadoras sean

cada vez mas complejas y supongan un peligro potencial que no puede en ningún caso ser ignorado.

Es por ello que el objetivo principal de cualquier comunicación y administración de datos confiable es cumplir con los principios de la seguridad computacional, que se resumen en los siguientes servicios [1]:

- **Confidencialidad.** Los componentes del sistema serán accesibles sólo por aquellos usuarios autorizados. Es decir se refiere a que la información sólo pueda ser leída por personas autorizadas. Por ejemplo, en la comunicación por teléfono, que alguien intercepte la comunicación y escuche la conversación quiere decir que no existe confidencialidad. Si mandamos una carta y por alguna razón alguien rompe el sobre para leer la carta, ha violado la confidencialidad.

- **Integridad.** Los componentes del sistema sólo pueden ser creados y modificados por los usuarios autorizados. Es decir se refiere a que la información no pueda ser alterada en el transcurso de ser enviada. Por ejemplo, cuando compramos un boleto de avión y los datos del vuelo son cambiados, puede afectar los planes del viajero. Una vez hecho un deposito en el banco, si no es capturada la cantidad correcta causará problemas. La integridad es muy importante en las transmisiones militares ya que un cambio de información puede causar graves problemas.

- **Disponibilidad.** Los usuarios deben tener disponibles todos los componentes del sistema cuando así lo deseen.

- **Autenticación.** La autenticación es el proceso de verificar y asegurar la identidad de las partes involucradas en una transacción. Es decir se refiere a que se pueda confirmar que el mensaje recibido haya sido mandado por quien dice lo mando o que el mensaje recibido es el que se esperaba. Por ejemplo, cuando se quiere cobrar un cheque a nombre de

alguien, quien lo cobra debe de someterse a un proceso de verificación de identidad para comprobar que en efecto es la persona quien dice ser, esto en general se lleva a cabo con una credencial que anteriormente fue certificada y acredita la identidad de la persona que la porta. La verificación se lleva a cabo comparando la persona con una foto o con la comparación de una firma convencional.

- **No Repudio.** El no repudio está asociado a la aceptación de un protocolo de comunicación entre emisor y receptor (cliente y servidor) normalmente este proceso se lleva a cabo a través de la autenticación. se refiere a que no se pueda negar la autoría de un mensaje enviado.

Cuando se diseña un sistema de seguridad, una gran cantidad de problemas pueden ser evitados si se puede comprobar la autenticidad, garantizar la confiabilidad, asegurar integridad y el no-rechazo de un mensaje. La criptografía simétrica y asimétrica conjuntamente con otras técnicas, permiten en un porcentaje alto lograr satisfactoriamente resolver los problemas planteados anteriormente .

### 2.3.1. Criptografía

La palabra criptografía proviene del griego kryptos, que significa ocultar y gráphein, escribir, es decir, escritura oculta [18]. La criptografía ha sido usada a través de los años para mandar mensajes confidenciales cuyo propósito es que sólo las personas autorizadas puedan entender el mensaje. Alguien que quiere mandar información confidencial aplica técnicas criptográficas para poder ocultar el mensaje (lo llamaremos cifrar o encriptar), manda el mensaje por una línea de comunicación que se supone insegura y después solo el receptor autorizado pueda leer el mensaje .ºculto"(lo llamamos descifrar o descencriptar)

La criptografía se divide en dos grandes ramas, la criptografía de clave privada o simétrica y la criptografía de clave pública o asimétrica.

### 2.3.2. Criptografía simétrica

La criptografía simétrica se refiere al conjunto de métodos que permiten tener comunicación segura entre las partes siempre y cuando anteriormente se hayan intercambiado la clave correspondiente que llamaremos clave simétrica. La simetría se refiere a que las partes tienen la misma llave tanto para cifrar como para descifrar. Este tipo de criptografía se conoce también como criptografía de clave privada o criptografía de llave privada. La criptografía simétrica ha sido la más usada en toda la historia. Esta a podido ser implementada en diferente dispositivos, manuales, mecánicos, eléctricos, hasta los algoritmos actuales que son programables en cualquier computadora. La idea general es aplicar diferentes funciones al mensaje que se quiere cifrar de tal modo que solo conociendo una clave pueda aplicarse de forma inversa para poder así descifrar.


Figura 2.5: Esquema e cifrado simétrico.

#### DES

Es el algoritmo simétrico mas extendido mundialmente. Se basa en el algoritmo LUCIFER [18], que había sido desarrollado por IBM a principios de los setenta y fue adoptado como estándar por el gobierno de EE.UU. para comunicaciones no clasificadas en 1976.

El algoritmo DES codifica bloques de 64 bits empleando claves de 56 bits. es una red de Feistel de 16 rondas, mas dos permutaciones, una que se aplica

al principio ($P_i$) y otra que se aplica al final ($P_f$), tales que $P_i = P_f^{-1}$
La función $f$(figura 2.6) se compone de una permutación de expansión ($E$), que convierte el bloque de 32 bits correspondiente en uno de 48 bits. después realiza un or-exclusivo con el valor $K_i$ , también de 48 bits, aplica ocho S-Cajas de 6*4 bits y efectúa una nueva permutación $P$. Se calcula un total de 16 valores de $K_i$ (figura 2.7), uno para cada ronda, efectuando primero una permutación inicial $EP1$ sobre la clave de 64 bits, llevando acabo desplazamientos a la izquierda de cada una de las dos mitades -de 28 bits- resultantes y realizando finalmente una elección permutada ($EP2$) de 48 bits en cada ronda, que será la $K_i$. Los desplazamientos a la izquierda son de dos bits, salvo para las rondas 1, 2, 9 y 16, en las que se desplaza solo un bit. Nótese que aunque la claves para el algoritmo DES tiene en principio 64 bits, se ignoran ocho de ellos -un bit de paridad por cada byte de la clave–, por lo que en la practica se usan solo 56 bits.

Para descifrar basta con usar el mismo algoritmo (ya que $P_i = P_f^{-1}$) empleando las $K_i$ en orden inverso.

### TDES

Es un algoritmo que surge con la necesidad de elevar los índices de seguridad del algoritmo DES. Consiste en aplicar varias veces el algoritmo DES con diferentes claves al mensaje original. Se puede hacer ya que DES no presenta estructura de grupo. Triple-DES corresponde a la siguiente ecuación:

$$C = E_{K3}(E_{K2}^{-1}(E_{K1}(M)))$$

es decir, codificamos con la subclave $K_1$, decodificamos con $K_2$ y volvemos a codificar con $K_1$. La clave resultante es la concatenación de $K_1$ y $K_2$, con una longitud de 112 bits. Es decir TDES consiste en aplicar 3 veces DES de la siguiente manera: la primera vez se usa una clave K1 junto con el bloque $B_0$, en modo de cifrado, obteniendo el bloque $B_1$. La segunda ves se toma a $B_1$ con la clave $K_2$, diferente a $K_1$ de forma inversa, en modo de descifrado

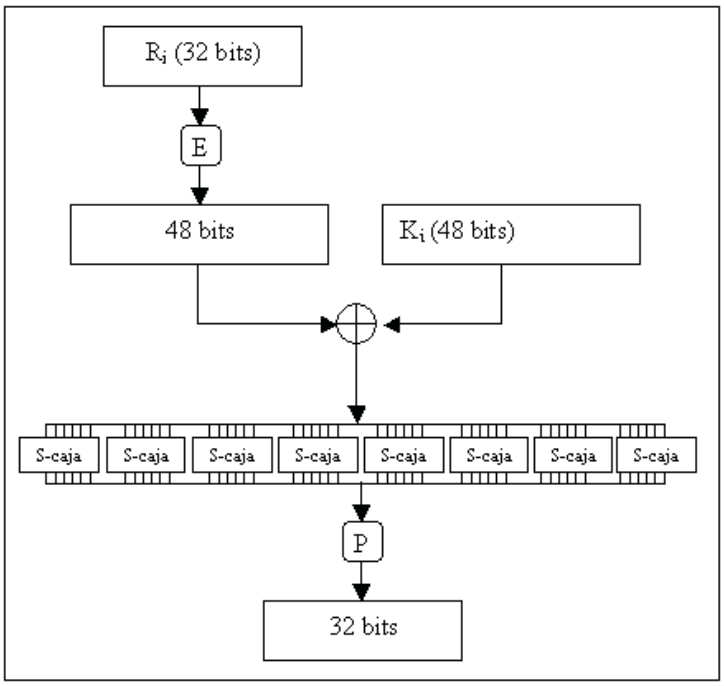


Figura 2.6: Esquema de la función $f$ del algoritmo DES.

y la tercera vez a $B_2$ con una clave $K_3$ diferente a $K_1$ y $K_2$ en modo cifrado (ver figura **??**), es decir, aplica de la interacción 1 a la 16 a $B_0$ con la clave $K_1$, después aplica de la 16 a la 1, a $B_1$ con la clave $K_2$, finalmente aplica una vez mas de la 1 a la 16 a $B_2$ usando la clave $K_3$, obteniendo finalmente a $B_3$. En cada una de estas tres veces aplica el modo de operación más adecuado.

### AES (Algoritmo Rijndael)

En octubre de 2000 el NIST (National Institute for Standars and technology) anunciaba oficialmente la adopción del algoritmo Rijndael [18], como nuevo estándar avanzado de cifrado (AES) para su empleo en aplicaciones criptográficas no militares, culminando así un proceso de mas de tres años, encaminado a proporcionar a la comunidad internacional un nuevo algoritmo

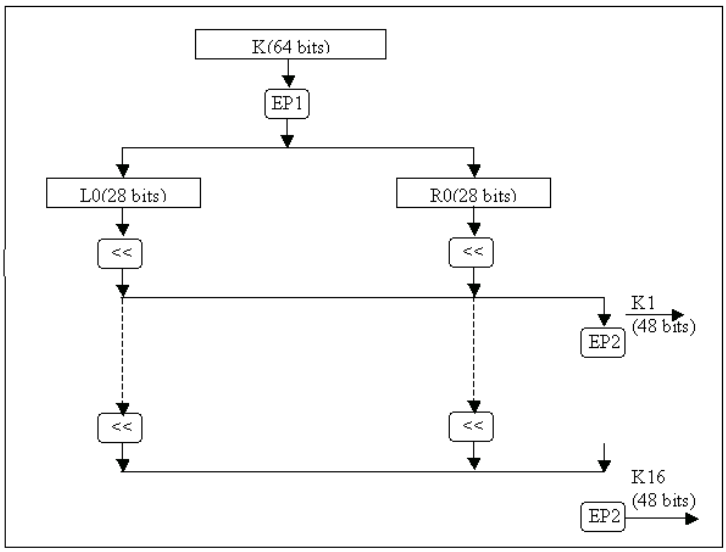


Figura 2.7: Calculo de las $K_i$ para el algoritmo DES.

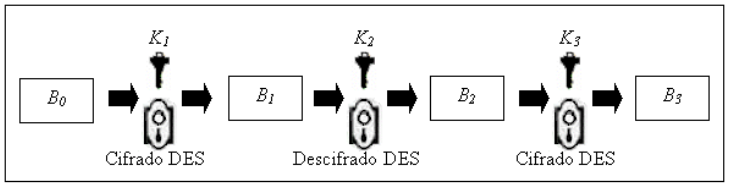


Figura 2.8: Esquema de cifrado de TDES.

de cifrado potente, eficiente y fácil de implementar. El cual sustituiría a DES.

AES es un sistema de cifrado por bloques, diseñado para manejar longitudes de clave y de bloque variables, ambas comprendidas entre los 128 y los 256 bits. realiza varias de sus operaciones internas a nivel de byte, interpretando estos como elementos de un cuerpo de Galois $GF(2^8)$. El resto de sus operaciones se efectúan en términos de registros de 32 bits. sin embargo, en algunos casos, una secuencia de 32 bits se toma como un polinomio de grado inferior a 4, cuyos coeficientes son a su vez polinomios en $GF(2^8)$. AES tiene definido cada ronda como una composición de cuatro funciones invertibles

formando tres capas, diseñadas para proporcionar resistencia frente a criptoanálisis lineal y diferencial. Cada una de las funciones tiene un propósito preciso:

- **La capa de mezcla lineal:** funciones desplazar fila (ShiftRows) y mezclar columnas (Mix Column), permiten obtener un alto nivel de difusión a lo largo de varias rondas.
- **La capa no lineal:** función ByteSub, consiste en la aplicación paralela de s-cajas con propiedades optimas de no linealidad.
- **La capa de adición de clave:** la función AddRounKey es un simple or-exclusivo entre el estado intermedio y la subclave correspondiente a cada ronda.

Tabla 2.1: Numero de rondas para AES en función de los tamaños de bloque y clave.

| | | Tamaño de | bloque | |
|---|---|---|---|---|
| | | 128 bits | 192 bits | 256 bits |
| Tamaño de | 128 bits | 10 | 12 | 14 |
| llave | 192 bits | 12 | 12 | 14 |
| | 256 bits | 14 | 14 | 14 |

### 2.3.3. Criptografía asimétrica

La criptografía asimétrica es por definición aquella que utiliza dos claves diferentes para cada usuario, una para cifrar que se le llama clave pública y otra para descifrar que es la clave privada. El nacimiento de la criptografía asimétrica se dio al estar buscando un modo más práctico de intercambiar las llaves simétricas. Diffie y Hellman [11], proponen una forma para hacer esto, sin embargo no fue hasta que el popular método de Rivest Shamir y Adleman

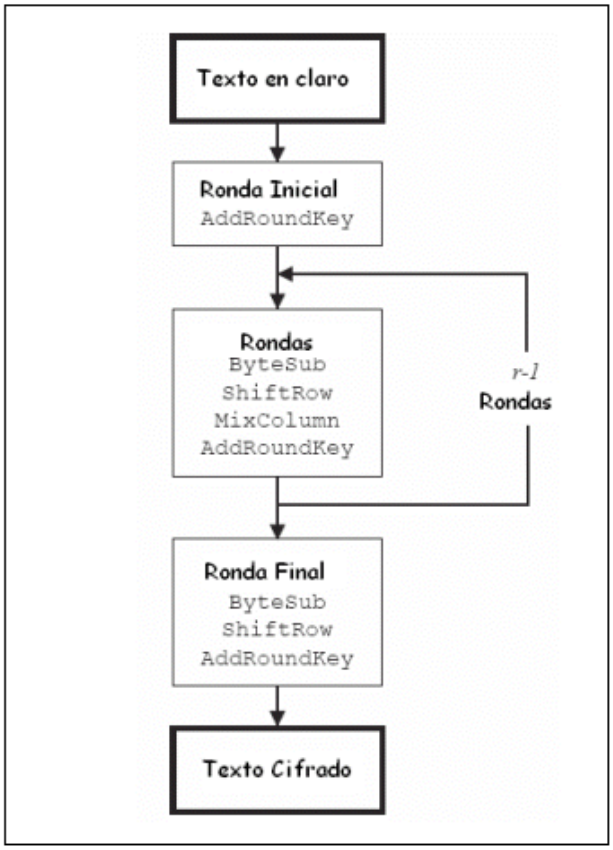


Figura 2.9: Esquema de cifrado de AES (r numero de rondas, ver tabla 2.1).

RSA publicado en 1978 [30], cuando toma forma la criptografía asimétrica, su funcionamiento esta basado en la imposibilidad computacional de factorizar números enteros grandes.

### RSA

El algoritmo RSA debe su nombre a las iniciales de sus tres inventores: Ronald Rivest, Adi Shamir y Leonard Adleman. Este sistema de cifrado basa su seguridad en la conjetura matemática que sostiene que el problema de factorizar números enteros en sus factores primos tiene una complejidad computacional

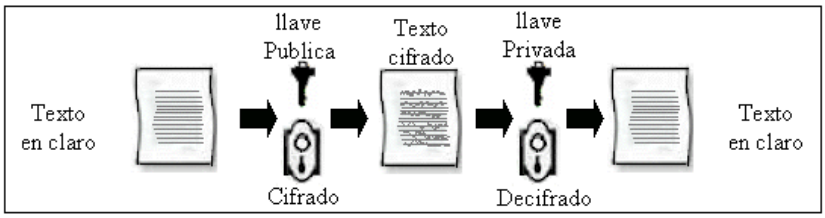


Figura 2.10: Esquema de cifrado asimétrico.

prohibitiva para el estado del arte de la tecnología de hoy en día y los tamaños en bits de los números utilizados. Las llaves pública y privad $(K_{pub}; K_{priv})$ se calculan a partir de un número que se obtiene como producto de dos primos grandes. El proceso para generar dicho par de llaves es el siguiente:

1. Se elige de manera aleatoria dos números primos grandes, $p$ y $q$
2. Se calcula el producto $n = pq$
3. Se elige ahora un número $e$ primo relativo con $(p-1)(q-1)$
4. La llave pública será $(e; n)$. Nótese que $e$ debe tener inversa módulo $(p-1)(q-1)$ para garantizar que existirá un número $d$,
   tal que $de \equiv 1 \mod ((p-1)(q-1))$ es decir, que $d$ es el inverso de $e$ mod $(p-1)(q-1)$
5. La llave privada será $(d; n)$

Por lo que:

$$K_{pub} = (e; n)$$

y

$$K_{priv} = (d; n)$$

Para poder garantizar un margen de seguridad aceptable en transacciones comerciales electrónicas, diferentes estándares recomiendan usar RSA con pares de llaves públicas y privadas con un tamaño no menor a 1024 bits [30].

### Criptografía de curvas elípticas (CCE)

Las curvas elípticas constituyen un formalismo matemático conocido y estudiado desde hace más de 150 años. Las primeras propuestas de uso de las curvas elípticas en la criptografía fueron hechas por Neal Koblitz y Victor Millar, de manera independiente, en 1985. La criptografía de curvas elípticas (CCE) fundamenta su seguridad en el alto grado de dificultad que supone resolver el problema del logaritmo discreto en el grupo abeliano formado por curvas elípticas definidas sobre campos finitos. De forma general, una curva elíptica $E(Fq)$ se define como el conjunto de puntos que satisface la ecuación:

$$E : y^2 = x^3 + ax + b$$

Donde $a$ y $b$ están en un campo finito apropiado $Fq$ de orden $q$, el cual puede ser el grupo de los números racionales, números complejos, enteros módulo $n$, campos de Galois, etc. Los coeficientes $a$ y $b$ caracterizan de manera unívoca cada curva. Se define también un punto en el infinito, denotado como $O$, a un punto imaginario situado por encima del eje de las abcisas a una distancia infinita, y que por lo tanto no tiene un valor concreto. Existe en el grupo la suma y una operación conocida como multiplicación escalar: Si $k$ es un entero y $P \in E(Fq)$ es un punto, entonces $kP$ es el punto obtenido al sumar $k$ copias de $P$. El elemento neutro es $O$. Las curvas elípticas definidas en un Campo de Galois $GF(P)$ siendo $P$ un número primo, forman un grupo donde todos los elementos, con excepción del cero, tienen inversa, por lo que se puede sumar, restar, multiplicar y dividir. Los puntos de estas curvas cumplen la ecuación:

$$y^2 = x^2 + ax + b \ (\mod P)$$

definiendo de esta forma el grupo $E(GF(P))$ [1]. En la figura 2.11 se muestran curvas elípticas definidas en el conjunto $R$, y en el campo $F(2^4)$.

Tómese un punto $G$ cualquiera de una curva elíptica $E$. Se denominará $(G)$ al conjunto $\{O, G, 2G, ..., \}$. En $E(GF(P))$ y $E(GF(2^m))$ los conjuntos de esta naturaleza deberán necesariamente ser finitos, ya que el número de puntos de la curva es finito. Por lo tanto, si se dispone de un punto $Q \in (G)$, debe existir un número entero $k$ tal que $kG = Q$. El problema de logaritmo discreto para las curvas elípticas consiste en hallar el número $k$ a partir de $G$ y $Q$. Debido a la enorme complejidad computacional que dicho problema matemático representa, es posible obtener con CCE niveles de seguridad similares a los proporcionados por otros sistemas de cifrado al precio de operaciones de campos finitos mucho menores a las requeridas por los otros esquemas. Las operaciones sobre campos finitos menores conducen al uso de llaves públicas y secretas también menores lo que a su vez tiene como resultado una mayor velocidad, y menores requerimientos de memoria y de poder de cómputo en las implementaciones de los algoritmos que conforman al esquema.

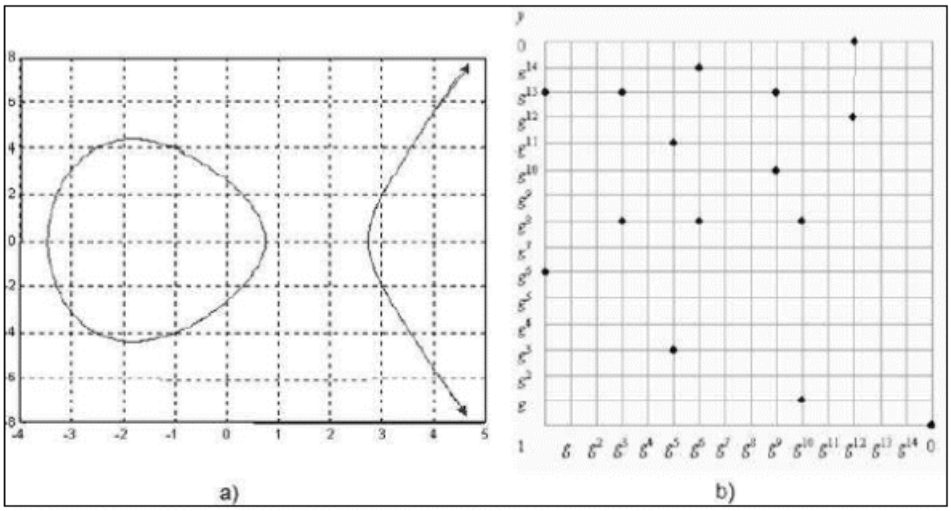


Figura 2.11: Gráficas de curvas elípticas : a) $y^2 = x^3 + 10x + 7$ sobre $R$ ; b) $y^2 + xy = x^3 + g^4x^2 + 1$ sobre $F(2^4)$.

### 2.3.4. Otros Algoritmos Asimétricos

#### Algoritmo de Diffie-Hellman

Es un algoritmo asimétrico, que se emplea fundamentalmente para acordar una llave comun (tambien conocida como llave de sesión) entre dos interlocutores, a través de un canal de comunicación inseguro. La ventaja de este sistema es que no son necesarias llaves publicas en el sentido estricto, sino una información compartida por los dos comunicacntes. Matemáticamente el algoritmo de cifrado de Diffie-Hellman se basa en las potencias de los números y en la funcion modular.

Sean $A$ y $B$ los interlocutores en cuestión. En primer lugar, se calcula un número primo $p$ y un generador $\alpha$ de $Z_p^*$, con $2 \leq \alpha \leq p-2$. esta información es pública y conocida por ambos. El algoritmo queda de la siguiente manera:

1. $A$ escoge un número aleatorio $x$, comprendido entre 1 y $p-2$ y envía a $B$ el valor $\alpha^x \pmod p$

2. $B$ escoge un número aleatorio $y$, análogamente al paso anterior, y envía a $A$ el valor $\alpha^y \pmod p$

3. $B$ recoge $\alpha^x$ y calcula $K = (\alpha^x)^y \pmod p$

4. $A$ recoge $\alpha^y$ y calcula $K = (\alpha^y)^x \pmod p$

Puesto que $x$ e $y$ no viajan por la red, al final $A$ y $B$ acaban compartiendo el valor de $K$, sin que nadie que capture los mensajes transmitidos pueda repetir el cálculo. Como se observa en la figura 2.12.

#### Acuerdo de llaves con CE

El problema del logaritmo discreto es la base para la seguridad de muchos sistemas criptográficos incluyendo al de curvas elípticas. La criptografía de curva elípticas utiliza al grupo de puntos definidos en una curva elíptica sobre

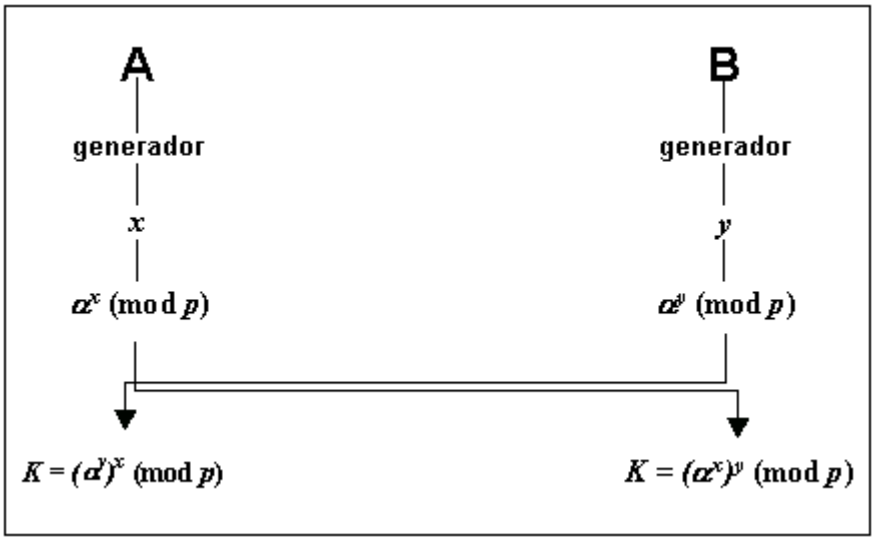


Figura 2.12: Diffie-Hellman.

un campo finito para obtener una variante del algoritmo para el acuerdo de llaves convencional Diffie-Hellman, ECDH [35].

La generación de llaves de sesión utilizando ECDH requiere ciertos parámetros de las curvas elípticas: la curva a ser utilizada, el punto generador $(P)$, el orden de $P$, $(n)$ y otros más.

Los interlocutores $A$ y $B$, generan cada uno, un entero aleatorio $r_a$ y $r_b$, dentro del intervalo [1, n-1], calculando la llave pública de ECDH $K_a = r_a P$ y $K_b = r_b P$. Esta llave pública es enviada a la contraparte. Entonces cada entidad calcula el punto $Z_b = r_b K_a$ y $Z_a = r_a K_b$. Los puntos $Z_a$ y $Z_b$ serán utilizados como la llave de sesión [1, 35]. Este proceso se muestra en la figura 2.13.


Figura 2.13: Diffie-Hellman para Curvas Elípticas.

# Capítulo 3

# Diseño e Implementación del Middleware

En este capítulo se describe la propuesta de solución al problema planteado en esta tesis. Es por ello que esta sección fue dividida en dos partes. La primera parte muestra el punto de partida hacia una propuesta, se describe dicha propuesta y también se muestra la arquitectura usada para resolver el problema planteado, así mismo, se presenta la interacción entre los elementos que componen a la arquitectura propuesta. La segunda parte se describe la implementación de cada uno de los módulos de que esta compuesta esta solución.

## 3.1. Propuesta

En una red área local inalámbrica (WLAN) cuyo medio de comunicación es la radio frecuencia (RF) se permite un ambiente de movilidad y flexibilidad, pero este ambiente es a su vez uno de sus puntos débiles en lo que a seguridad se refiere. Aunque dentro del estándar para una WLAN se establece el protocolo WEP cuya finalidad es brindar seguridad (simulando la seguridad

física que poseen las redes de área local cableadas), se ha demostrado que este protocolo no cumple en la mayoría de veces con su propósito[**?**]. Por otro lado los protocolos de comunicación soportados en una WLAN, están basados en el modelo de referencia OSI (Open System Interconnection, por sus siglas en ingles).

El modelo de referencia OSI divide en siete capas el proceso de transmisión de datos entre computadoras, donde cada capa se encarga de ejecutar una determinada parte del proceso global. Este marco de trabajo estructurado en capas, aun siendo puramente conceptual, puede utilizarse para describir y explicar el conjunto de protocolos reales.

- **Capa física:** Se encarga de establecer el medio físico por el cual se transmiten los datos (cables, radio frecuencia, etc.).
- **Capa enlace de datos:** Se encarga de establecer la manera en que las computadoras envían y reciben los datos a través del soporte físico proporcionado en la capa anterior.
- **Capa de red:** Se encarga de definir la forma en que un mensaje se transmite a través de distintos tipos de redes hasta llegar a su destino.
- **Capa de transporte:** Esta capa se encarga de controlar el flujo de datos entre los nodos que establecen una comunicación; los datos no sólo deben entregarse sin errores, sino además en la secuencia que proceda.
- **Capa de sesión:** Se encarga de establecer el enlace de comunicación o sesión entre las computadoras emisora y receptora.
- **Capa de presentación:** Se encarga de traducir los paquetes (la creación del paquete para la transmisión de los datos por la red empieza en realidad en la capa de aplicación) de la capa de aplicación y los convierte a un formato genérico que pueden leer todas las computadoras y viceversa.

- **Capa de aplicación:** Se encarga de proporcionar la interfaz y servicios que soportan las aplicaciones de usuario. También se encarga de ofrecer acceso general a la red.

Uno de los protocolos mas usados en las comunicaciones en redes es el protocolo TCP/IP, que prácticamente es el estándar de-facto para la conexión en redes. Las redes TCP/IP son ampliamente escalables, por lo que TCP/IP puede utilizarse tanto para redes grandes y pequeñas. TCP/IP es un conjunto de protocolos que pueden ejecutarse en distintas plataformas de software (Windows, UNIX, etc.) y casi todos los sistemas operativos de red lo soportan como protocolo predeterminado. Además consta de una serie de protocolos que componen la pila TCP/IP. Puesto que el conjunto de protocolos TCP/IP se desarrollo al mismo tiempo que el modelo de referencia OSI [3], los protocolos que lo conforman no se corresponden perfectamente con las distintas capas del modelo de referencia OSI como se muestra en la figura 3.1.

De acuerdo a la figura 3.1 se observa que TCP/IP engloba pero no implementa en su capa de aplicación a las capas de sesión y presentación del modelo de referencia OSI. En este trabajo se propone una capa intermedia (a la cual se le denomina Middleware) entre la capa de transporte y de aplicación de TCP/IP, la cual se encarga de proveer seguridad en la transmisión de datos, además de trabajar como una capa de sesión y presentación de acuerdo al modelo de referencia OSI. El nuevo esquema se observa en la figura 3.2.

### 3.1.1. Arquitectura del Middleware

En esta sección se presentan los componentes del middleware que hacen que se pueda proveer a la aplicación una seguridad ajustable y flexible.

En la figura 3.3 se observa que el middleware consta de varios módulos que a su vez forman dos bloques principales. El primer bloque se encarga de establecer los parámetros necesarios para obtener una comunicación segura a la que se le denomina ***Negociación***, con la cual se establece los niveles de

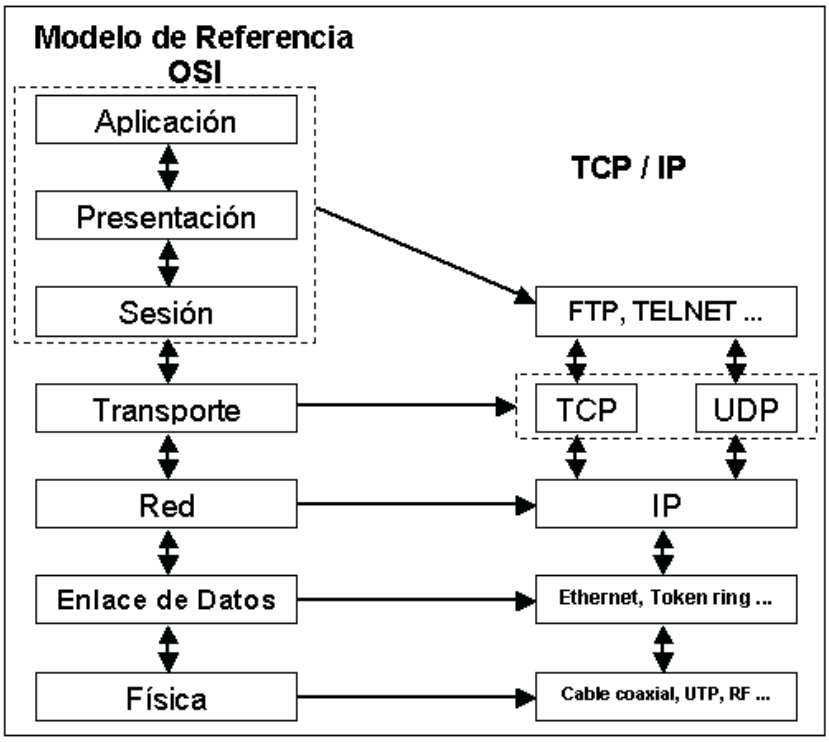


Figura 3.1: Modelo OSI vs TCP/IP.

seguridad de los que dispone la aplicación en ese momento. El segundo bloque se encarga de la comunicación, al que se le llama ***Sockets Seguros***, y permite garantizar que los datos viajen en la red de manera segura proporcionando confidencialidad en la comunicación. A continuación se describe la función que realiza cada modulo dentro del middleware:

- **Negociación:** Este modulo provee tres servicios de seguridad: confidencialidad, autenticación e integridad. Es por ello el encargado de autenticar entre un servidor y un cliente, una vez que la autenticación es correcta se lleva a cabo el proceso de negociación en el cual el cliente y el servidor establecen los parámetros de seguridad requeridos para poder establecer una comunicación segura entre ese cliente y ese servidor, por

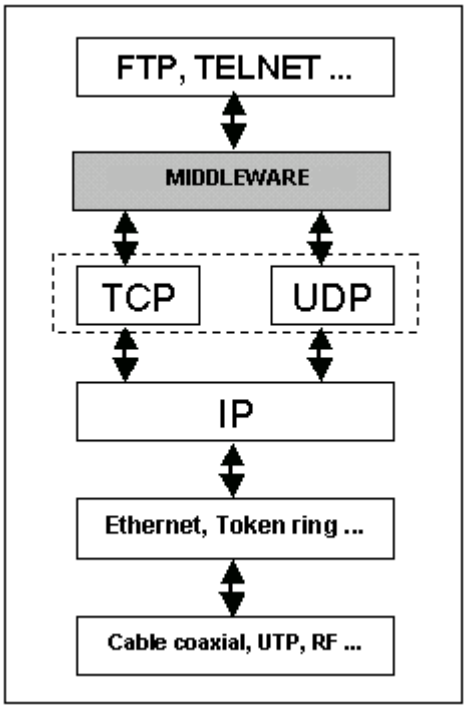


Figura 3.2: Esquema propuesto.

ultimo realiza el intercambio de la llave de sesión que permitirá que los mensajes entre el cliente y el servidor sean cifrados obteniendo confidencialidad de los datos.

- **Seguridad:** Este módulo provee confidencialidad de los datos a el middleware. Por lo tanto es el encargado de cifrar los datos usando algoritmos simétricos de bloques, para cifrar se usa la llave de sesión que se estableció para cada par cliente-servidor en el modulo de negociación.

- **Sockets:** Este módulo se encarga de proveer las herramientas necesarias para que la aplicación desarrollada interactué con el middleware y a su vez el middleware interactué con la capa de transporte del protocolo TCP/IP. Permitiendo con ello que la comunicación cliente-servidor

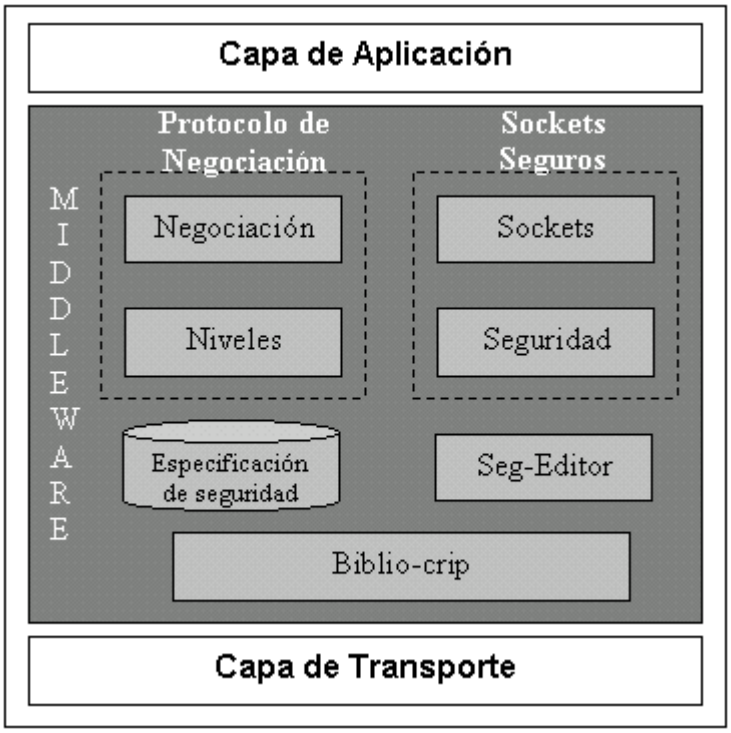


Figura 3.3: Arquitectura del middleware.

entre dos aplicaciones sea transparente y segura.

- **Biblio-crip:** Este modulo representa a una biblioteca de criptografía la cual contiene todas las herramientas criptográficas (algoritmos simétricos, asimétricos, entre otros) necesarias para generar la capa de seguridad de los datos dentro del middleware.

- **Niveles:** Este modulo es el encargado de administrar los niveles de seguridad determinados en el modulo de negociación o por el modulo Seg-editor. Es decir establece que nivel y que tipo de seguridad se va a efectuar en cada momento de la comunicación.

- **Seg-editor:** Este modulo es el único de todos los módulos que com-

ponen al middleware que opera fuera de línea. Es el encargado de establecer las preferencias de seguridad de un usuario a una aplicación especifica para que en el momento de la negociación la aplicación proponga estos niveles como estándares de seguridad en la comunicación.

### 3.1.2. Interacción entre módulos

Después de conocer la arquitectura del middleware en esta sección se presenta la interacción entre los elementos que forman parte del middleware. En la figura 3.4 se observa la interacción que existe entre los diferentes módulos que integran al middleware. La interacción esta dividida en dos partes fundamentales: la que se ejecuta en línea y la que se ejecuta fuera de línea.

En la parte fuera de línea se ejecuta el modulo **Seg-editor**, el cual es un editor proporcionado para que el usuario de una aplicación pueda determinar los niveles de seguridad que requiere para establecer su comunicación. **Seg-editor** trabaja de manera directa con un archivo de configuración de niveles de seguridad, llamado especificación de seguridad. En este archivo se almacenan los niveles disitintos de seguridad que permiten que la parte del middleware que opera en modo cliente pueda obtener una configuración de acuerdo a su usuario y pueda negociar estos niveles de seguridad con la parte que actué como servidor.

Por otro lado, en la parte en línea existe una interrelación entre varios módulos, la capa de aplicación y la capa de transporte de TCP/IP. La aplicación tiene interacción primeramente con el modulo de negociación que se encarga de autenticar e inmediatamente convenir los niveles de seguridad para la aplicación. Una vez que se lleva acabo la negociación, este módulo registra las preferencias de seguridad al módulo niveles. La aplicación interactúa con el módulo de sockets después de haber terminado la fase de negociación. La interacción recae en que la aplicación le pasa los datos al módulo sockets para su manejo, este a su vez pasa los datos al módulo seguridad donde los datos son recibidos e inmediatamente se le solícita al módulo niveles los

parámetros (algoritmo de cifrado, llave de sesión) con los cuales serán cifrados los datos. Cuando el módulo seguridad ya tiene dichos parámetros este envía los datos al módulo biblio-crip en el cual son cifrados los datos de acuerdo a los parámetros y posteriormente los datos son regresados al módulo seguridad. Este a su vez los envía al módulo sockets, y de ahi los envía a la capa de transporte de TCP/IP. Para el proceso invertido la capa de transporte de TCP/IP pasa los datos a los sockets donde los transfiere a la capa de seguridad obtiene los datos y consulta al módulo niveles los parámetros de seguridad. Después son pasados al módulo biblio-crip para ser descifrados y por último ser pasados a la aplicación a través del módulo sockets.

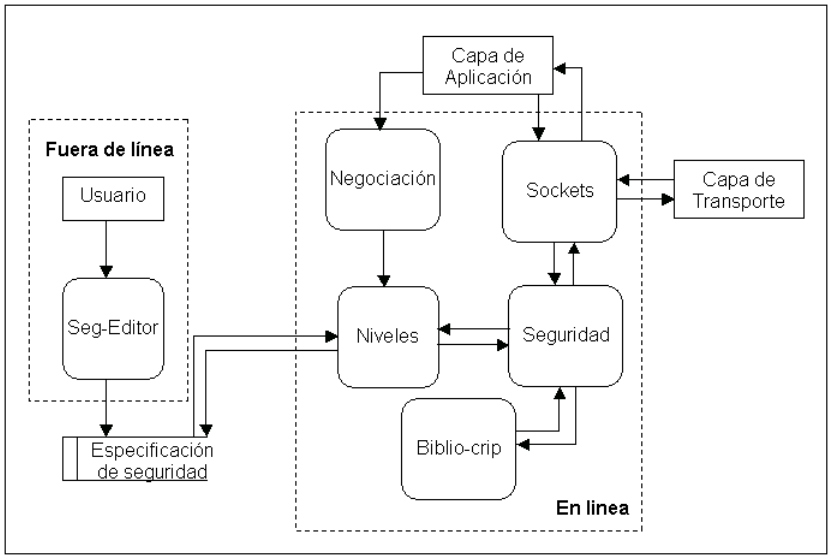


Figura 3.4: Esquema de interacción.

## 3.2. Implementación

### 3.2.1. Negociación

Esta parte fue implementada retomando el esquema del protocolo de Negociación de WTLS [33] del protocolo WAP (Wireless Aplication Protocol). En el protocolo de Negociación se acuerdan los parámetros criptográficos para establecer o reiniciar una conexión segura entre un cliente y un servidor. Cuando un cliente y un servidor inician una comunicación, ellos deciden que versión del protocolo usarán, seleccionan los algoritmos criptográficos, y utilizan técnicas de criptografía de llave publica para autenticarse mutuamente y generar, finalmente, la llave de sesión compartida [33]. Esta llave de sesión secreta será posteriormente utilizada por la parte de seguridad del middleware para cifrar la comunicación.

Dependiendo de las opciones tomadas en el protocolo de negociación se pueden distinguir tres clases de implementaciones de WTLS definidas en su especificación [17], éstas son:

- **Clase 1:** Únicamente brinda privacidad e integridad de datos mediante un intercambio de llaves anónimo sin autenticación.

- **Clase 2:** Brinda privacidad e integridad de datos además de autenticación a nivel del servidor. Aquí, la autenticación del servidor se basa en certificados digitales. La llave del servidor puede ser anónima o autenticada, la llave del cliente es anónima.

- **Clase 3:** Brinda privacidad e integridad de datos además de autenticación tanto del servidor como del cliente. Aquí, la autenticación del servidor y el cliente se basa en certificados digitales. Tanto la llave del cliente como del servidor puede ser anónima o autenticada.

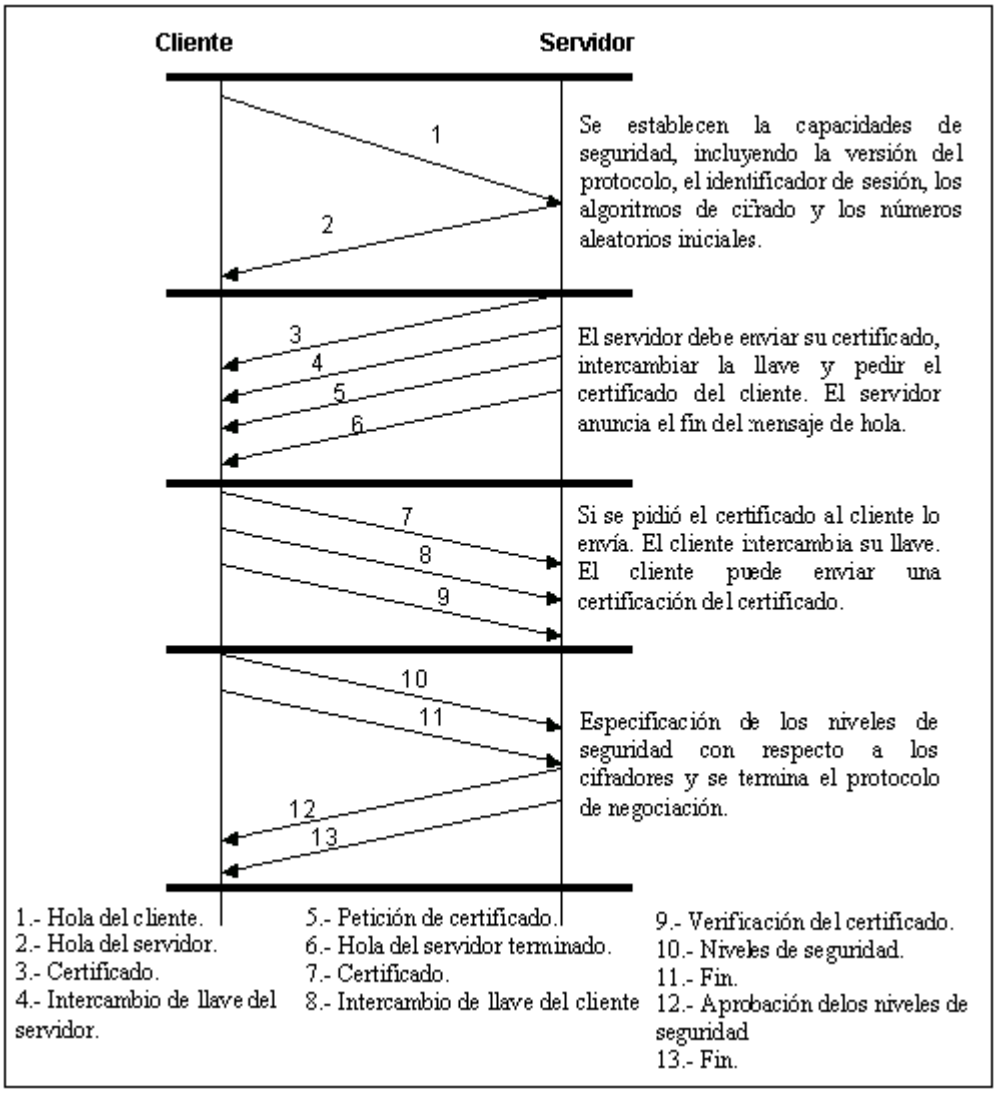


Figura 3.5: Esquema de la Negociación completa.

En la figura 3.5, se observan los pasos que se deben de seguir para establecer una comunicación segura, con el protocolo WTLS clase 3. El protocolo de negociación admite el uso de únicamente dos criptosistemas de llave pública: RSA y Criptosistemas de Curvas Elípticas (ECC). Los algoritmos para el

intercambio de llaves de sesión permitidos por el estándar WTLS incluyen ECDH, RSA y Diffie-Hellman. En particular para esta implementación se usa como criptosistema de llave publica a ECC y para el intercambio de llaves de sesión se escogió ECDH.

### 3.2.2. Niveles

El módulo de niveles fue implementado utilizando una matriz dinámica en la cual se van almacenando los identificadores de los algoritmos de cifrado simétrico que a través de las preferencias de seguridad del usuario fueron establecidas o en su caso fueron establecidas por el modulo de negociación. Cuenta con un mecanismo que hace que los algoritmos sean identificados por el módulo Biblio-crip de tal manera que no existan confusiones entre los módulos sobre los algoritmos. Otro aspecto importante es que la matriz crece de manera dinámica conforme se van requiriendo mas niveles de seguridad para una aplicación. Por default se manejan cuatro niveles que son:

- **Nivel 0:** Sin cifrar.
- **Nivel 1:** DES.
- **Nivel 2:** TDES.
- **Nivel 4:** AES.

Cabe señalar que este orden puede ser o no ser alterado por el proceso de negociación.

### 3.2.3. Seg-editor

Este módulo fue implementado como una aplicación con entorno gráfico. El entorno gráfico fue desarrollado utilizando GTK para linux. Esta aplicación funciona como una interfaz entre el usuario y el middleware en el cual el

usuario puede establecer sus preferencias de seguridad para una aplicación especifica. Una vez que el usuario definió su perfil de seguridad, la aplicación procede a guardar ese perfil en un archivo de configuración; el cual es el puente de enlace entre Seg-editor y el proceso niveles encargado de la administración de los perfiles de seguridad para cada una de las aplicaciones del usuario.

### 3.2.4. Sockets

El módulo de sockets fue implementado usando un empaquetamiento (wrapping) de los sockets de la librería <sockets.h> del lenguaje C de linux (GCC), usada en el desarrollo de aplicaciones cliente/servidor.

Es decir, a los sockets (TCP y UDP) se les creó una interfaz la cual es transparente para la capa aplicación. Bajo esa interfaz se llevan a cabo dos procesos: el primero que consiste en establecer un mecanismo de envío (ver figura 3.6 ) y el segundo en establecer un mecanismo de recepción (ver figura 3.7) en el cual los paquetes de datos posean el atributo de confidencialidad durante su paso por la WLAN; brindando con ello al usuario una herramienta con la cual incrementara los niveles de seguridad en sus aplicaciones de tipo cliente/servidor.

La forma en que los sockets procesan el envío y recepción de datos es la siguiente: en la fase de envío, la aplicación transfiere los datos a la interfaz de sockets y una vez ahí se verifica el nivel de seguridad con el que se desea cifrar. En caso de que exista un nivel de cifrado los datos serán transferidos al módulo de seguridad donde se lleva a cabo el proceso de cifrado. En el caso contrario, no son cifrados los datos (los datos se envían en un nivel 0). Posteriormente, si los datos fueron cifrados son regresados al módulo sockets. También es regresado un nuevo tamaño de datos al cual se le llama ***lenfull***. Este parámetro es necesario puesto que al ser cifrados los datos, el tamaño de los mismos tiende a aumentar. Para el envió de los datos se establece un tamaño máximo de envió denominado ***MAXLENPAQ***. Si el tamaño del bloque de datos excede a ***MAXLENPAQ*** es necesario partir

ese bloque en la cantidad necesaria de subpaquetes ***npack***. Una vez que se tienen todos los parámetros necesarios se procede al armado del encabezado del paquete de datos en el cual se describen las características del paquete (tamaño de datos, numero de paquetes, nivel de cifrado, entre otros) y las cuales permiten interpretar en el lado del receptor los datos. Por último, se envía el encabezado y posterior se procede al envío de los subpaquetes. En la fase de recepción, los sockets reciben primeramente el encabezado, con el cual proceden a esperar los ***npack*** subpaquetes, desensamblándolos y formando así un bloque de datos toda vez que ha llegado el último subpaquete. Cuando ya se tiene el bloque de datos en caso de que tengan algún nivel de cifrado este bloque de datos es transferido el módulo de seguridad en el cual se lleva acabo el proceso de descifrado y los datos en claro recuperados son regresados al modulo sockets que a su vez los envía a la capa de aplicación para finalizar el proceso de recepción. Los dos procesos antes descritos son realizados de manera transparente para las aplicaciones.

Para el proceso de envió y recepción descritos anteriormente se implemento una estructura de paquete de mensaje (véase la figura 3.8) que nos permite establecer un control sobre los paquetes a enviar. El tamaño total del paquete es de 1024 bits. A continuación se describen cada uno de los campos de la estructura del paquete:

- **Paquete:** este campo indica el número de paquete, el cual es utilizado cuando el paquete original excede el tamaño máximo ***MAXLENPAQ***, de tal manera que ha sido particionado en dos o más subpaquetes para completar su envío. Su tamaño es de 8 bits.

- **Tamaño total de datos:** este campo indica el tamaño de datos que se van a enviar. Es especialmente útil en caso de que la cantidad de información a transmitir supere el tamaño máximo permitido en el área de datos. Su tamaño es de 16 bits.

- **Nivel de seguridad:** este campo indica el nivel de cifrado que fue

utilizado en los datos originales. Sirve para que en la parte receptora del paquete sea capaz de conocer qué algoritmo y nivel de cifrado ha sido utilizado en los datos. Su tamaño es de 8 bits.

- **Tamaño total de datos en el paquete:** este campo indica la cantidad de datos que van en el área de datos cifrados. Esto es en caso de que la cantidad de datos sea menor que el área de datos cifrados. Su tamaño es de 16 bits.

- **Datos cifrados:** este campo es donde se colocan los datos cifrados para su envío. Su tamaño es de 976 bits.

### 3.2.5. Seguridad

Este módulo fue implementado para administrar los mecanismos de seguridad dentro del middleware. Consiste en verificar, el tipo de operación(cifrado ó descifrado) a realizarse. Si es una operación de cifrado se tiene que obtener del modulo ***Niveles*** , el algoritmo de cifrado que será usado para este paquete de datos (esto es debido a que los algoritmos son asignados de manera dinámica). Una vez que se conoce el algoritmo se procede a recuperar la llave de sesión ***Skey***. Por ultimo, los datos, la llave de sesión y la especificación del algoritmo de cifrado son pasados al módulo ***Biblio-crip*** donde se lleva acabo el proceso de cifrado. Cuando ya han sido cifrados los datos son regresados al modulo ***Seguridad*** junto con el nivel de cifrado, y a su vez al modulo ***Sockets***. Para el proceso de descifrado se obtiene los datos cifrados y el nivel de cifrado con el cual se sabe el algoritmo de cifrado con el cual llegaron los datos. Los datos cifrados, la llave de sesión ***Skey*** y el algoritmo de cifrado son pasados al modulo ***Biblio-crip*** para llevar acabo el proceso de descifrado. Por ultimo, los datos en claro son regresados al módulo ***Sockets***. Este proceso se puede observar en la figura 3.9.

### 3.2.6. Biblio-crip

Para la implementacion de la biblioteca criptográfica Biblio-crip se utilizo el trabajo realizado en la universidad de Oregon State, donde se desarrolló un conjunto de herramientas criptográficas, englobadas en una biblioteca conocida como RCT. Entre los algoritmos incluidos en la biblioteca se encuentran implementaciones de criptografía de llave pública con RSA y Curvas Elípticas, AES, TDES y DES para la criptografía de llave simétrica y la familia SHA para las funciones hash [25]. Cabe señalar que para este trabajo se realizó una migración de este conjunto de herramientas a la plataforma de linux ya que dicha herramienta se encontraba desarrollada para la plataforma Windows. Asi mismo, fue necesaria la migración también a la plataforma embebida de linux distribución familiar con ambiente GPE (Gui Palmtop Environment) [13].

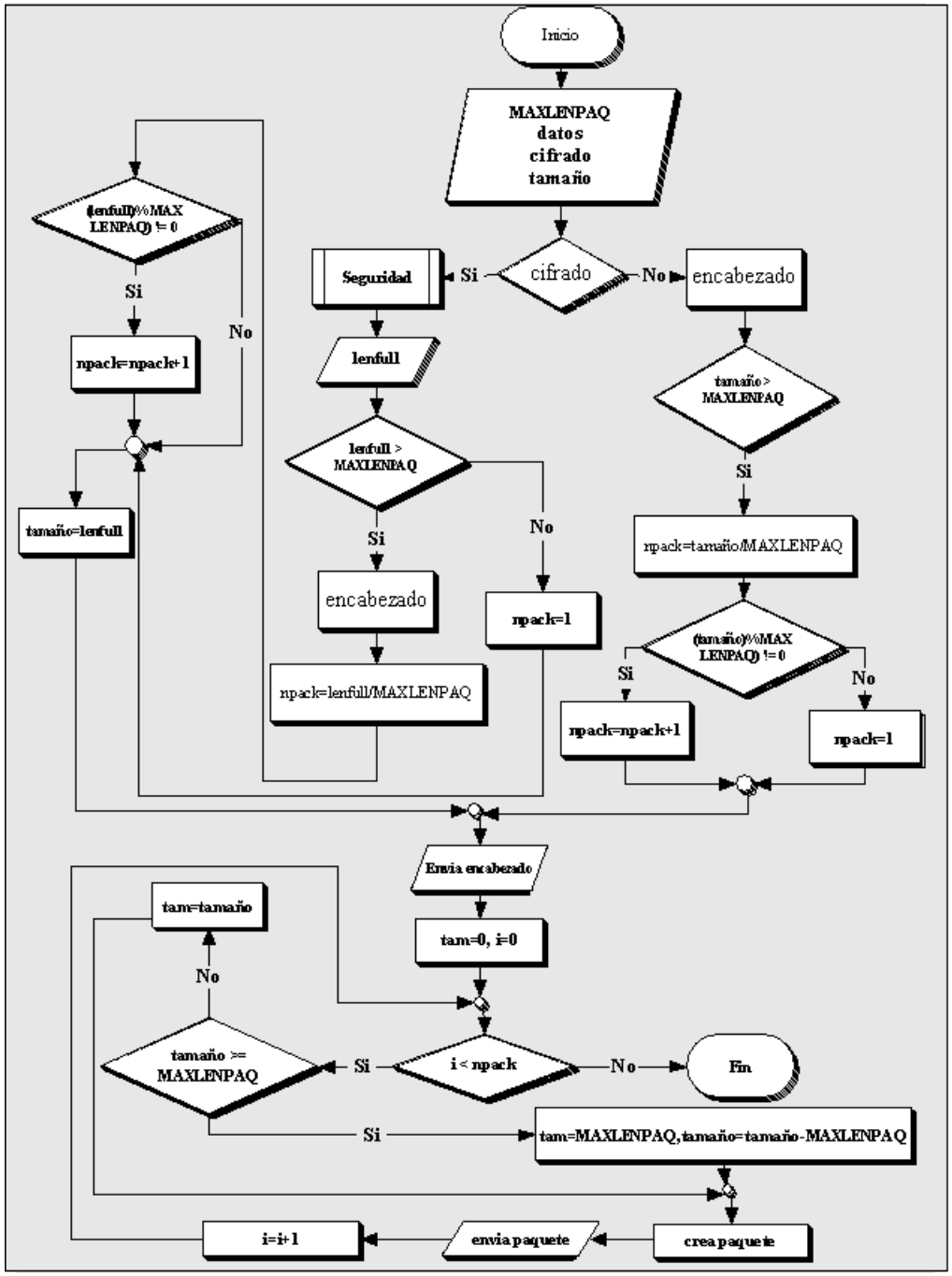


Figura 3.6: Diagrama de Envío.

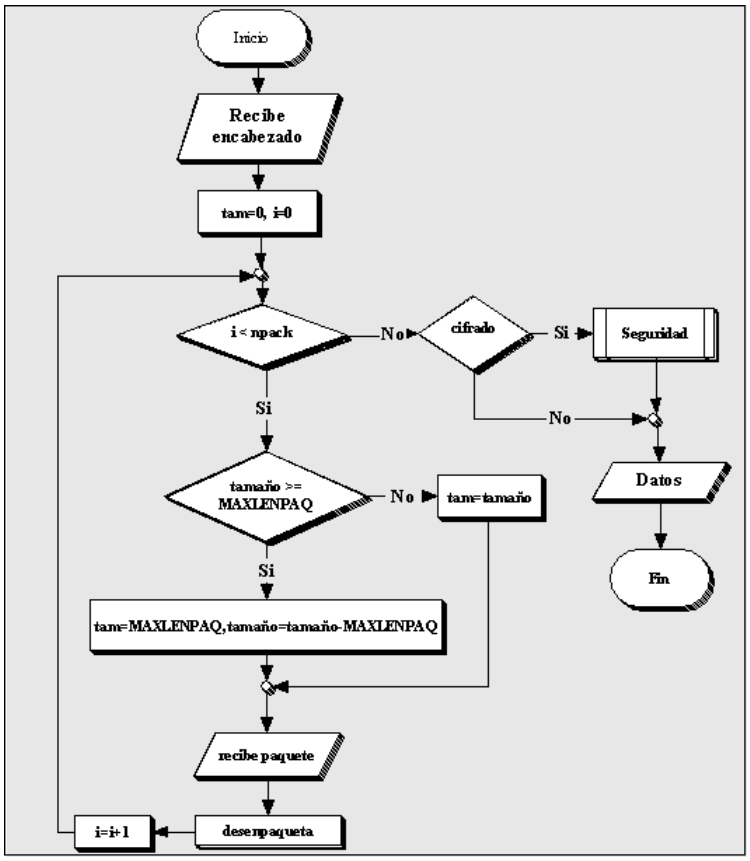


Figura 3.7: Diagrama de Recepción.

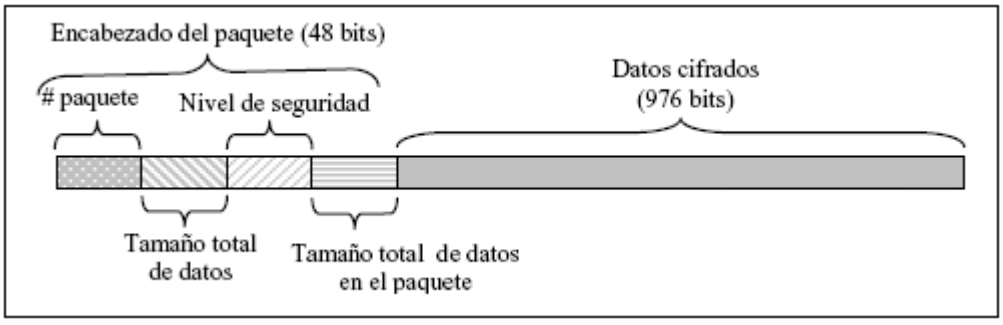


Figura 3.8: Esquema del paquete.

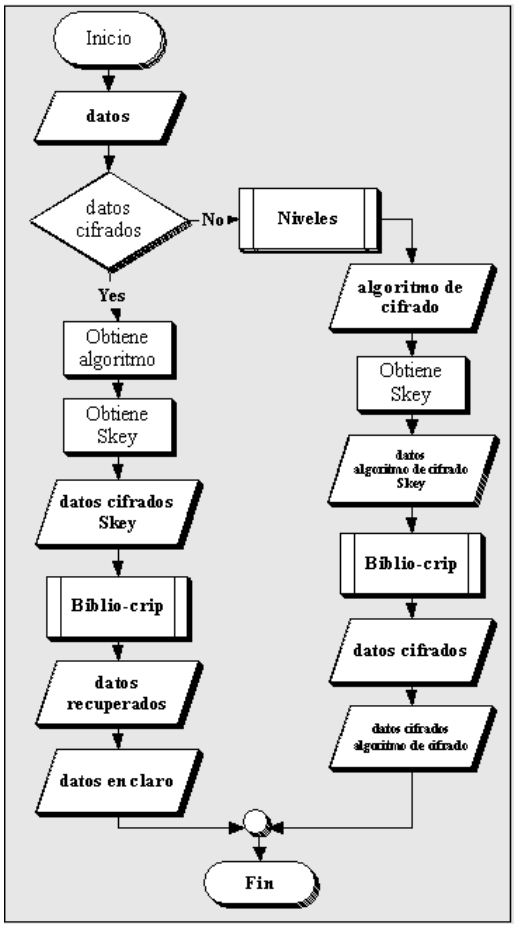


Figura 3.9: Diagrama del modulo Seguridad.

# Capítulo 4

# Resultados

En este capítulo, se presenta la parte experimental de esta tesis, la cual consiste en probar el middleware propuesto en un sistema de video en tiempo real sobre una red de área local inalámbrica.

Primero se describen los elementos que forman parte de la plataforma de prueba y las métricas que se utilizan para medir los resultados. Luego entonces se muestran y se evalúan los resultados obtenidos.

## 4.1. Plataforma de prueba

Consiste en un sistema multimedia distribuido (Cliente-Servidor) de transmisión de video en tiempo real sobre una red inalámbrica. El programa servidor consiste en un proceso encargado de obtener los frames de video de la cámara Web, además de colocarlos para que se encuentren disponibles para cualquier cliente que lo solicite. El programa servidor se ejecuta sobre una plataforma de tiempo real conocida como RTAI(Real-Time Application Interface)[28]. El proceso servidor escucha por conexiones de clientes en un puerto predeterminado al cual los clientes solicitaran su conexión. Cuando existe una solicitud de conexión de parte de un cliente, el servidor atiende

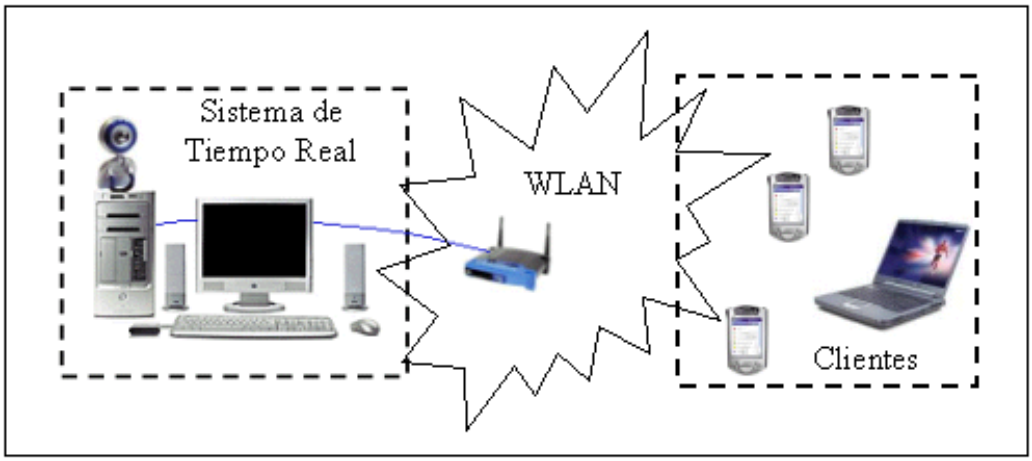


Figura 4.1: Configuración de la plataforma experimental.

esta solicitud creando un nuevo proceso que será el encargado de atender al cliente que realizo la solicitud, enviándole los frames de video. Esta operación se repite cada vez que exista una solicitud de conexión de un cliente al servidor. Para la implementacion de este programa se desarrolló una interfaz entre la aplicación y el driver de la cámara Web, lo que permitió que la obtención de los frames fuera de manera transparente. Por otro lado, el programa cliente consiste en una aplicación de reproducción de video. En dicha aplicación se tienen dos versiones: una para clientes que se ejecutan sobre plataforma PDA y otra para clientes que se ejecutan sobre plataformas Laptop ó Desktop. Nuestro middleware fue utilizado para el desarrollo de las comunicaciones tanto del programa cliente como del programa servidor. El uso del middleware es simple y se utiliza del mismo modo que los sockets proporcionados por el sistema operativo. Esto hace que el desarrollo de sistemas cuya arquitectura cumpla con el modelo Cliente-Servidor sea fácil y sencilla. La arquitectura del sistema desarrollado se muestra en la figura 4.1.

### 4.1.1. Características de la plataforma de prueba

En la figura 4.1, se muestra la configuración de nuestra plataforma de prueba. Esta plataforma está dividida en tres partes y cada una de ellas tiene características especiales de hardware y software. La primera parte, la cual se refiere al servidor de video en tiempo real tiene las siguientes características:

- Dell PRECISION 360, Pentium IV, 2.8 GHz, 512 MB en RAM.
- Sistema operativo Linux, distribución Red-Hat 9.0, kernel 2.4.18, sobre el cual se ejecuta el módulo de tiempo real RTAI.
- Webcam creative Go Plus, con una resolución de 320x240 pixeles y 25 frames/segs.
- Proceso servidor de video, el cual fue desarrollado en lenguaje C y utiliza el middleware. Consiste en un programa que provee el servicio de video en tiempo real a los clientes.

En la segunda parte, se encuentra la red de área local inalámbrica cuyas características son las siguientes:

- Estándar IEEE 802.11b/g
- Taza de transferencia de datos de 11 Mb/seg a 54 Mb/seg.
- Modo infraestructura.
- WEP implementado.
- Access point 3com Office conect.

En la tercera y ultima parte, se encuentran los clientes, que son aplicaciones que despliegan el video proveído por el servidor. Estos clientes se ejecutan sobre diferentes plataformas, en la tabla 4.1 se muestran estas plataformas usadas por los clientes.

## 4.2. Pruebas realizadas

Las pruebas realizadas a este sistema están basadas en el tiempo de sobre carga que implica brindar seguridad a cualquier aplicación, lo cual repercute de manera directa en la QoS de la aplicación. Cabe mencionar que para estas pruebas en el análisis de resultados no se toman en cuenta el ancho de banda, la latencia y la QoS de la red de área local inalámbrica, puesto que no estaba dentro del objetivo del marco de referencia de esta tesis.

En particular estos son los puntos que observamos:

- Latencia
  - Fase de negociación.
  - Cifrado del servidor.
  - Descifrado de los clientes.
- Utilización de memoria del dispositivo.
- Degradación de la QoS de la aplicación cliente.

La manera de medir los puntos anteriores están dadas en función de las condiciones presentes en cada uno de los casos en particular. Para la latencia en la

Tabla 4.1: Plataformas de las aplicaciones clientes.

| **Nombre** | **Tipo** | **Sistema Operativo** | **CPU** | **RAM** |
|---|---|---|---|---|
| PDA | Ipaq h3950 | Linux familiar GPE | Intel Xscale 400MHz | 64 MB |
| Laptop | Dell Inspiron 600M | Linux fedora core 1 | Intel centrino 1.4 GHz | 256 MB |
| Desktop | HP Vectra VE | Linux Red-Hat 9.0 | Intel Pentium III 500 MHz | 128 MB |

fase de negociación se toma una muestra de tiempos para cada par Cliente-Servidor (Servidor-PDA, Servidor-Laptop y Servidor-Desktop) puesto que al existir una interacción entre ambos, la fase de negociación depende directamente de las características de cada una de las maquinas involucradas. Para medir el cifrado del Servidor se realizan muestras de tiempo para cada uno de los niveles de seguridad establecidos (DES, TDES, AES). Para medir el descifrado de los clientes se realiza de igual manera que para el caso anterior, es decir, se toman muestras de tiempo en cada uno de los niveles establecidos. La utilización de la memoria del dispositivo se observa al mostrar el tamaño del middleware y el tamaño de cada una de la aplicaciones desarrolladas. Para el caso de la degradación de la QoS de la aplicación se analiza la latencia de los procesos anteriores y se ve su repercusión en el desempeño de la aplicación.

## 4.3. Resultados obtenidos

### 4.3.1. Fase de Negociación

El proceso de negociación entre los clientes (PDA, Desktop y Laptop) y el servidor suele ser una operación costosa, puesto que en la interacción entre ambos dispositivos se involucran factores (poder de cómputo, memoria, entre otros) que afectan el rendimiento de este proceso, con lo cual el tiempo de cómputo entre una plataforma y otra podría variar. Es por ello que la manera en que se evalúa el desempeño de nuestro middleware en la fase de negociación dentro del sistema consistió en tomar muestras del tiempo de cómputo del proceso en las ejecuciones en las diferentes pares de plataformas, los resultados obtenidos se muestran en la tabla 4.2.

Los tiempos de cómputo que se muestran en la tabla 4.2 corresponden al proceso de negociación clase 3, de acuerdo a lo discutido en el capítulo anterior, en el cual se estableció que en la implementación del middleware

era necesario un proceso de negociación completo donde se autentique tanto al cliente como al servidor.

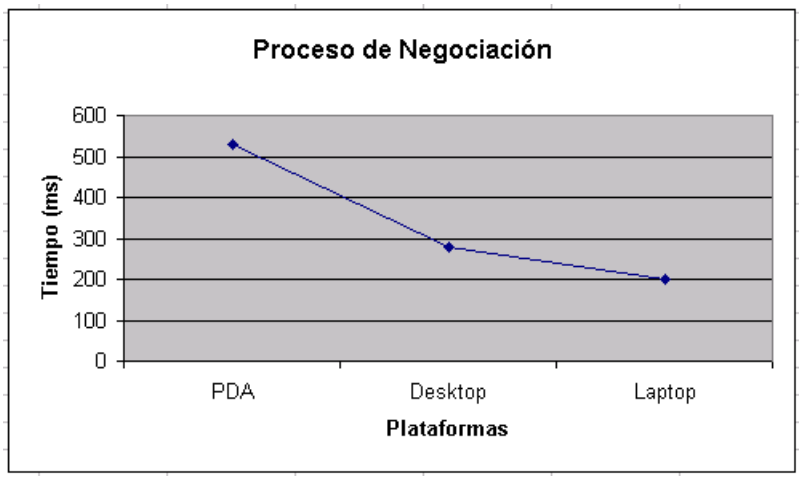


Figura 4.2: Grafica de tiempos de cómputo para la fase de negociación

En la figura 4.2 se observa que el peor caso en la fase de negociación es cuando el cliente se ejecuta sobre la PDA. Esto es de esperarse puesto que la PDA es la plataforma que mas limitantes de recursos posee, como se muestra en la tabla 4.1. El mejor caso se presenta cuando el cliente se ejecuta sobre Laptop.

### 4.3.2. Cifrado del servidor

El proceso de cifrado de los datos en el sistema de prueba se encuentra del lado del servidor. Esto se realiza asi debido a que el servidor obtiene cada uno de los frames de la cámara Web y posteriormente se realiza el cifrado

Tabla 4.2: Fase de negociación.

| | **PDA** | **Desktop** | **Laptop** |
|---|---|---|---|
| Server | 528 ms | 277 ms | 200 ms |

de los datos de acuerdo a los niveles de seguridad preestablecidos que fueron descritos en el capítulo anterior. Los factores que influyen en el cifrado de los mismos es primordialmente el tamaño de bloque de datos a cifrar( para esta aplicación de prueba existen dos tamaños de bloques de datos). Puesto que las limitaciones que tiene una PDA la hacen diferente de los otros dispositivos los tamaños son los siguientes:

- 230400 bytes es el tamaño de los frames que utilizan clientes Laptop y Desktop.
- 108000 bytes es el tamaño de los frames en clientes PDA.

De acuerdo a los parámetros anteriores se obtuvieron los resultados que se muestran en la tabla 4.3.

Tabla 4.3: Cifrado de datos.

| **Niveles** | **PDA** | **DESKTOP** | **LAPTOP** |
|---|---|---|---|
| Nivel 1 | 11 ms | 24 ms | 22 ms |
| Nivel 2 | 34 ms | 71 ms | 69 ms |
| Nivel 3 | 2 ms | 5 ms | 4 ms |

Como se observa en la figura 4.3 el cifrado mas eficiente es el que se obtiene del nivel 3 (AES). En lo que a seguridad se refiere también es el que nos brinda el mejor incremento a los niveles de seguridad de la aplicación. El algoritmo que muestra el peor desempeño es el TDES, este resultado es previsible puesto que este algoritmo como se mostró en el capítulo 2, no es mas que una triple interacción del algoritmo DES, por lo cual el tiempo de computo de este algoritmo resulta el peor aunque no por ello sea el que mejor nivel de seguridad proporciona.

Por otro lado, también se observa que el cifrado que realiza el servidor con respecto a la plataforma PDA, resulta en el mas rápido, esto se debe a que la cantidad de datos cifrados es menor a la cantidad de datos que se

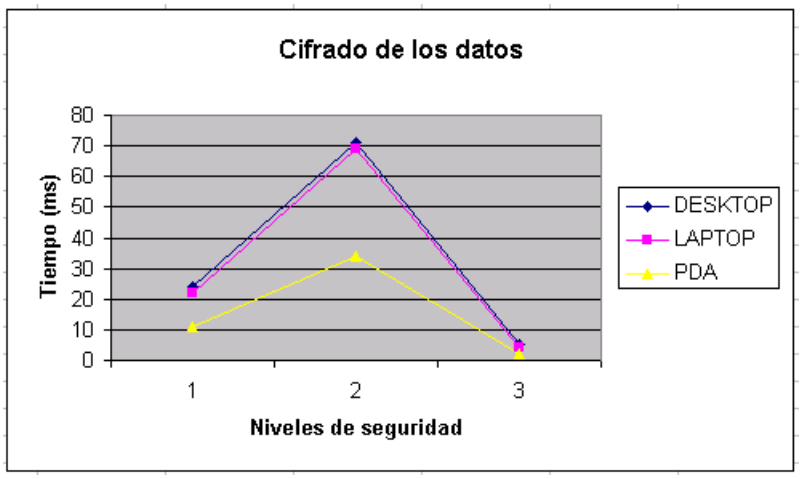


Figura 4.3: Grafica de tiempos de cómputo para la fase de cifrado.

cifran para las otras plataformas (108000 bytes en PDA y 230400 bytes para Laptop y Desktop).

### 4.3.3. Descifrado de los clientes

El descifrado de los datos para esta aplicación de prueba se realiza del lado del cliente. Para esta parte se tomaron muestras del tiempo de cómputo que resulta del proceso de descifrado en cada uno de los tres niveles de seguridad establecidos. Los resultados obtenidos se muestran en la tabla 4.4. El proceso de descifrado es el inverso al proceso de cifrado realizado por el servidor lo cual nos permite obtener los datos en claro y posteriormente son llevados a la aplicación y con ello garantizar la confidencialidad sobre la red de área local inalámbrica.

Como se observa en la figura 4.4 el descifrado mas eficiente dentro de las plataformas clientes es el que se obtiene en la Laptop, seguido de la plataforma Desktop y por ultimo la PDA. Además es de resaltar que la cantidad de datos que descifra la PDA es menor a las de las plataformas Laptop y Desktop como se observo en la sección anterior. El algoritmo que

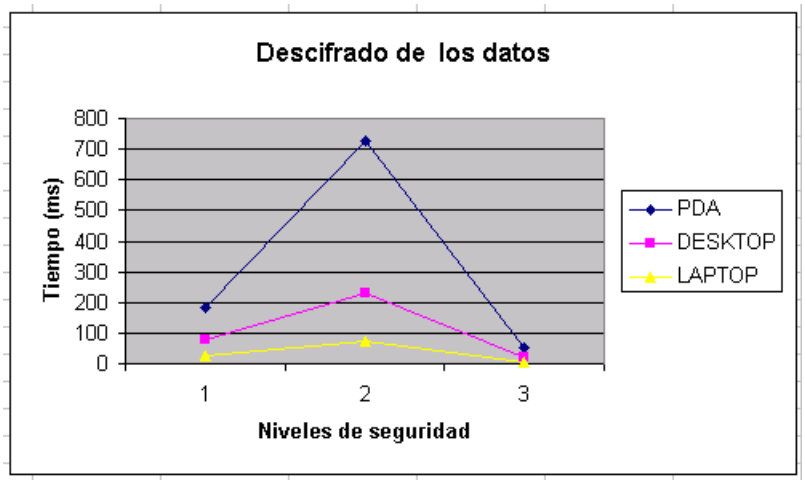


Figura 4.4: Grafica de tiempos de cómputo para la fase de descifrado.

implica mayor tiempo de descifrado es el TDES de igual forma que ocurrió en la fase de cifrado de datos y el mejor es el algoritmo AES.

### 4.3.4. Utilización de memoria del dispositivo

Uno de los aspectos que se deben de tomar en cuenta cuando se trabaja bajo restricciones de capacidad de memoria sobre todo en la plataforma PDA (como se mostró en la tabla 4.1) es la cantidad y tamaño de los archivos que se usan dentro del sistema.

La tabla 4.5 presenta el tamaño de la librería que implementa los componentes del middleware a la cual se le denomino libMiddleware. Además

Tabla 4.4: Descifrado de los datos.

| **Niveles** | **PDA** | **DESKTOP** | **LAPTOP** |
|---|---|---|---|
| Nivel 1 | 185 ms | 76 ms | 24 ms |
| Nivel 2 | 726 ms | 232 ms | 71 ms |
| Nivel 3 | 51 ms | 21 ms | 7 ms |

Tabla 4.5: Tamaño de los archivos.

| **Componente** | **Tamaño (KB)** |
|---|---|
| libMiddleware | 1028 |
| Cliente | 664 |
| Servidor | 51 |

se presenta el tamaño total de la aplicación cliente y la aplicación servidor. Como se observa el tamaño de la aplicación cliente es de 664 kb, lo cual no resulta tan grande si se toma en cuenta que dentro de este binario ya se encuentran las funciones de nuestro middleware, puesto que libMiddleware no se requiere dentro de la parte cliente o servidor, es decir cada una de las aplicaciones es independiente de dicha librería.

### 4.3.5. Degradación de la QoS de la aplicación cliente

En las aplicaciones de visualización de video es necesario considerar que la calidad de servicio que se brinda al usuario debe ser la óptima o por lo menos se debe estar muy cerca de ella. En este aspecto, el factor que influye directamente en la disminución de la calidad de servicio de una aplicación, es la frecuencia de despliegue de los frames. Esta frecuencia está dada por el número de frames que despliegan en un periodo de tiempo (frames x segundo). Por ejemplo, la frecuencia óptima utilizada en un video cinematográfico es de 24 frames por segundo; para un video digital su frecuencia óptima es de 30 frames por segundo. Para el caso de envió y recepción de video digital en tiempo real a través de una red, la frecuencia óptima está en el rango de 15 a 20 frames por segundo [4, 31]. Tomando en cuenta lo anterior la frecuencia de envió y recepción de frames con respecto al tiempo se observa en la tabla 4.6.

De acuerdo a la tabla 4.6 para estar en un nivel optimo de visualización de video es necesario procesar cada 50 ms un frame de video, para que al transcurso de un segundo se logre la frecuencia de 20 frames por segundo

Tabla 4.6: Frecuencia de envió y recepción.

| **Frames x segundo** | **Tiempo (ms)** |
|---|---|
| 15 | 66.6 |
| 16 | 62.5 |
| 17 | 58.8 |
| 18 | 55.5 |
| 19 | 52.6 |
| 20 | 50 |

(la cual es la frecuencia optima que se necesita en una aplicación de envió y recepción de video en tiempo real en una red). Por lo tanto, nuestra aplicación cliente dispone de 50 ms para procesar un frame para estar dentro de un nivel optimo de calidad de servicio o bien dispone de 66.6 ms para procesar un frame para alcanzar al menos un nivel aceptable de calidad de servicio; pero el simple hecho de que nuestra aplicación cliente implementa el middleware hace que exista una sobre carga inherente en el procesamiento de los frames.

En la tabla 4.4 se observó que la plataforma que presenta peor desempeño en el descifrado de los datos es la plataforma PDA. Por lo tanto, esta plataforma es la que mas sobre carga agregaría a nuestros clientes. El tiempo mínimo que se tarda en descifrar un frame de video es de 51 ms, lo cual implica solo el proceso de descifrado hace que la aplicación no esté dentro del nivel óptimo de calidad de servicio, pero si está dentro de un nivel de calidad de servicio aceptable puesto que su frecuencia esta entre 15 a 18 frames por segundo. Para el tiempo máximo de descifrado de la PDA( el cual fue de 726 ms obtenido con el descifrado para el algoritmo TDES) la aplicación presenta una calidad de servicio mala, puesto que su frecuencia esta dada en 1.1 frames por segundo. A diferencia de la plataforma PDA, la plataforma que logró los mejores tiempos en el proceso de descifrado fue la Laptop. El mejor tiempo de descifrado de la laptop fue de 7 ms, con el cual podemos garantizar que su calidad de servicio es la óptima(puesto que su frecuencia es

de 20 frames por segundo). El peor tiempo de descifrado fue de 71 ms, con lo cual se observa que la aplicación excedería el rango de tiempos establecidos para estar dentro de un nivel aceptable de calidad de servicio (de acuerdo a la tabla 4.6). Por lo tanto la aplicación solo puede tener un nivel de calidad de servicio aceptable puesto que su frecuencia está en 13 frames por segundo.

# Capítulo 5

# Conclusiones y trabajo futuro

En este capítulo, se expresan las conclusiones que se obtuvieron del desarrollo y prueba de este trabajo de tesis, al mismo tiempo se expone una serie de posibles caminos para continuar y darle seguimiento a este tema considerándolo como un posible trabajo futuro.

## 5.1. Conclusión

La seguridad de los sistemas informaticos es una tarea difícil de efectuar, ya que implica de una mayor experiencia de los diseñadores del sistema y de mayores recursos computacionales. Recientemente, han habido numerosas investigaciones dedicadas a brindar mayor seguridad a sistemas de cómputo, sin embargo, siempre existen vulnerabilidades en el sistema que impiden garantizar totalmente seguridad de un sistema. Otro punto a destacar en cuanto a seguridad es que el constante desarrollo tecnológico, es difícil siempre proporcionar un optimo nivel de seguridad, ya que el hardware y el software están siempre en constante mejora.

Por otro lado, en muchos sistemas de computo, la seguridad no siempre es considerada como el factor mas importante en el diseño. La nueva tendencia del desarrollo tecnológico, en el cual se encuentran las PDAs y las redes

de área local inalámbricas, son de gran utilidad ya que proveen movilidad, flexibilidad, sin embargo presentan problemas a la seguridad debido a que están basadas en una plataforma abierta poco segura y muy vulnerable a ataques externos.

El hecho de que una PDA tiene limitaciones de recursos hace que los desarrolladores de estos sistemas no tomen en cuenta la seguridad como atributo en sus aplicaciones, ocasionando con ello un gran vació de seguridad en estos dispositivos. Para el caso de las redes de área local inalámbricas, por el simple hecho de usar como medio de transmisión la radio frecuencia (RF), las vuelve vulnerables. Es decir, el canal de comunicación es inherentemente inseguro y puede ser atacado de manera pasiva comprometiendo la confidencialidad de los datos, o bien de manera activa en la cual un intruso puede enviar, recibir, alterar o falsificar mensajes.

Sin embargo podemos decir que la seguridad es un atributo que no se puede cubrir de manera total, sino de forma parcial solamente. Por lo tanto, en este trabajo de tesis se presenta una propuesta la cual consiste en el desarrollo de un middleware que provee seguridad flexible y ajustable mediante distintos niveles de seguridad.

Estos niveles de seguridad pueden ser ajustados en cualquier instante de tiempo especifico en cualquier aplicación con arquitectura cliente- servidor. En nuestra aplicación se considera además el efecto que produce sobre la calidad de servicio el ajuste de dichos niveles de seguridad para la aplicación.

En nuestro desarrollo efectuamos pruebas sobre un sistema de transferencia de video en tiempo real sobre una red de área local inalámbrica . Los resultados obtenidos muestran que los niveles de seguridad no son equivalentes en cuanto a los niveles de latencia. Observamos que el algoritmo AES es el mas óptimo en cuanto a tiempo de cómputo y nivel de seguridad.

Las pruebas realizadas han proporcionado resultados favorables en la aplicabilidad, la flexibilidad y la utilidad del middleware dentro de sistema de transmisión de video en tiempo real en una red de área local inalámbrica. Por

lo tanto podemos concluir que este trabajo de tesis cumple con los objetivos trazados al inicio de este trabajo.

## 5.2. Trabajo futuro

La tesis desarrollada provee una plataforma de trabajo para sistemas multimedia de tiempo real que operan sobre redes de cómputo inalámbricas. En general este desarrollo puede ser visto como una base para futuros desarrollos que ayudarían a que este middleware brindara aun mas beneficios en el desarrollo de aplicaciones cliente-servidor sobre redes de área local inalámbrica.

El trabajo futuro que se propone realizar sobre esta plataforma es el siguiente:

- Desarrollar un módulo para garantizar integridad en la transferencia de los datos.
- Desarrollar nuevos niveles de seguridad (variando los parámetros en los algoritmos de cifrado).
- Estudiar la QoS de redes para poder integrarlo con el middleware.
- Integrar el middleware a un kernel de tiempo real, para que aun sea mas transparente para las aplicaciones.
- Desarrollar algoritmos óptimos de asignación de seguridad y calidad de servicio para sistemas de tiempo real.

# Bibliografía


[1] A. Meneses, P. Van Oorschot, S. Vanstone. *Handbook of Applied Cryptography*. CRC Press, New York 2001. Quinta Edición.

[2] Anand Raghunathan, Srivaths Ravi, Sunil Hattangady and Jean-Jacques Quisquater. *Securing Mobile Appliances: New Challenges for the System Designer*. NEC Laboratories, USA; Texas Instruments Inc., USA; Universite catholique de Louvain, Louvain-la-Neuve, Belgium. 2003.

[3] Andrew S. Tanenbaum.*Redes de computadoras*. Prentice Hall, USA 1996.Tercera edición.

[4] Aura Ganz, Zvi Ganz, Kitt Wongthavarawat. *Multimedia Wireless Networks: Technologies, Standards and QoS*. Prentice Hall, USA 2003.

[5] Bluetooth. *Specification of the Bluetooth System*. Version 1.1, February 22 2001.

[6] Christian Poellabauer and Karsten Schwan. *Power-Aware Video Decoding using Real-Time Event Handlers*. College of Computing Georgia Institute of Technology. 2002.

[7] Chui Sian Ong, Klara Nahrstedt and Wanghong Yuaan. *Quality of protection for mobile multimedia applications*. Department of Computer Science University of Illinois at Urbana-Champaign.

[8] D. E. Comer. *Internetworking with TCP/IP. Vol. I: Principles, Protocols and Architecture*. Prentice-Hall, USA 1991. Segunda Edición.

[9] Daemen J. and Rijmen V. *The Design of Rijndael*. Springer-Verlag,Germany, 2002.

[10] David K. Y. Yau and Simon S. Lam. *Adaptive Rate-Controlled Scheduling for Multimedia Applications*. IEEE/ACM transactions on networking, vol. 5, n° 4.1997.

[11] Diffie W. and Hellman M. *New directions in cryptography.* IEEE Trans. Information Theory, pages 644–654, 1976. IT-22(6).November 1976.

[12] Geier Jim. *Wireless Lans*. SAMS, USA 2002. Segunda Edición.

[13] Handhelds Linux Familiar.*Distribucion de linux embebido.* http://www.handhelds.org, 2002,versin 0.7.

[14] Jahanzeb Khan, Anis Khwaja. *Building secure wireless networks with 802.11*. Wiley, USA 2003.

[15] K. Nichols Randall, C. Lekkas Panos. *Wireless Security: Models, Threats, and Solutions*. McGraw-Hill, USA 2002. pp 329-388.

[16] Kay A. Robbins, Steven Robbins. *Unix systems programming, comunication, concurency and threads*. Prentice Hall, 2003.

[17] M. van der Heijden, M. Taylor. *Understanding WAP: Wireless Applications, Devices, and Services*. Artech House, USA 2000.

[18] Manuel J. Lucena López. *Criptografía y seguridad en computadores*. http://www.di.ujaen.es/ mlucena. 2003.

[19] Matthew S. Gast. *802.11 Wireless Networks The Definitive Guide*. O'Reilly, USA 2002. pp 14-34.

[20] Merritt Maxim and David Pollino. *Wireless Security*. McGrawHill/Osborne, 2002.

[21] Randall K. Nichols, Panos C. Lekkas. *Seguridad para comunicaciones inalámbricas*. McGraw-Hill, 2003. Primera edición. pp 187-420.

[22] Reyes Montiel Laura. *Estudio, diseño y evaluación de protocolos de autenticación para redes Inalámbricas*. Departamento de Ingeniería Eléctrica, sección de computación, CINVESTAV-IPN. 2003.

[23] Robert Laberge, Srdjan Vujosevic. *Building PDA Databases for wireless and mobile development*. Wiley, 2003. Primera edición. pp 187-420.

[24] Sandeep Singhal, Thomas Bridgman, Calita Suryanarayana and et al. *The Wireless Application Protocol: Writing applications for the mobile internet*. Addison-Wesley, 2001.

[25] Savas E., Rodríguez F., Koc C., and et al. *RCT, RSA and ECC Toolkit*. ISL Group, Oregon State University, 2002.

[26] Srijan Chakraborty and David K. Y. Yau. *Predicting Energy Consumption of MPEG Video Playback on Handhelds*. Departament of Computer Sciences, Purdue University, West Lafayette, USA.

[27] Stephen Thomas. *SSL and TLS essentials*. Wiley, USA 2000.

[28] Real-Time Application Interface.*RTAI*. www.rtai.o rg.

[29] W. Richard Stevens, Bill Fenner, Adrew M. Rudoff. *Unix network programming the sockets networking API*. Vol. I, Addison-Wesley, 2004. Third edition.

[30] RSA Laboratories. http://www.rsasecurity.com/rsalabs/.

[31] UNIVERSIDAD EAFIT Departamento de Informática y Sistemas. *Sistemas y Aplicaciones Multimedia* http://dis.eafit.edu.co/cursos/st780/material/.

[32] Wanghong Yuan, Klara Nahrstedt. *Energy-Efficient Soft Real-Time CPU Scheduling for Mobile Multimedia Systems*. Departament of Computer Science University of Illinois at Urbana-Champaign, USA. 2003.

[33] WAP Forum. *WAP-261-WTLS-20010406-a, WTLS Specification*. http://www.wapforum.com, 2001. Versión 06-Abr- 2001.

[34] William A. Arbaugh, Narendar Shankar, Y.W. Justin Wan. *Your 802.11 Wireless Network has No Clothes*. University of Maryland, Department of Computer Science, USA 2001.

[35] Zuccherato R. and Adams C. *Using Elliptic Curve Diffie-Hellman in the SPKM GSS API*. 1999.