MESEM: MECANISMO DE SEGURIDAD PARA PRODUCTOS MULTIMEDIA

■ John Font
john_font_1@ucab.edu.ve

Tutor: Lourdes Ortiz
lortiz@ucab.edu.ve

RESUMEN

El presente proyecto, titulado Mecanismo de seguridad para productos multimedia generados en la Universidad Católica Andrés Bello (MESEM-UCAB), tiene como objetivo principal, desarrollar y evaluar una solución viable para la protección del ***software*** multimedia desarrollado en la UCAB.

Una vez planteado el proyecto, se realizó una investigación documental sobre los mecanismos de seguridad existentes y otras áreas relacionadas, seguido de un levantamiento de información de los requerimientos de la Universidad Católica Andrés Bello, con esta información se procedió al desarrollo del prototipo del mecanismo de seguridad para *software* multimedia, durante el cual se desarrolló una arquitectura de seguridad para *software,* enfocada a servir como gula para eí desarrollo del mecanismo.

Al culminar todas estas fases del desarrollo y de acuerdo a lo estimado se logró obtener, una tabla comparativa de los distintos mecanismos de seguridad existentes, clasificados y con recomendaciones de uso; el levantamiento de información de los requerimientos de la UCAB en cuanto a protección de *software;* un mecanismo de evaluación de mecanismos de seguridad para *software;* una arquitectura de seguridad para *software* y el prototipo del mecanismo de seguridad para *software* multimedio.

Durante el desarrollo del proyecto se llegó a un conjunto de conclusiones entre las que destacan que

existe poca información sobre los mecanismos de seguridad para *software* debido a que basan su seguridad en el algoritmo, por lo que se trata de mantener en secreto; que muchos mecanismos de seguridad se basan en características especificas de algunos *hardware* o *software;* que los mecanismos de seguridad para *software* no son infalibles, pero lo que se busca con ellos es una forma de persuasión; que hay que tener en cuenta que un mecanismo de seguridad es bueno si el costo para violarlo supera las ganancias que se pueden obtener al utilizarlo; la seguridad de la información digital es tan segura como el medio donde se encuentre; el desarrollo o implementación de un mecanismo de seguridad debe ser a la par del desarrollo del *software,* y que la selección de un mecanismo de seguridad no es sencillo, por lo que se recomienda la utilización de un sistema de apoyo o arquitectura que ayude en su diseño.

Palabras Claves: Seguridad, *Software,* Arquitectura, Modelo, Evaluación.

PLANTEAMIENTO DEL PROBLEMA:

Muy a pesar de los grandes estudios, inversiones y productos realizados para combatir el uso ilegal de *software* a escala mundial, todavía se plantea como una problemática sobre todo económica, que ha generado grandes perdidas en todo el mundo.

Más específicamente en Venezuela, esta problemática ha venido en un incremento bastante acelerado en los últimos cuatro años, alcanzando un grado de piratería del 58% para el año 2000 según cifras de la BSA (2001) en su Informe Global sobre Piratería de *Software;* con una pérdida de 20,792 millones de dólares. Por este motivo se ha iniciado la lucha antipiratería mediante el desarrollo de nuevas leyes en el país. Las empresas han asumido dos puntos de vista en cuanto a la protección de *software,* la primera es que la piratería de *software* le hace propaganda a los productos que se copian ilegalmente, debido a que los usuarios de los mismos, al tener la oportunidad de utilizar el producto, conocen la calidad del mismo y si les gusta lo más probable es que lo adquieran o que adquieran una versión más actualizada y la segunda, es tratar de impedir la copia ilegal mediante mecanismos de protección cónsonos a sus necesidades.

En este contexto, se presentaron una serie de interrogantes por parte de algunas de las dependencias de la Universidad Católica Andrés Bello, principalmente por la Escuela de Ingeniería Informática, específicamente en el Grupo de Investigación en Ingeniería del *Software* (G.I.I.S.), y por el Centro de Investigación de Comunicación (C.I.C.). Algunas de estas interrogantes fueron:

1. ¿Cuáles son los mecanismos de seguridad para *software* existentes en la actualidad y cuáles son las condiciones recomendadas para su aplicación?
2. ¿Cómo pueden ser evaluados los mecanismos de seguridad existentes?
3. ¿Será posible desarrollar un mecanismo de seguridad de *software* adaptable al contexto de la UCAB?

Buscando respuesta a estas interrogantes, se planteó el presente proyecto con el objetivo principal, y el conjunto de objetivos específicos descritos a continuación.

OBJETIVOS

Objetivo general:

Desarrollar y evaluar una solución viable para la protección del *software* multimedia desarrollado en la Universidad Católica Andrés Bello.

Objetivos específicos:

Conocer los mecanismos basados tanto en *hardware* como en *software* existentes, que permitan la protección contra copias ilegales de *software,* principalmente *software* multimedia, estableciendo una tabla comparativa entre los mismos.

Establecer las diferentes necesidades de seguridad de *software* requeridos por la Universidad Católica Andrés Bello, principalmente en el Centro de Investigación de la Comunicación (CIC), evaluando los mecanismos utilizados actualmente, si este fuera el caso.

Establecer los métodos de evaluación de los mecanismos de seguridad de *software* existentes en el mercado y utilizados por compañías como Microsoft, Sybase y Sun Microsystem, para evaluar la eficiencia de los mismos.

Desarrollar un prototipo de un mecanismo de seguridad de productos multimedia propios de la Universidad Católica Andrés Bello, basados tanto en *hardware* como en *software,* adaptado a su contexto y necesidades, para garantizar que se mantengan como los principales medios de difusión de información sobre nuestra historia y nuestra cultura.

MARCO DE REFERENCIA

Para comenzar, al problema de la seguridad de *software* se le relacionan tres principales áreas, como son: la tecnología, los aspectos legales y las investigaciones de distintas organizaciones, que están relacionadas a esta problemática y que se presentan a continuación:

1. Tecnología.

Según el Instituto Tecnológico de la Paz, la tecnología "es un conjunto ordenado de instrumentos, conocimientos, procedimientos y métodos aplicados en las distintas ramas industriales".

De la cual se desprende que, la tecnología es aquella que nos permite optimizar los procesos y además son todos aquellos conocimientos organizados que facilitan el desarrollo de un producto, siendo importante organizar la amplitud de conceptos y conocimientos existentes en el área de investigación de la seguridad de *software* antes de abordarlos para su aplicación.

La tecnología existente nos apoya en el día a día, y también nos puede apoyar en el desarrollo de nuevos medios de protección antipiratería de *software.*

1.1. Estándares de seguridad

En la actualidad se han desarrollado diversidad de reglas, que han permitido estandarizar una gran cantidad de procesos y en una gran variedad de áreas, lo que permite el desarrollo controlado de aplicaciones, además de apoyar el desarrollo, mejorando la calidad. Una de ellas es la seguridad computacional, para ello existen organizaciones que han desarrollado estándares, como es por ejemplo, el estándar de la arquitectura de seguridad de datos 051, desarrollado por la ISO, el cual se explica más adelante en este mismo capítulo.

1.2. Protección de *software* a través de *hardware*

La protección basada en *hardware,* como su nombre lo dice son todos aquellos medios de seguridad, que han sido desarrollados por un conjunto de empresas abocadas a la lucha contra la piratería de *software,* con la utilización de algún medio físico o las características de los mismos.

Existen una gran variedad de ellos, y se pueden dividir en distintas categorías dependiendo del medio físico que utilizan, como por ejemplo la protección en CD, la protección en discos flexibles, y otras formas de protección como son las llaves electrónicas o dispositivos de seguridad en la memoria. Para comprender mejor cada uno de ellos es necesaria comprender el funcionamiento de los mismos, y saber cuáles son los mecanismos de seguridad de este tipo.

1.2.1. Protección en CD.

Para poder hablar de los mecanismos de protección en CD, es necesario conocer un poco más sobre lo que es un CD y cómo funcionan sus lectoras.

Los CD están hechos según Maikel (2000) de un material plástico llamado olicarbonato, con agujeros formando una especie de espiral, y sobre él hay una capa de laca y plásticos, que protegen a los agujeros para que no se tapen o se creen nuevos.

Según Maikel (2000), "La información digital del disco se almacena en un área que comienza a 25 mm del centro y se extiende hasta los 58 mm. Bordeando esta área existen dos anillos o guías, uno interno y otro externo. La guía interna contiene la tabla de contenidos del disco (lead in), y permite al láser sincronizarse y saber el contenido de información de audio o los datos antes de proceder a su lectura. La longitud de la guía interna depende de las dimensiones de la tabla de contenidos (que puede almacenar hasta 99 pistas de audio 33mm). A continuación viene la información del CD, capaz de almacenar hasta unos 76 min. de audio y 99 pistas como máximo. Finalmente se encuentra la guía externa (lead out), que marca el fin de los datos (1 mm de ancho). Este esquema es válido tanto para discos compactos de audio corno de datos, aunque puede haber variaciones sobre todo en formatos híbridos (audio+datos)."

Y los lectores de CD están compuestos de un cabezal, donde hay un láser que envía un haz de luz al disco y posee según Maikel (2000) "un fotorreceptor (foto-diodo)" el cual recibe el rebote de la luz en el disco.

Los CD, se leen desde la parte inferior y lo que realmente se lee son salientes y no salientes.

Ya sabiendo un poco más sobre lo que es un CD y el funcionamiento de las lectoras de CD, podemos adentramos en tos mecanismos de seguridad que basan su protección en la modificación de este medio de distribución digital o en la violación de los estándares de escritura de los mismos.

Muchos de los mecanismos utilizados en la actualidad se basan en realizar una marca digital al momento de la producción del CD, estas marcas pueden ser físicas o lógicas, y tratan de no permitir ser copiadas, con lo cual sólo basta verificar que el CD contiene la marca para saber si es el original o no.

1.2.2. Protección en *Floppy*

La protección en *floppy,* son todos aquellos mecanismos que se basan en un disco flexible, estos mecanismos casi no se utilizan, generalmente se usa el *floppy* como una llave electrónica que contiene la clave del CD para instalarlo o utilizarlo y se protegían para evitar su copia como por ejemplo con muecas en disco.

1.2.3. Otras formas de protección

Además de las distintas formas de protección presentadas anteriormente, existen otras formas de proteger *software* basado en *hardware,* como por ejemplo son aquellas que utilizan algún *hardware ex*tra, que ayuda a identificar si la persona que está utilizando el producto es la autorizada o no.

1.3. Protección de *software* a través de *Software*

La protección basada en *Software,* son aquellos mecanismos que se basan en programas desarrollados para proteger los datos o para tratar de detectar si el *software* que se intenta proteger esta siendo utilizado de forma ilegal.

Además de todos los algoritmos existentes, se siguen desarrollando un conjunto de nuevos algoritmos y arquitecturas de seguridad computacional, siguiendo la búsqueda de mejores métodos de protecciones.

1.3.1. Algoritmos más conocidos

En la actualidad existe un conjunto de algoritmos desarrollados, los cuales se pueden dividir en dos categorías, los especificas y los generales.

Cuando se habla de algoritmos generales, se habla de todos aquellos, que son desarrollados ofreciendo protección a todo el material contenido en el medio de almacenamiento secundario protegido, sin pensar en el tipo de elemento a proteger.

Mientras que cuando se habla de algoritmos específicos, se refiere a todos aquellos que han sido desarrollados pensando en proteger algún tipo de elemento almacenado en especial y bajo unas condiciones determinadas, como por ejemplo la encriptación de datos.

1.3.1.1. Protección General

Cuando se habla de protección general, se refiere a que el mecanismo de seguridad se encarga de proteger todo, sin importar en qué formato se encuentra lo que se intenta proteger, por ejemplo aquellos mecanismos encargados de proteger todo lo que se encuentre en un medio de almacenamiento como un CD sin importar el contenido del mismo.

Por lo general esta protección intenta impedir la copia del material protegido o de impedir su instalación.

1.3.1.1.1. Protección en Windows

Uno de los factores importantes a tener en cuenta durante el desarrollo de un mecanismo de seguridad, es la plataforma donde funciona el *software* que se desea proteger, debido a que las distintas plataformas como por ejemplo Windows, en sus diferentes versiones contienen distintas características que pueden ser usadas para la protección de la información o de un *software,* como par ejemplo la capacidad de colocar a un archivo la propiedad de oculto impidiéndole a un usuario ver ese archivo. Además de este, existen otras propiedades más avanzadas pero que varían de una versión a otra.

1.3.1.2. Protección Especializada

Existe un conjunto de algoritmos y mecanismos de seguridad destinados a proteger tipo de elementos específicos, por ejemplo proteger datos o la transmisión de datos, como es el caso de la arquitectura de seguridad OSI de la ISO, o el ocultamiento de la información. Además de otros mecanismos de seguridad basados en el tipo o formato de un archivo en especifico.

1.3.1.2.1. Ocultamiento de Información

El ocultamiento de la información o *Information hidding* es, como su nombre lo indica, ocultar la información para que no sea descubierta fácilmente, para ello existen diversas técnicas como son:

- La criptografía o cifrado: que es una de las principales técnicas para esconder la información. Según Oppliger (1998, pág. 11) "se utiliza para proteger la confidencialidad de las unidades de datos y la información de flujo de tráfico, o para dar soporte o complementar otros mecanismos de seguridad".
- La esteganografia: basado en StegoArchive.com (2000), la esteganografía se basa en ocultar la existencia como tal de un elemento, como por ejemplo esconder un mensaje dentro de una imagen.

En "Watermarks" y "Fingerprints": según Garrido (2000) "es una o varias marcas o señales transparentes o imperceptibles que se incluyen en material de audio o de vídeo con el propósito de prevenir copias no autorizadas. Estas marcas son inaudibles, invisibles y además no es posible borrarlas,

estas marcas suelen ser de diversos tipos dependiendo del fabricante/desarrollador, podrían consistir en el titulo de la obra y su autor, o bien en uno o varios códigos con determinada información."

1.3.2. Principios de arquitecturas de protección ISO.

Existe un conjunto de mecanismos de seguridad de la OSI, denominado mecanismos de seguridad O31 para datos, los cuales brindan un conjunto de servicios bien definidos y preestablecidos por la ISO.

Según Oppliger (1998, pág. 7) "La arquitectura de seguridad 031 proporciona una descripción general de los servicios y mecanismos relacionados con la seguridad, y discute sus interrelaciones. Muestra también las correspondencias entre la arquitectura de seguridad y la arquitectura estándar, y discute su situación apropiada dentro del modelo OSI-RM". La clasificación de los servicios de seguridad y sus correspondientes mecanismos de seguridad se presentan a continuación.

2. Investigaciones

En la actualidad existen organizaciones enfocadas en la seguridad computacional de los cuales un grupo esta enfocado en la investigación sobre seguridad de *software,* además de esto existe un conjunto de investigaciones sobre otras áreas relacionadas como es la evolución del *software.*

Servicios de seguridad OSI	
Servicios de autentificación	Autentificación de entidades parejas
	Autentificación del origen de los datos
Servicios de control de acceso	
Servicios de confidencialidad de datos	Confidencialidad orientados a conexión
	Confidencialidad no orientados a conexión
	Confidencialidad de campo selectivo
	Confidencialidad de flujo de tráfico
Servicios de integridad de datos	integridad orientados a conexion con recuperación
	Integridad orientados a conexión sin recuperación
	Integridad de campo seleccionado orientados a conexión
	Integridad no orientados a conexión
	Integridad de campo selecc ionado no orientados a conexión.
Servicios de no rechazo	No rechazo con prueba de origen
	[No rechazo con prueba de

Tabla No 1. Servidos de seguridad OSI, basado en Oppliger (1998)

Mecanismos de seguridad	
Mecanismos de seguridad específicos	Cifrado
	Firma digital
	Control de acceso
	Integridad de datos
	Intercambio de autentificación
	Relleno de tráfico
	Control de encaminamiento
	Certificación
Mecanismos de seguridad generalizados	Funcionalidad de confianza
	Funcionalidad de seguridad
	Detección de eventos
	Rastreo de autoría de seguridad
	Recuperación de seguridad

Tabla No 2. Mecanismos de seguridad 051, basado en Oppliger (1998).

2.1. Evolución del *Software*

Debido a la rápida evolución del *software* desde sus inicios, ha provocado que el tiempo de vida del mismo sea por lo general corto, por esto se presenta el caso de que los mecanismos de seguridad no necesariamente tienen que ser inviolables debido a que sólo tienen que ser capaces de soportar el tiempo de vida del software, hasta que aparezca una nueva versión en el mercado, perdiendo así su valor.

2.2. "Hackers"

Según Pague (1996), un "hacker" es, "un individuo que ansia conocimientos, disfruta explorando los detalles de un sistema operativo o un lenguaje de programación, programa constantemente (incluso obsesivamente), disfruta más programando que sólo haciendo teorías sobre programación y disfruta del reto intelectual de vencer limitaciones buscando constantemente aumentar sus capacidades". Gracias a ese disfrute de vencer las limitaciones, es que los "hacker" buscan violar la seguridad de los sistemas, y existe una derivación de los "hacker" denominados "cracker" que buscan violentar el acceso a los programas adquiriendo su clave o burlando su seguridad para poder utilizarlos de forma ilegal.

Por esas ansias de buscar nuevos retos es que se ha vuelto tan complicado el desarrollo de un mecanismo de seguridad de software, debido a que siempre están investigando y desarrollando nuevos métodos para violar esos mecanismos y toman a un nuevo mecanismos como un nuevo reto, lo que ha hecho muy competitiva esta área de desarrollo.

2.3. GIIS

El Grupo de Investigación en Ingeniería de *Software,* tiene como objetivo principal desarrollar proyectos que se enmarcan en una diversidad tecnológica que conduce hacia la investigación en nuevas herramientas, metodologías y paradigmas en el mundo de la Ingeniería del *Software.* Entre los proyectos actualmente en proceso se destaca el desarrollo del proyecto "MISS Modelo Inteligente de Seguridad de *Software',* el cual cuenta con el apoyo del Consejo de Desarrollo Científico y Humanístico (CDCH) de la UCAB.

2.3.1. Nivel de protección

Según el G.I.I.S. (2001), cuando se habla sobre protección de *software se* puede hablar de tres niveles de protección corno son, la protección de copia, la protección de instalación y la protección de ejecución.

El G.I.I.S. plantea *un* modelo de evaluación de los mecanismos de seguridad de *software* más adecuados para cada tipo de proyecto, el cual se

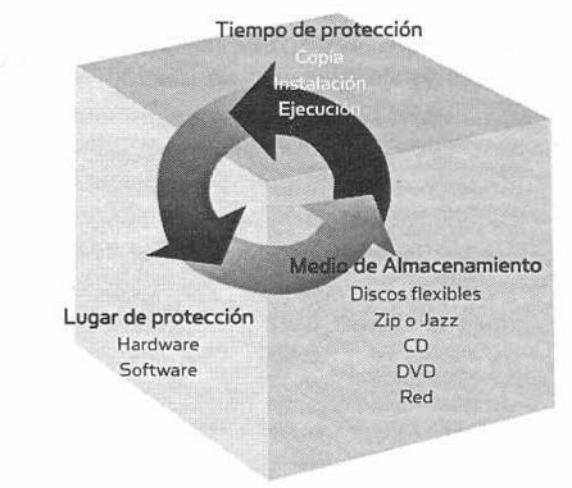

Fig. No 1. Modelo de clasificación para mecanismos de seguridad GIIS, tomado del GIIS (2001)

caracteriza por ser un modelo multidimensional sistémico, basado en tres líneas o ejes principales (lugar de protección, tiempo de protección y medio de almacenamiento), de acuerdo a las variables de clasificación de mecanismos de seguridad antes mencionados (ver figura #1: Modelo de clasificación para mecanismos de seguridad GIS).

En el modelo se plantea que hay tres maneras de clasificar a un mecanismo de seguridad, la primera, es el medio de almacenamiento, en donde se clasifica al mecanismo dependiendo del medio donde se distribuye o almacena el *software,* la segunda dimensión, se refiere al lugar de la protección; si la protección es basada en *hardware* o se basada en software; y la última el tiempo de la protección, en donde se clasifica dependiendo si la protección es durante la copia del producto, durante la instalación o cuando se ejecuta el programa.

3. Aspectos legales

Como menciona Barrera, "La velocidad del Desarrollo Tecnológico e Intelectual de la sociedad, torna sumamente complicado que los legisladores puedan normar de inmediato las relaciones jurídicas que surgen de las transformaciones científicas. Problemas complejos e incluso insólitos irrumpen repentinamente, obligando a un permanente proceso de búsqueda de principios jurídicos aplicables, y perfeccionamiento de leyes. Si hablamos de los derechos intelectuales, es aún mayor la dificultad de percibir adecuadamente el objeto de derechos y crear un marco jurídico".

Como Barrera menciona, el desarrollo tecnológico obliga a que se actualicen las leyes sobre los derechos

de autor constantemente, debido a que se crean nuevas formas de difusión de información como por ejemplo el CD-ROM que ahora permite transmitir libros completos y cualquier tipo de información como sonidos, imágenes y hasta videos en un solo CD, algo que hace poco tiempo no podía imaginar, era casi imposible pensar que todo este tipo de información se pudieran incluir en un mismo medio, por lo cual no existía ninguna ley que protegiera los derechos de autor de la información en este medio.

En el caso de Venezuela ya existe un conjunto de leyes que penalizan la piratería, como lo expone la BSA de Venezuela en su página web (www.bsa.com.ve). Las leyes venezolanas prohiben la copia y el uso ilegal de programas de computación. Según la BSA (2001) "Esta protección tiene su origen en la Ley sobre Derecho de Autor y la Decisión 351 de fa Comisión del Acuerdo de Cartagena (Pacto Andino), las cuales expresamente protegen al *software* (incluyendo el soporte magnético y fa documentación gráfica) como una obra literaria. La única excepción a esta ley, es el derecho que adquiere el usuario que posee legalmente una licencia para realizar una sola copia con fines de resguardo o seguridad. Cualquier infracción a los derechos de las productoras de *software* puede generar una sanción civil o penal, según sea el caso."

3.1. Necesidades y acciones relativas a la protección del *software*

Gracias al crecimiento tecnológico y a que la piratería de *software sigue* planteando desafíos a las industrias y a la economía global, como se puede apreciar en las cifras aportadas por la BSA (2001), como lo indica el informe global sobre piratería de *software,* "Por primera vez en la historia del estudio, el índice de piratería mundial en el 2000 no se ha reducido sino que muestra un ligero incremento al situarse en el 37%." (ver gráfico #1: índice de piratería mundial según la BSA, tomado de la BSA)

Gráfico No 1. Índice de piratería mundial según la BSA, tomado de la BSA (2001).

Aunque se puede notar una disminución en las pérdidas en dólares debido a la piratería no significa el descenso de la piratería, según la BSA (2001) se trata del resultado de varios factores como son que el dólar estuvo fuerte en 2000, los precios de *software* siguieron bajando, además el mercado de *software* experimentó el menor índice de crecimiento desde 1994, gracias a estos factores es que se dio una ligera reducción de pérdidas debido a la piratería.

Según la BSA (2001), las pérdidas en dólares por piratería de *software en* el 2000 son de 11.750 millones de dólares, lo que hace a la piratería de *software* un gran problema a escala mundial, y por esta razón es que se presenta la necesidad de realizar una fuerte lucha antipiratería y desarrollar nuevos mecanismos de seguridad, además del la creación de nuevas leyes que se adapten constantemente a las necesidades actuales, debido al crecimiento y el desarrollo constante de la tecnología.

METODOLOGÍA

En el intento de definir una metodología para el desarrollo de este proyecto, se recurrió a la creación de una metodología fundamentada en la utilizada por Matout, y Moronta (1997), basada en una serie de elementos de investigación documental, planteada por De La Torre (1991, pág. xiv), para el cual "Una metodología no es otra cosa que un conjunto de proposiciones lógicas, graduadas y jerarquizadas, destinadas a facilitar y mejorar el ejercicio intelectual, la capacidad creadora de la mente humana en cualquier ramo del saber.", "La metodología va unida indisolublemente a la adquisición del conocimiento, y cada tipo de saber tiene a su vez una metodología específica adecuada a su naturaleza."

Complementariamente, en el contexto de desarrollo de *software* Piattini, Calvo-Manzano, Cervera y Fernández (2000, pág. 62) definen metodología "como un conjunto de procedimientos, técnicas, herramientas, y un soporte documental que ayuda a los desarrolladores a realizar nuevo *software.* Normalmente consistirá en un conjunto de fases descompuestas en subfases (módulos, etapas, pasos, etc.). Esta descomposición del proceso de desarrollo guía a los desarrolladores en la elección de las técnicas que debe elegir para cada estado del proyecto, y facilita la planificación, gestión, control y evaluación de los proyectos. Una metodología, por tanto, representa el camino para desarrollar *software* de una manera sistemática."

En virtud de todo lo antes expuesto, la metodología planteada para el proyecto desarrollado estuvo compuesta de tres etapas, la primera conformada por una serie de elementos de investigación, denominada fase de Investigación Documental, y la segunda fase, utilizada para el desarrollo del prototipo del mecanismo de seguridad antipiratería, la cual está basada en el modelo de entrega por etapas (McConell, 1998) y la tercera etapa conformada por la documentación y exposición del proyecto. (ver figura #2: Esquema de la metodología utilizada para el desarrollo del proyecto.)

FASE I: Investigación Documental

La finalidad de esta etapa es buscar y procesar toda la información referente al tema de investigación y desarrollo, y de todas las áreas de la computación relacionadas con el tema de la seguridad.

Por lo que el objetivo fundamental de esta etapa es organizar de una forma eficiente toda la información obtenida, esta etapa se divide en dos subfases como son:

1. Obtención del conocimiento: se basa en la búsqueda de fuentes de información, como son las fuentes tradicionales como las sugeridas por De La Torre (1991), las cuales incluyen fuentes bibliográficas como libros, revistas y publicaciones, entre otras. Y fas fuentes automatizadas, como la red global Internet y trabajos no publicados de centros de investigación.
2. Registro de las fuentes y captación del material en la investigación: la cual consiste en organizar la información y generar tablas, fichas resumen y diagramas sobre la información obtenida en la etapa anterior.

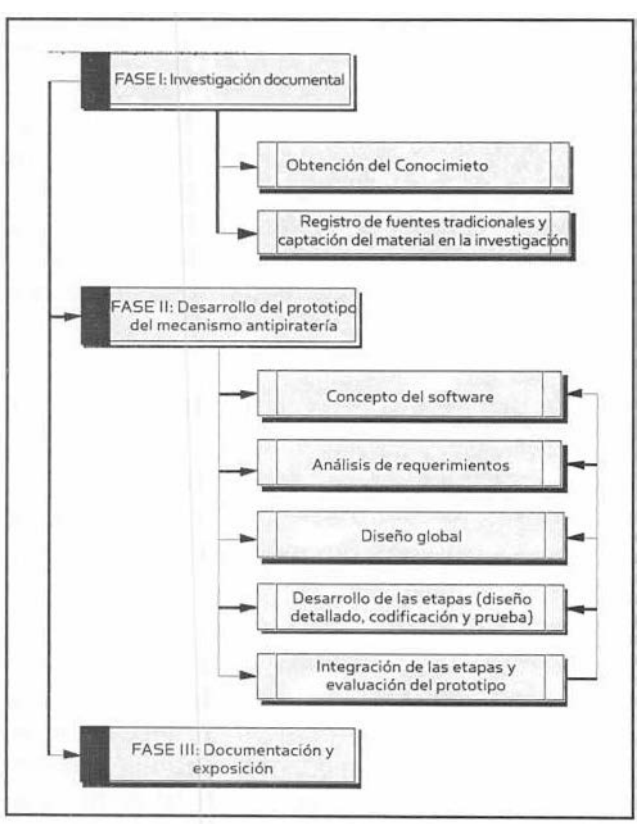

Fig. No 2. Esquema de la metodología utilizada para el desarrollo del proyecto

FASE II: Desarrollo del prototipo del mecanismo antipiratería

El desarrollo del prototipo del mecanismo de seguridad antipiratería de *software,* se basó en la metodología de entrega por etapas, la cual según McConell (1998, pág. 161), "es otro modelo de ciclo de vida en el que el *software* se muestra al cliente en etapas refinadas sucesivamente. A diferencia del modelo de prototipado evolutivo, cuando se utiliza la entrega por etapas, se conoce exactamente qué es lo que se va a construir cuando se procede a construirlo". Por lo que se plantea una metodología que permita el desarrollo por etapas independientes e interrelacionadas que permitan al final tener un producto completo, basado en etapas, todo esto debido a que se procura construir un mecanismo formado por un conjunto de submecanismos independientes y probados, cuya unión preste una seguridad integral al de cada uno por separado.

Esta metodología a la vez se conforma de una serie de subfases (ver figura #2: Esquema de la metodología utilizada para el desarrollo del proyecto), como son:

a. Concepto del *software.* La primera fase, denominada concepto del *software,* se basa en la definición de los objetivos del proyecto a desarrollar, buscando una visión clara de lo que se desea realizar y obtener al final del proyecto.
b. Análisis de requerimientos. En esta fase, se realiza el levantamiento de la información sobre los requerimientos básicos del mecanismo de seguridad antipiratería, para obtener toda la información necesaria para el diseño global, y posteriormente desarrollar cada etapa del proyecto.
c. Diseño global. Mediante los requerimientos básicos obtenidos en la fase anterior se procede al diseño del prototipo, en esta fase se decide en que cantidad de etapas *se* debe dividir el desarrollo del proyecto.
d. Desarrollo de la etapas. Para cada una de las etapas definidas en la fase anterior, se procede a

realizar el diseño detallado, la codificación de la etapa, la depuración y la prueba de la misma, con lo cual se obtiene un producto terminado, que conforma una parte del producto definitivo.

e. Integración de las etapas y evaluación del prototipo. Al finalizar todas estas fases se procede a integrar las distintas piezas o etapas del desarrollo y se realiza una evaluación al prototipo del mecanismo de seguridad para *software.*

FASE III: Documentación y exposición

En esta última etapa se elabora toda la documentación de este proyecto de investigación, esta etapa se comienza desde el principio del desarrollo del proyecto documentando cada etapa del desarrollo.

DESARROLLO

Para la realización del proyecto se realizaron una serie de actividades que conforman la metodología presentada anteriormente. A continuación se presentan estas actividades enmarcadas en las distintas fases de la metodología.

FASE I: Investigación Documental

En esta etapa se realizaron una serie de actividades con la finalidad de obtener la mayor cantidad de información referente al tema de investigación, que permitiera dar las bases para el desarrollo del proyecto, estas actividades son:

- Investigación en fuentes bibliográficas:
  - Internet.
  - Bibliografía proporcionada por profesores y expertos.
  - Biblioteca UCAB.
- Entrevistas a especialistas en el área.

En esta etapa se recolectó información referente a la seguridad de *software,* la principal fuente de investigación utilizada fue el Internet para la búsqueda de información referente a los mecanismos de seguridad antipiratería existentes en la actualidad, lo cual resultó una tarea complicada debido a la poca información existente sobre el tema, principalmente debido a que los mecanismos de seguridad específicamente para *software* son por lo general secretos, ya que en eso basan su garantía. Posteriormente se realizó una búsqueda exhaustiva sobre las áreas relacionadas con la seguridad de *soft ware,* como son, la seguridad de datos y demás algoritmos específicos, además de los estándares de seguridad, las investigaciones en esta área realizadas por distintas organizaciones, la construcción de nuevos algoritmos, la protección bajo Windows como principal plataforma de *software* comercial en ambiente de ausencia de red y el ocultamiento de información o "information hidding". Se estableció la relación entre estos elementos mediante un mapa mental de idea. (ver Apéndice A. Mapa mental de ideas).

Luego de realizar el levantamiento de información por medio de Internet, se procedió a contactar a una serie de especialistas en el área de seguridad, y a realizar reuniones con los mismos, a la mayoría se le realizó una entrevista estructurada (ver apéndice B Modelo de encuestas) para poder poseer un medio de comparación y análisis de las respuestas obtenidas. Adicionalmente fueron consultados otros especialistas en aspectos técnicos del proyecto. Este conjunto de especialistas contactados se conformó con profesores de la escuela de Ingeniería Informática especialistas en áreas relacionadas en el proyecto (Prof. Pedro Brao, Prof. Wilmer Pereira, Prof. Jacinto Fung), profesionales con experiencia aplicada en el área (Ing. José Lorenzo Flores, Grupo Santander) y especialistas de reconocimiento internacional en el área de seguridad (Dr. Miguel Soriano, Universidad de Cataluña — España.)

Gracias a este conjunto de entrevistas que logran combinar los conocimientos teóricos con los conocimientos prácticos y las experiencias de los mismos en esta área o áreas afines al proyecto, se desarrolló una tabla comparativa de las respuestas obtenidas, permitiendo un mejor análisis de las mismas.

FASE II: Desarrollo del prototipo del mecanismo antipiratería

Para esta segunda fase, denominada desarrollo del prototipo del mecanismo antipirateria, se siguió la metodología planteada anteriormente, cada fase, como se explica a continuación.

a. Concepto del *software.*

En esta etapa se procedió a mejorar y profundizar el concepto sobre el prototipo del mecanismo antipiratería a desarrollar, basándose en la investigación documental realizada en la primera fase del proyecto y en la selección del *software* a la que se le iba a aplicar el prototipo del mecanismo de seguridad antipiratería. Con esta nueva

conceptualización, ya más cercana a la definitiva, se procedió a realizar el análisis de requerimientos basado en las necesidades de seguridad de *software* de la Universidad Católica Andrés Bello.

b. Análisis de requerimiento.

Se realizaron una serie de encuestas (ver Anexo B: Modelo de encuestas.) a las principales organizaciones de la Universidad Católica Andrés Bello, encargadas en desarrollar *software* o material digital multimedia, que pudieran verse beneficiadas con un mecanismo de seguridad antipiratería.

Estas entrevistas fueron realizadas a: Lic. Héctor Alvarez perteneciente al Centro de Investigación de Comunicación — C.I.C; Lic. Daniel Graterol, director del Centro de Tecnología y Sistema - C.T.S y a la Lic. Carolina Marturet Medina, perteneciente a la Unidad Académica del Centro para la Aplicación de la Informática - C.A.I.

Mediante las entrevistas se buscaba conocer los requerimientos específicos en cuanto a qué tipo de recursos multimedios eran los que más necesitaban proteger y en qué medio de distribución, para lo cual se realizó un cuadro comparativo de los resultados de las encuestas, el cual es presentado en el capítulo de resultados (ver tabla #4: Resultados encuestas a organizaciones desarrolladoras de *software* en la UCAB).

c.- Diseño global.

Durante esta etapa, diseño global, se tomaron los resultados de la etapa anterior, y debido a la variedad de requerimientos que se podían presentar se procedió a no solo realizar el diseño del prototipo de un mecanismo antipiratería para la UCAB, sino que además se definió y planteó una arquitectura de seguridad para *software,* que permitiera en un momento dado desarrollar nuevos mecanismos de seguridad con la ayuda de la arquitectura, y que brindara una guía al momento de seleccionar el nivel de seguridad necesario, dependiendo de las necesidades planteadas y la disponibilidad de recursos.

Para comenzar el diseño del prototipo del mecanismo de seguridad para *software,* se tomó como base la arquitectura desarrollada en este proyecto, la cual recomienda que para construir un mecanismo de seguridad, es preferible estudiar principalmente los requerimientos de seguridad, para de esta forma poder conocer datos fundamentales del mecanismo, como por ejemplo el nivel de protección que se requiere, tos cuales pueden ser protección de copia, de instalación, de ejecución, o cualquier combinación de las mismas. Además de definir cual es el medio de almacenamiento y el lugar de la protección *(hardware* o *software).*

Al poseer estos datos fundamentales, se pasó a utilizar las recomendaciones de la arquitectura de seguridad para *software,* con lo cual se definieron cuáles mecanismos de seguridad o qué tipo de mecanismo de seguridad eran posibles de utilizar para el nivel de protección deseado, los cuales iban a conformar las piezas del rompecabezas del mecanismo final.

Al tener definidas las posibles piezas a utilizar se continuó con un estudio de factibilidad para desarrollar o implementar cada una de las piezas, debido a que para cada tipo de mecanismo existe una gran variedad, y otra gran variedad que se pueden desarrollar. Por lo cual se estudiaron los requerimientos de los mecanismos seleccionados como son; protección basada en *software* con seguridad específica, protección basada en *software* con seguridad general y mecanismo de control de acceso, para cada uno de estos tipos de mecanismos de seguridad, se procedió a analizar un conjunto de ellos, y seleccionar los más factibles y que además prestarán el nivel de seguridad deseado.

Con esta visión más clara de los posibles mecanismos de seguridad a utilizar, se procedió a profundizar la investigación, referente a la factibilidad de los mismos, debido a que algunos mecanismos requieren, unidades específicas para grabar un CD o de algún otro tipo de equipo especializado, otros requieren de un determinado software de grabación, o de algún medio de almacenamiento con características especificas, etc.

Luego de estudiar la factibilidad de implementación de los mecanismos de seguridad existentes, se estudió la posibilidad de desarrollar mecanismos de seguridad propios, con lo cual se tiene la ventaja de que los algoritmos de los mecanismos no son conocidos, por lo que el nivel de seguridad aumenta.

Al tener más definidos los posibles mecanismos a implementar se procedió a definir exactamente cuáles de los mecanismos factibles se iban a utilizar, para esto se estudió las ventajas y las desventajas de cada uno para, posteriormente, seleccionar aquellos mecanismos que uniéndose

disminuyeran en gran número las debilidades que presentan individualmente, para que al final el nivel de seguridad prestado sea superior al ofrecido por cada una de las piezas por separado.

Posteriormente se seleccionó el orden a desarrollar de cada una de las piezas siguiendo las recomendaciones presentadas por McConell (1998, pág. 162) "En un nivel técnico, hay que asegurarse de que se han tenido en cuenta todas las dependencias técnicas entre los diferentes componentes de un producto", por esto se hizo una selección sobre qué piezas del mecanismo se tenían que desarrollar antes, o después y cuáles en paralelo debido a sus dependencias.

d.- Desarrollo de las etapas.

Como su nombre lo dice, en esta fase se desarrollan cada una de las etapas o piezas definidas en el diseño global, para cada una de ellas se realiza el diseño detallado, la codificación, depuración y pruebas.

Teniendo ya definidas cada una de las etapas o piezas a desarrollar, se realizó un análisis detallado, en donde se estudió la forma de desarrollarlo para que se pudiera combinar con las demás piezas del mecanismo, durante esta etapa se procedió a elegir qué piezas eran necesarias de programar o codificar, y se realizó el diseño de cada una de las mismas, como por ejemplo el estudio del formato de las imágenes gráficas que se deseaban proteger y cómo se podía hacer, o las distintas formas de generar un serial seguro.

Al tener definidas cuáles etapas o piezas había que programar y cuáles sólo se tenían que aplicar al *software* que se deseaba proteger, se procedió a codificar las distintas piezas que eran necesarias, para lo cual se codificaron algoritmos basados en los estudios realizados en el diseño detallado de cada una de las etapas.

e.- Integración de etapas y evaluación del prototipo.

Al finalizar el desarrollo de todas las piezas establecidas anteriormente, se comenzó a unir las diferentes piezas del rompecabezas, de la forma que se diseñó en la fase anterior. El mecanismo de seguridad desarrollado es un prototipo, y presenta una serie de ajustes para permitir su aplicación a un *software* multimedia ya desarrollado, lo que presenta el problema de que el prototipo no se realizó ala par del desarrollo del *software* multimedia, por lo que está implementado de una forma particular por este motivo ya que no se tenía el acceso al código fuente de este *software.* Por lo que se debió crear un programa que realizara una introducción mientras se descifraban las imágenes al iniciar el *software* y que las cifrara nuevamente al salir del mismo.

Durante el proceso de integración de los mecanismos o piezas seleccionadas, se presentó la dificultad de averiguar en qué formato se debían realizar las grabaciones de las multisesiones; para permitir quemar primero una parte del CD y posteriormente el resto, pero desde una imagen del CD multimedia ya protegido con el mecanismo de archivos de gran tamaño; de modo de lograr alcanzar el tamaño de 61 minutos de grabación aproximados para poder superar la capacidad normal del CD y con esta lograr realizar el mecanismo de *overburning* sobre un CD de 80 minutos.

Al terminar de integrar todas las etapas se procedió a la realización de un conjunto de pruebas basadas en un modelo de evaluación de mecanismos de seguridad planteado en este proyecto, que se basa en la evaluación de la calidad durante el desarrollo del producto y en modelos de evaluación que califican la facilidad de uso, la compatibilidad, la versatilidad y la seguridad.

FASE III: Documentación y exposición.

Durante esta fase se realizó toda la documentación necesaria para respaldar el desarrollo del presente proyecto y sus conclusiones, así como todo el material necesario para su exposición ante **el** jurado calificado para su evaluación.

Además, se procedió a crear un manual sobre la utilización de la arquitectura de seguridad planteada, que presenta un conjunto de pasos recomendados a seguir en el desarrollo de un mecanismo de seguridad para *software.*

RESULTADOS

Los resultados obtenido en este proyecto son variados y se pueden enunciar dependiendo de cada una de las fases de desarrollo del proyecto.

FASE I: Investigación Documental.

En esta fase se realizó principalmente una investigación documental basada en Internet, y debido a la cantidad de información recolectada durante la investigación en fuentes bibliográficas, en donde se pudo apreciar la gran variedad de áreas relacionadas con el tema de la seguridad de *software,* se construyó

un mapa mental (ver Apéndice A, Mapa mental de ideas), sobre las distintas áreas relacionadas y sus interrelaciones, con esto se pretende facilitar el entendimiento del material recolectado y de la importancia del conocimiento de un determinado tema para poder comprender con mayor profundidad el tema de la seguridad y la importancia de cada área para el desarrollo de este proyecto.

Otro de los resultados obtenidos durante esta primera fase del proyecto es que se lograron establecer los principales mecanismos de seguridad existentes en el mercado, y basados en esta información se desarrolló una tabla comparativa de los mismos, que permitiera una mejor visión al momento de estudiar las ventajas de uno respecto a los otros y sus requerimientos de aplicación.

En esta tabla comparativa (ver Apéndice B, Clasificación de mecanismos de seguridad para *software* existente), se pueden apreciar los distintos mecanismos de seguridad de *software* existentes en el mercado, con las características de los mismos, y su clasificación de acuerdo al lugar de protección, medio de almacenamiento, y el tiempo de protección, los cuales son los parámetros expuestos por el G.I.I.S (2001), en su modelo de clasificación de los mecanismos de seguridad de *software* referido en el segundo capitulo.

Esta tabla es de gran valor para el proyecto, debido a que la investigación documental relacionada a este tema presenta grandes inconvenientes, por la dificultad de encontrar información sobre los mecanismos de seguridad existentes, debido a que su seguridad se basa en el secreto del algoritmo, por lo que las empresas desarrolladoras cuidan mucho este secreto. Además la investigación documental sirvió como punto clave para el desarrollo del prototipo del mecanismo de seguridad, y para la arquitectura definida en este proyecto.

Durante la investigación documental se realizaron una serie de encuestas a un conjunto de especialistas, tanto en el área de seguridad como en otras áreas relacionadas. Durante estas encuestas se logró acumular la opinión del grupo de especialistas y se realizó una tabla comparativa para poder apreciar sus distintas opiniones. A continuación se presenta dicha tabla. (ver tabla #3: Resultados de las encuestas a especialistas en el área de seguridad o afines).

FASE II: Desarrollo del prototipo del mecanismo antipiratería.

1. Requerimientos de la UCAB

Durante el proceso de desarrollo del prototipo del mecanismo de seguridad, posterior a la conceptualización del *software,* se realizó una de las etapas fundamentales para el desarrollo del prototipo del mecanismo de seguridad adaptado a las necesidades de la U.C.A.B., el levantamiento de la información sobre los requerimientos de la UCAB en cuanto a la seguridad de *software,* para lo cual se realizaron un conjunto de entrevistas estructuradas a distintas organizaciones desarrolladoras, durante esta etapa se realizó una tabla comparativa de las respuestas

Pregunta\Entrevistado	Experto #1	Experto # 2	Experto # 3
Tipos de mecanismo de seguridad antipiratería que conoce.	Marca de agua, mecanismos para proteger derecho de autor de audio, encriptamiento, tarjetas inteligentes.	Algoritmo de encriptamiento	Protección de datos, seguridad de red, Tarjetas inteligentes.
Mecanismos de seguridad que conoce.		Sólo mecanismos de encriptamiento, como DES, etc.	CipherPack
La opinión sobre los mecanismos que conoce.	No existe un mecanismo totalmente seguro, debido a que la información digital es muy difícil de proteger.	Son muy seguro si se desea, pero basan su seguridad en la clave y por lo general no en los algoritmos.	
Qué empresa conoce que desarrollan mecanismos de seguridad.	NA		
Qué mecanismo de evaluación conoce.	Ninguno	Ninguno	Ninguno
De qué manera piensa que un mecanismo de seguridad debe ser probado para demostrar su funcionalidad.	NA		
En qué lugar se puede ampliar la información.	www.ieee.org/organizations/pubs/pub_preview/PROC/87proc07_toc.htm, www.jjtc.com/Steganography		http://www.antipiracyworld.com/cobdown.htm
Qué bibliografía o experto recomienda.	Libro de Fabien A. P. Petitcolas.		

Tabla No. 3. Resultados de las encuestas a especialistas en el área de seguridad o afines

Pregunta\Entrevistado	C.I.C. Héctor Álvarez.	C.A.I. Carolina Marturet	C.T.S Daniel Graterol
Tipos de productos que desarrollan	Multimedia.	Multimedia educativos.	Productos a la medida de la UCAB.
Medio de distribución	CD-ROM.	Ninguno.	Ninguno.
Lenguaje de desarrollo	Macromedia Director, y están empezando a utilizar B/D como Access.		Power Building con Oracle.
Lo que más desean proteger	Fotografías, texto, video, por los derechos de autor de los mismos.		Los datos.
¿Utilizan algún medio de protección?	Ninguno	No, lo máximo que han hecho es colocar un mensaje en sus productos diciendo que no se puede vender o reproducir.	Ninguno para el software, debido a que es para uso interno y a la medida.
¿Han considerado utilizar algún mecanismo de seguridad?	Si, pero no han buscado información sobre el tema.	No, pero si pensaran en venderlo lo más seguro es que sí.	No aplica, no distribuyen.
¿La organización estaría dispuesta en invertir en seguridad?	Si, pero están limitados a los financiamientos que logren obtener para los proyectos.	Si, si lo vendieran.	No aplica.

Tabla No. 4. Resultados encuestas a organizaciones desarrolladoras de *software* en la UCAB

obtenidas. (ver tabla #5: Resultados encuestas a organizaciones desarrolladoras de *software* en la UCAB.)

Con la información recolectada durante las entrevistas realizadas, se obtuvo como resultado, que los principales requerimientos de la UCAB en cuanto a protección de *software* son, la protección de imágenes, videos y textos, debido a los derechos de autor que presentan los mismos. Además de tener como medio de distribución los discos compactos.

Durante la fase de diseño global se desarrolló la arquitectura para un mecanismo de seguridad orientado a la protección del *software,* el cual se presenta a continuación.

2. Arquitectura de seguridad para *software*

La arquitectura de seguridad se fundamenta en que existe un conjunto de elementos que brindan distintos niveles de seguridad, y una buena combinación de ellos pueden brindar un nivel óptimo de seguridad dependiendo de las requerimientos de seguridad que se necesiten. Estos requerimientos se pueden dividir en cuatro categorías básicas como son:

- Durabilidad de la protección.
- Momento de la protección. (Copia, instalación, ejecución)
- Medio de almacenamiento. (Zip, CD, DVD, etc)
- Lugar de la protección. *(Hardware o Software)*

Durabilidad de la protección: como su nombre lo indica se refiere al tiempo que se desea que el *software se* mantenga protegido. Bien se sabe, que en la actualidad no existe algún mecanismo totalmente inviolable, todos ellos son desarrollados y luego de un tiempo se descubre la manera de romper ese mecanismo de seguridad. Se podría pensar de qué sirve entonces proteger algo si se conoce que aún así es vulnerable, esta pregunta se puede contestar poniendo un punto de comparación de la cotidianeidad, por ejemplo, los mecanismos de seguridad que la mayoría de las personas colocan es sus vehículos para que no se los hurten, como se sabe esos mecanismos tampoco son muy seguros o invulnerables, es de conocimiento público que en cuestión de unos pocos minutos se logran romper estos mecanismos, pero de qué sirven entonces, son sencillamente mecanismos de persuasión, con lo cual se logra evitar por un tiempo que personas especializadas logren romper dichos mecanismos, además de persuadir a los ladrones a elegir un objetivo más sencillo como sería un carro sin ninguna protección. Por esto, se habla de durabilidad de la protección en mecanismos de seguridad para *software.*

Momento de la protección: otro factor importante en las requerimientos que se debe tomar en cuenta para la protección de un *software,* es el momento de la protección. Se considera momento de la protección, el instante en que el mecanismo de seguridad actúa para evitar su copia o uso ilegal. Los momentos de protección se clasifican en tres categorías:

- Copia: si se desea que el *software no* sea duplicado sin autorización, se coloca un mecanismo de seguridad que actúe en el momento en que se trate de hacer una copia del mismo, puede ser tan sencillo como evitar que no se copie un archivo en especifico u otras técnicas existentes.

Instalación: en algunos casos no nos importa que se duplique el software, esto se hace generalmente para permitirle a los usuarios realizar copias de seguridad de su *software* legal, como una medida de protección si se daña el original, para estos casos el mecanismo de seguridad se encarga de verificar en el momento de la instalación del *software* si está autorizada y si cumple con los términos de la licencia adquirida, corno por ejemplo el número de máquinas donde se puede instalar o el tiempo de uso, todo depende de la licencia establecida.

- Ejecución: muy utilizado para versiones de *software de* pruebas. En este caso el mecanismo de seguridad se ejecuta en el momento en que comienza la ejecución para verificar si es permitida, como se mencioné anteriormente es muy utilizado para software de prueba que verifica que sólo se use un tiempo determinado y que no permita su ejecución si ya transcurrió el tiempo de prueba, otro ejemplo es el caso en que se solicita que se tenga el CD original en la unidad de CD-ROM.

Medio de almacenamiento: el medio de almacenamiento se refiere al medio físico o electrónico que se utiliza para almacenar y distribuir el *software,* en la actualidad existen diversos medios de almacenamientos de una variada capacidad de almacenamiento y funcionamiento, las principales y más utilizados son los CD y los DVD por su bajo costo y capacidad de almacenamiento, otros de los medios existentes son íos discos flexibles, los zip y los jazz.

Lugar de la protección: se refiere como su nombre lo indica al lugar de la protección, como pueden ser *hardware o software.* En la actualidad existe una gran variedad de mecanismos de seguridad basados en *hardware,* los más famosos son las llaves electrónicas, mientras que en la protección de software para software, son por ejemplo los mecanismos que utilizan clave de acceso, o el conocido Safedisc de C-Dilla y de Macrovision Corporation.

Tomando en cuenta los distintos tipos de requerimiento y la cantidad de recursos destinados a proteger un *software* en especifico, se plantea la arquitectura de seguridad para *software,* que sencillamente trata de establecer una manera de combinar un conjunto de distintas técnicas y mecanismos, para brindar la máxima protección para el *software.* Para ello se explican primero los distintos tipos de mecanismos o técnicas de seguridad utilizados en la actualidad que representan cada una de las capas de la arquitectura.

Control de acceso: el control de acceso es uno de los mecanismos más utilizados en la actualidad, que intenta impedir que usuarios no autorizados utilicen el *software* protegido con este mecanismo. Aunque es muy utilizado, brinda muy poca seguridad debido a que si nos ponemos a analizar, este mecanismo se basa en autentificar a la persona que está utilizando o instalando el *software,* para lo cual le solicita un serial entregado por el fabricante del *software,* y esto no es seguro debido a que los usuarios no mantienen en secreto el serial entregado por el fabricante, ya que no los perjudica de manera directa.

En la actualidad se han desarrollado mejoras en este mecanismo, tratando de buscar una manera de identificar unívocamente al usuario y que el mismo no se lo pueda pasar a otra persona, como por ejemplo el reconocimiento de la voz, la huella digital, el iris de los ojos, pero aún son muy costosos.

Protección basada en *software:* la protección basada en *software* son todos aquellos mecanismos de seguridad que sencillamente son una aplicación que se le agrega al *software* que se desea proteger. En la actualidad *este* es uno de los tipos de mecanismos mas desarrollados, principalmente por el bajo costo que presentan al momento de implementarlo. Aunque su costo de desarrollo es generalmente elevado, aún sigue siendo más económico que otros tipos de mecanismos, como son los basados en *hardware* que se presenta a continuación. Ala vez esta capa se encuentra dividida en dos subcapas, como son los mecanismos basados en *hardware* especificos y generales, como su nombre lo dice se basan en un formato de archivo en especial y los generales en cambio sirven para cualquier tipo de archivo.

Protección basada en *hardware:* este tipo de mecanismos se basa en la utilización de componentes tipo hardware utilizados para proteger un *software* en especifico, los más comunes en la actualidad son las llaves electrónicas, las cuales son utilizadas como medio para poder identificar a las personas no autorizadas, con este mecanismo se ha logrado que aunque se copie el *software sólo* una persona lo pueda utilizar, debido a que si la llave electrónica no está conectada a la computadora, mediante el puerto paralelo o el puerto USB, no se permite su ejecución. Al igual que la protección basada en *software* se divide en dos subcapas, que son la protección basada en *hardware* específica y la general.

Antidebugging se basan en un conjunto de técnicas que impiden que se le aplique un *debugger* al *software* que se desea proteger, esta técnica es

muy importante debido a que la forma más común de detectar el funcionamiento exacto de un mecanismo de seguridad es aplicándole un *debugging* o corrida paso a paso, y de esta forma buscar el punto débil del mismo.

Criptografía: la criptografía es utilizada para encriptar el código de las aplicaciones y/o de los mecanismos de seguridad utilizados. La criptografía aunque se conoce como la ciencia de la comunicación segura, también es utilizada para la protección del *software.* Lo que generalmente se hace, es cifrar o encriptar el código del mecanismo para dificultar que los *cracker* logren leer con facilidad el código del mecanismo.

Requerimiento	Protección software		Protección hardware		Control de Acceso	Criptografía o Antidebugging
	General	Específica	General	Específica		
Proteger imágenes		✓				
Proteger archivos de sonido		✓				
Proteger archivos de texto		✓				
Proteger ejecutable	✓		✓			
Proteger durante instalación	✓		✓	✓	✓	
Proteger durante ejecución	✓	✓	✓	✓	✓	
Proteger durante copia	✓		✓	✓		
Proteger toda la información	✓		✓		✓	
Riesgo de violación alto	✓	✓	✓	✓		✓
Riesgo de violación baja	✓				✓	
Riesgo de violación media		✓		✓		

Tabla No S. Recomendación de uso de mecanismos de seguridad

En la figura #3 (Arquitectura para la protección del *software),* se presenta la relación entre estos tipos de mecanismos y técnicas de seguridad, y como la combinación de ellos pueden hacer más seguro un *software.* Hay que tomar en cuenta que si se opta por no utilizar algunas de estas capas, se disminuye la seguridad prestada por el mismo, pero también disminuye su costo. Todo esto depende de la cantidad de recursos destinados a la protección del *software,* y de los requerimientos del mismo, por lo cual se debe buscar la solución más óptima que satisfaga ambas cosas, pero que se ajuste a la arquitectura.

La arquitectura de seguridad para *software* está conformada por tres capas o tipos de mecanismos de seguridad, entre estas tres capas la seguridad prestada es menor en la capa superior y mayor en la capa inferior, por lo que si se desea un mecanismo más seguro se recomienda utilizar las capas inferiores de la pirámide. Anexas a estas tres capas existen dos capas que recubren a las otras tres, que pueden ser utilizadas para completar o mejorar la seguridad prestada por cada una de ellas, estas son la capa de criptografía y la de *antidebugging.*

Fig. 3. Arquitectura para la protección del software

De estas tres capas la de protección *hardware* es la más costosa como se mencionó, y la menos transparente al usuario, por lo cual es una de las menos utilizadas, pero que a su vez es una de las más seguras en la actualidad, gracias al nivel que han alcanzado, y a la utilización de técnicas criptográficas y *antidebugging.*

Recomendaciones de uso de la arquitectura

La arquitectura de seguridad que se esta planteando, intenta recomendaren que caso es bueno utilizar o no una capa en especifica, lo cual se puede ver en la tabla #5 (Recomendación de uso de mecanismos de seguridad.)

La tabla indica cuáles de las capas se pueden utilizar bajo unos requerimientos dados, si para un requerimiento se recomienda un conjunto de mecanismos queda a consideración del desarrollador cuales de ellos puede usar o desarrollar, siempre teniendo en cuenta el nivel de seguridad deseado y los recursos que se posean, debido a que como por ejemplo los mecanismos basados en *hardwaretienen* un costo muy superior a uno basado en *software,* por esto se requiere de un cuidado al momento del diseño global del mecanismo, además de tener un buen estudio de las personas a que va dirigido el *software y* su capacidad de violar el mecanismo.

3. Prototipo de un mecanismo de seguridad para *software* multimedio

Esta arquitectura sirvió para realizar el diseño global del prototipo del mecanismo de seguridad para *software,* para lo cual se procedió a analizar, basados en los requerimientos, qué piezas del rompecabezas planteado en la arquitectura se tenían que desarrollar, para lo cual se seleccionaron los siguientes tipos de mecanismos, como son:

- Protección *software* para seguridad especifica.
- Protección *software* para seguridad general.
- Control de acceso.

Para cada uno de los distintos mecanismos de seguridad se realizó un estudio de factibilidad para saber cuales se podían utilizar, los mecanismos seleccionados fueron:

Tipo de mecanismos	Mecanismo de seguridad	Implementación	Factibilidad
Protección software para seguridad especifica.	Cifrado imágenes jpg.	Codificar	Si
Protección software para seguridad específica.	Cifrado imágenes tif.	Codificar	Si
Protección software para seguridad específica.	Cifrado de imágenes bmp.	Codificar	No
Protección software para seguridad especifica.	Cifrado de sonido way.	Codificar	No
Protección software para seguridad general.	Cifrado de **archivos** genérica.	Codificar	Si
Protección software para seguridad general.	Dummy Piles.	Aplicar	No
Protección software para seguridad general.	Archivos de gran tamaño Oversize u Overburning.	Aplicar	Si
Protección software para seguridad general_		Aplicar	Si
Protección software para seguridad general_	CD can multisesion.	Aplicar	Si
Protección software para seguridad general_	illegal TOC.	Aplicar	No
¡Protección software para seguridad general.	CD Check	Aplicar	No
Control de acceso_	Seriales seguros en el momento de instalación.	Codificar	Si

Tabla No 6. Mecanismos **de seguridad para** *software* **analizados**

Como se puede apreciar en la tabla #6 (Mecanismos de seguridad para software analizados), los distintos mecanismos evaluados fueron seleccionados dependiendo de su factibilidad de implementación para este proyecto, algunos de las mecanismos sólo se tenían que implementar y otros que codificar, por lo cual se procedió a iniciar la fase de desarrollo de etapas, en donde los algoritmos codificados fueron principalmente los de cifrado de archivos, en el que se buscó la manera de distorsionar las imágenes pero que no se dañara el archivo o perdiera su formato, para esto fue necesario conseguir editores de formato hexadecimal que permitiera analizar los archivos en hexadecimal y buscar una forma de distorsionarlas, se procedió a comparar un conjunto de imágenes del mismo formato hasta encontrar un patrón en hexadecimal, y probar si sólo distorsionaba la imagen o se dañaba, hasta saber como hacerlo, a! encontrarla se procedió a diseñar un algoritmo, así se realizó para los distintos formatos. Sólo las imágenes de formato JPG, y algunas de las imágenes TIF siguen un patrón que permite la distorsión, a las otras imágenes fue necesario realizar un cifrado genérico, que se le puede aplicar a cualquier archivo, para esto también se diseñó un algoritmo específico, desarrollado especialmente para este mecanismo, que cifrara los archivos con una clave.

Para los mecanismos que sólo tienen que aplicarse, se realizaron un conjunto de pruebas hasta lograr implementarlo de forma correcta, otros de estos mecanismos no se lograron implementar debido a dificultades de tipo tanto de *hardware* como de *software,* ya que mucho de ellos requieren de *software* de grabación específicos o de quemadoras de CD con determinadas características. Por ejemplo para poder realizar *overburning es* necesario quemadoras de CD que poseen esta característica, al igual que la técnica de "Illegal TOC", para más detalles de las quemadoras que permiten esta técnica de *overburning* (ver el Apéndice C: Lista de quemadoras con capacidad de *overburning).*

Luego se investigaron los algoritmos para la generación de seriales seguros con dígito de verificación, para lo cual se diseñó un algoritmo propio, que mostrara la posible utilización de este tipo de mecanismos en la seguridad de *software.*

Y por último se realizó el diseño de aquellos programas necesarios para poder unir el *software* ya desarrollado que se deseaba proteger con el mecanismo de seguridad, para lo cual se diseñó un algoritmo que bloqueara las teclas de salida de Windows como Ctr+Alt+Supr y Alt+Tab, además de otro programa que realizar una presentación al programa mientras se descifran los archivos protegidos.

Para proteger el *software* multimedia se seleccionaron siete mecanismos distintos de seguridad, que se muestran en la tabla #7 (Mecanismos de seguridad utilizados.)

Mecanismo de seguridad	Finalidad	Funcionamiento
Cifrado imágenes JPG.	Impedir el copiado de las imágenes JPG sin la utilización del software multimedia protegido.	Modifica el archivo de la imagen JPG en hexadecimal, alterando las propiedades de tamaño, colores y calidad de la imagen, logrando distorsionarla.
Cifrado imágenes TIF.	Impedir el copiado de las imágenes TIF sin la utilización del software multimedia protegido.	Modifica el archivo de la imagen TIF en hexadecimal, alterando las propiedades de tamaño, colores y calidad de la imagen, logrando distorsionarla.
Cifrado de archivos genérico.	Impedir la utilización y la copia de aquellos archivos que no cumplen con el formato de las imágenes JPG o TIF, sin la utilización del software multimedia protegido.	Cifra secciones del archivo a proteger, modificando cada tanto bits del archivo mediante operaciones aritméticas utilizando una clave de cifrado.
Archivos de gran tamaño	Impide la copia del CD protegido al disco duro o a otro disco compacto.	Para este mecanismo se realiza una imagen del CD en formato ISO 9660 Joliete y posteriormente se edita en hexadecimal, buscando los nombres de los archivos a proteger y alterando el tamaño de los mismos, para que ocupen 3,99Gb.
Oversize u Overburning.	Impide la copia del CD protegido a otro disco compacto.	Se quema un poco más de la información que indican la capacidad de los CD's, es decir más de 74min o de 80min, dependiendo del tamaño del CD utilizado, esto solo lo permite una cantidad determinada de quemadoras de CD y algunos software de grabación.
CD con multisesión.	Impide la copia de la información almacenada en el CD, específicamente la segunda multisesión, al disco duro o a otro disco compacto.	Se quema una parte del CD indicando que es multisesión y posteriormente se quema el resto del CD señalando que sé esta continuando. Todo esto se debe realizar en mode11 para permitir agregar la seguridad de archivos de gran tamaño en la segunda multisesión de CD.
Seriales seguros en el momento de instalación.	Protege el instalador del programa multimedia, verificando que la persona que intenta instalar el software multimedia es el propietario del software.	Los seriales generados poseen un digito de validación que indica si el serial es valido o no.

Tabla 6. Mecanismos de seguridad utilizados

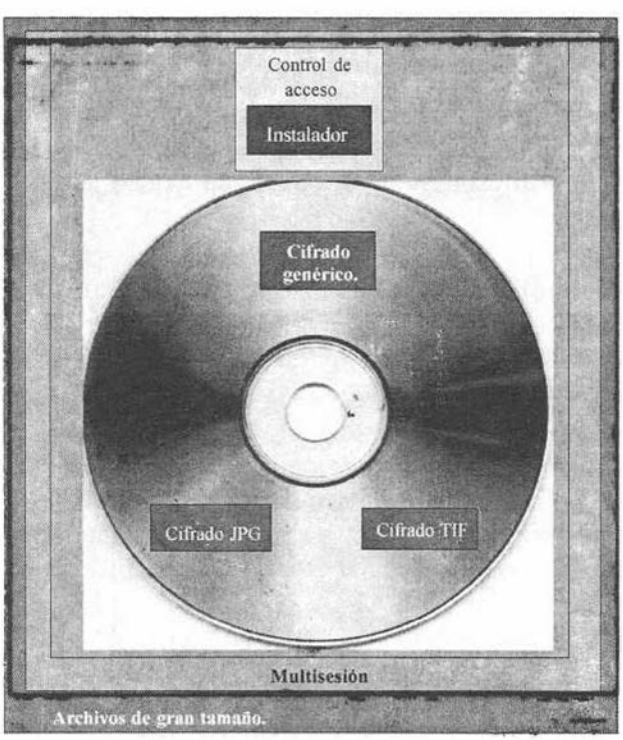

Fig. 4. Esquema del prototipo del mecanismo de seguridad para software multimedia

Quedando el *software* multimedia que se deseaba proteger de la siguiente manera, como lo indica la figura #4 (Esquema del prototipo del mecanismo de seguridad para *software.)*

Para los distintos algoritmos de cifrado codificados se seleccionaron los siguientes lenguajes de programación, como se puede apreciar en la siguiente tabla. (ver tabla #8, Selección del lenguaje de programación para el mecanismo de seguridad.)

Al finalizar de realizar las pruebas individuales se procedió a la unión de las piezas donde se seleccionó que archivos debían ser protegidos, debido al gran tiempo que genera el cifrado de los distintos archivos (ver tabla #9: Tiempo estimado de uso de CPU por algoritmos de cifrado), por lo cual se realizaron ajustes a los algoritmos de cifrado general que se le aplicaba a un grupo de archivos *tif* que no cumplen con el formato del algoritmo de cifrado específico, se modificó el algoritmo de cifrado general para proteger sólo el inicio de los mismos y se unificó en el mismo algoritmo de cifrado de imágenes ***tif,*** reduciendo el tiempo de cifrado de 151 segundos a 2 segundos aproximado.

Mecanismo	Lenguaje	Observaciones
Cifrado imágenes jpg.	Lenguaje C	Seleccionado por su rapidez y compatibilidad.
Cifrado imágenes tif.	Lenguaje C	Seleccionado por su rapidez y compatibilidad, el algoritmo sólo se puede usar con archivos tif específicos.
Cifrado de archivos genérico.	Lenguaje C	Seleccionado por su rapidez y compatibilidad, utilizado para archivos tif que no cumplen el f o rmato.
Seriales seguros en el momento de instalación.	Lenguaje C	Seleccionado por su rapidez y compatibilidad
Presentación de software y bloqueo de tecla de salida.	Director 6	Seleccionado por ser el mismo lenguaje del software **a** proteger
Bloqueo de teclas de escape de Windows (Ctr+Alt+Supr, Alt+Tab)	Visual Basic 6	Seleccionado por las librerías del lenguaje que permiten bloquear con facilidad estas teclas de ,escape.

Tabla 6. Selección del lenguaje de programación para el mecanismo de seguridad

Mecanismo	Número de archivos cifrados	Tiempo de cifrado
Cifrado imágenes jpg.	2157	93 seg.
Cifrado imágenes tif.	259	2 seg.
Cifrado de archivos genérico.	3	**1** seg.
Total	2419	96 seg.

Tabla 9. Tiempo estimado de uso de CPU par algoritmo decifrado

Al finalizar las pruebas individuales y la unión de las mismas, se le integró al *software* seleccionado el mecanismo, en donde se presentó el problema de que no se tenia acceso al código del programa que se estaba protegiendo, por lo que los procesos de descifrado y cifrado de los distintos archivos se realizan todos al inicio y al final del programa respectivamente, con lo que se le agrega un tiempo estimado de dos minutos al inicio y al final del programa, caso que no ocurriría si se realizara por bloques de tema, en donde se descifrarían y cifrarían sólo los archivos necesario para ese tema, haciendo casi insignificativo este tiempo.

Al concluir la codificación de cada uno de los mecanismos o etapas se realizaron un conjunto de pruebas basados en el modelo de evaluación que se presenta a continuación.

4. Modelo de evaluación

Al terminar el desarrollo de los distintos mecanismos o piezas a utilizar se procedió a realizar la evaluación del mecanismo de seguridad para software, con la información recolectada durante la fase de investigación documental sobre los modelos de evaluación de mecanismos de seguridad, se estableció un modelo para evaluar mecanismo de seguridad, este modelo está basado principalmente en la calidad del desarrollo y en modelos de evaluación utilizados por algunas compañías como la NSTL en su NSTL Report (1999).

La seguridad debe ir a la par del desarrollo del *software(ver*figura #5: ¿Dónde comienza la seguridad del *software?),* donde se puede apreciar que la evaluación del mecanismo debe estar integrada en el ciclo de desarrollo del proyecto, debido a que el software y la seguridad deben ir todo el tiempo de la mano para garantizar un *software* seguro, ya que el mecanismo de seguridad así es madurado igual que el *software,* como es expuesto por Ortiz (2001). Para esto existen las normas ISO 9000, que es una Familia de Normas Internacionales sobre las buenas prácticas de Gestión y Aseguramiento de la Calidad.

Además un mecanismo de seguridad debe cumplir con una serie de características que permitan compararlo con otro del mismo tipo, algunas de estas características son según NSTL Report (1999).

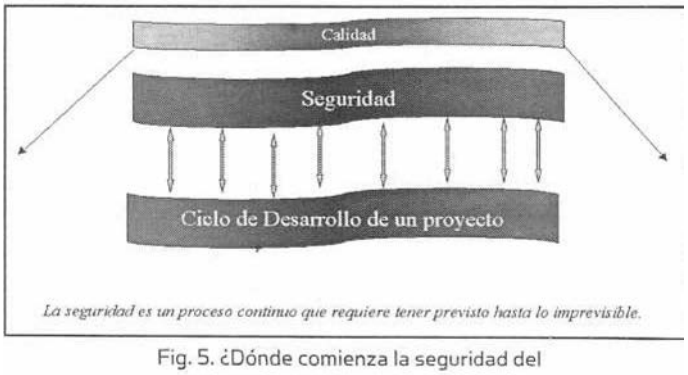

Fig. 5. ¿Dónde comienza la seguridad del *software*?, extraído de Ortiz (2001)

- Seguridad: se mide la efectividad de la seguridad del mecanismo bajo prueba en cuanto a frustración de piratería de *software.*

- Facilidad de uso: compara la facilidad de instalación, configuración, y uso.

E Compatibilidad: se evalúa la compatibilidad del mecanismo de seguridad con el sistema operativo o cualquier dispositivo que utiliza.

- Versatilidad: evalúa la flexibilidad y habilidad de soportar distintos ambientes, plataformas, redes e incluso clientes.

Al finalizar la integración de las distintas etapas del prototipo del mecanismo de seguridad para *software,* se le aplicó el mecanismo de evaluación presentado anteriormente, obteniendo como resultado lo siguiente.

- La seguridad, fue evaluada mediante el modelo de clasificación de seguridad del G.I.I.5, donde un mecanismo es ubicado de acuerdo a las dimensiones, tiempo de protección, lugar de protección y medio de almacenamiento, siendo un mecanismo más seguro al cubrir más dimensiones de este modelo, parlo que sólo faltaría la evaluación mediante un conjunto de expertos en el área de violación de *software,* para medir el tiempo de duración del mecanismos de seguridad, pero no fueron realizadas por la dificultad de encontrar verdaderos expertos que estuvieran dispuesto a realizar estas pruebas.

- La facilidad de uso, se mide en que el mecanismo de seguridad para el cliente o el usuario del *software* es casi transparente, sólo se presenta el problema de que hay un tiempo de inicio de la aplicación de aproximadamente dos minutos y medios, pero esto debido a que es un prototipo y no se tenía acceso al código del *software* que se protegió. Este tiempo de inicio y cierre del programa es causado por el descifrado y cifrado de los archivos protegidos, para lo cual se realizaron pruebas de tiempo de uso del CPU para cada algoritmo de descifrado y cifrado, buscando obtener el menor tiempo, hasta que se logró obtener los resultados mostrados en la tabla #9 (Tiempo estimado de uso de CPU par algoritmos de cifrado)

- La compatibilidad, del mecanismo de seguridad se basó en utilizar sólo herramientas de desarrollo que fueran de estándar abierto, para de esta forma obtener las mismas ventajas de compatibilidad de las herramientas de desarrollo utilizadas durante el desarrollo, con lo cual el *software* protegido no es limitado por el mecanismo.

■ Y la versatilidad se obtiene también de utilizar herramientas de desarrollo con estándares abiertos, con lo cual no se limita la plataforma, debido a que como esta desarrollado con el lenguaje C, sólo seria necesario re-compilarlo en la otra plataforma que se desea utilizar.

CONCLUSIONES

Las principales conclusiones alcanzadas durante el proceso de desarrollo del presente proyecto, en las fases de investigación documental, desarrollo del prototipo del mecanismo antipiratería y la fase de documentación y exposición.

Estas conclusiones son presentadas a continuación:

- En el área de la seguridad de datos existe una gran variedad de información, debido a que basan su seguridad en las claves y no en los algoritmos, por lo que no existe ningún problema en hacer públicos estos algoritmos, mientras que cuando nos adentramos en el mundo de la seguridad de *software,* la búsqueda de información y de material bibliográfico se convierte en dificultosa, debido a que la mayoría de los mecanismos de seguridad destinado a proteger *software* se basan en lo secreto de sus algoritmos o de su forma de aplicación y no en las claves, debido a que las claves son entregadas a los usuarios y ellos no tienen interés en mantener en secreto la clave.

- Muchos de los mecanismos de seguridad se han basado en características especiales del *hardware* de grabación de CD, o en el desarrollo de *software* para la grabación de CD especiales, que permiten explotar características especificas de los mismos, y que los *software* de grabación comunes no son capaces de hacer. Por esto existe una gran variedad de mecanismos de seguridad que aprovechan las características de *hardware* o *software,* y que hacen necesario que para violarlos se requiera de un *hardware o software que* cumplan con esas características, dificultando esta tarea.

- Aunque se piense que la piratería de *software* solo afecta a los desarrolladores, al parecer no es así, sino que además afecta a los compradores y a los creadores. Debido a que para aumentar la seguridad de los productos se ha tenido que invertir en seguridad lo que aumenta el costo del *software,* y además las empresas desarrolladoras de *software* le colocan un precio a sus productos que le asegure la recuperación de su inversión y en este cálculo ya se toma en cuenta que a partir de un número de copias vendidas comienzan a percibir una disminución en las ventas debido a la piratería. Por último el más perjudicado de todos es el creador del producto, este caso se

aprecia más en la piratería de música, en donde las disqueras le ofrecen al artista que las primeras ventas son de la compañía para recuperar el costo y desde un determinado número de copias vendidas en adelantes es del artista, por lo que con la piratería el que más pierde es el artista ya que por lo general no se logra vender lo suficiente como para percibir un buen ingreso.

- Un gran número de organizaciones han hecho grandes esfuerzo en la lucha contra la piratería pero muy a pesar de esto no se ha logrado realizar una disminución en las perdidas por esta índole, esto debido principalmente a que los *crackerstoman* como un reto cualquier nuevo mecanismos de seguridad, por lo que algunas de las compañías han decidido no utilizar mecanismos de seguridad, pero otras han entendido la seguridad del *software* como un mecanismo de persuasión que les permite retrasar las copias ilegales de su *software* en el mercado, por lo que logran obtener mejores ingresos.

- Hay que tener siempre en cuenta que un mecanismo de seguridad es bueno si el costo para violarlo supera las ganancias que se pueden obtener al utilizarlo, debido a que a nadie le interesaría violar un producto si este no le puede hacer recuperar su inversión y brindarle un poco de ganancia.

- No existe información digital segura, ni existirá, debido a que toda información digital es accesible de una u otra forma. La seguridad de la información digital es tan segura como el medio donde se encuentre. Cuando se desarrollen plataformas y sistemas operativos seguros, que permitan brindarle a la información digital un ambiente seguro, este será más seguro.

- La seguridad de *software* está basada principalmente en su manera de implantación y en sus algoritmos; todo mecanismo de seguridad para *software* entre más desconocido sea su funcionamiento y su algoritmo son más seguro.

- El desarrollo o implementación de un mecanismo de seguridad debe ser a la par del desarrollo del *software,* para lograr una perfecta unión entre los mismos y para que ambos maduren a la vez, pasando por las mismas fases de desarrollo y bajo un mismo esquema de calidad, que garantice su buen desarrollo.

- La selección de un mecanismo de seguridad no es sencillo, por lo que es necesario la utilización de un sistema de apoyo o arquitectura que ayude en su diseño.

- En la seguridad de *software* multimedia, los algoritmos de seguridad específicos pueden brindar un nivel de seguridad superior a los mecanismos de seguridad general, debido a que requieren de un mayor conocimiento en el tipo de formato protegido, y no solo conocimientos de cifrado y encriptamiento.

RECOMENDACIONES

Las principales recomendaciones del presente proyecto son:

- Profundizar la investigación de mecanismos de seguridad basado en *hardware,* o en las características de los medios de almacenamiento, en búsqueda de un nuevo mecanismo de seguridad basado en esto, para lo cual se requiere de equipo más especializados como estampadoras de CD que permitan realizar marcas digitales, u otro equipo especializado.

- Se recomienda profundizar el levantamiento de información, realizando un trabajo más integrado con las compañías desarrolladoras de *software* y desarrolladoras de mecanismos de seguridad, tanto a nivel nacional como a nivel internacional, además de aquellas empresas especializadas en la evaluación de este tipo de mecanismos.

- Realizar un estudio de evaluación de la seguridad de los mecanismos presentado en este proyecto, con pruebas de tiempo de violación, realizada por expertos en romper los mecanismos de seguridad para *software* y realizar estudios de mercado para evaluar el tiempo de duración de la seguridad prestada por el mecanismo, en un conjunto de personas representativas de la población que utilizaría el *software.*

REFERENCIAS BIBLIOGRÁFICAS

Aladdin Knowledge Systems Inc. (2001), Productos de seguridad de software.

Consultado Febrero de 2001, de la World Wide Web: http://www.hardlock.com.ar/productos/index.html

Anónimo (1999). Comparative Analysis of Security Dongles NSTL Concise Report.

NSTL Report. Consultado el 18 de Febrero de 2001, de la World Wide Web: http://www.hardlock.co.kr/news/nstl_report_99_HL.pdf

Barrera, M. H. Los Derechos Intelectuales y el CD-Rom. Consultado Febrero de 2001, de la World Wide: http://www2.lahora.com.ec/paginas/rjudi54.htm

BSA (2001). Informe Global sobre Piratería de Software. Consultado el 20 de Julio de 2001, de la World Wide: http://www.bsa.org/resources/2001-05-22.56.pdf

De la Torre, E. & Navarro de Anda, R. (1991). Metodología de la Investigación.

Bibliográfica, archivista y documental. México: McGraw-Hill.

GameCopyWorld, **Copias de Seguridad. Consultado el 12 de Mayo de 2001,** de la World Wide Web: htto://www.vermail.net/defkon/protecciones.html

Garrido, M. (2000), Watermark, Marcas Transparentes?. Consultado el 21 de Abril de 2001, de la World Wide Web:http:11club.idecnet.com/ -modegar/audio/watermark.html

Gawain (2001), Protección de CDs. Consultado el 26 de Marzo de 2001, de la World Wide Web: http:// castellano.hypermart.net/indprothtm

G.I.I.S (2001). MISS: Modelo Inteligente de Seguridad de Software. Caracas, Venezuela: ASOVAC, Convención anual 2000.

Hispamp3 (2000), **Sistemas anticopia en las memorias. Consultado Febrero de 2001,** de la World Wide Web: http://www.hispamp3.com/ noticias/0012/001229 6.shtml

Instituto Tecnológico de la Paz, Consultado Agosto de 2001, de la World Wide: http://www.itlp.edu.mx/publica/tutoriales/produccion1/tema2_1.htm

Maikel (2000). El Disco Compacto o «Compact Disc". Consultado el 21 de Abril de 2001, de la World Wide Web: http://www.fortunecity.es/virtual/hardware/386/cd/ElDiscoCompacto.html

Matout, S. & Moronta, A. (1997). EDTEC: Modelo para el aprendizaje interactive y cooperativo de herramientas de análisis y diseño de sistemas. Venezuela, Universidad Metropolitana.

McConell, S. (1998). Desarrollo y gestión de proyectos informáticos. España: McGraw-Hill.

Oppliger, R. (1998), Sistemas de Autentificación para Seguridad en Redes. Madrid, España: RA-MA Editorial

Ortiz, L. (2001). Seguridad de sistemas informáticos. Consideraciones a tornar para implementar seguridad en un desarrollo de software. Material no publicado. Caracas, Venezuela: UCAB.

Pagua, **G. (1996). Hacker Underground.** Consultado el 01 de Agosto de 2001, de la World Wide Web: **http://www.lared.com.ve/archivo/hacker5.html**

Piattini, M.G., Calvo-Manzano, J.A., Cervera J. & Fernández L. (2000). Análisis y diseño detallado de Aplicaciones Informáticas de Gestión. España, Madrid: RA-MA Editorial.

Pobladores (2000). CD's con multisesión. Consultado el 18 de Febrero de 2001, de la World Wide Web: http://www.pobladores.com/territorios/informatica/ copiarCDs/pagina/1

StegoArchive.com (2000). What is Steganography?. Consultado el 21 de Abril de 2001, de la World Wide Web: **http://www.cl.cam.ac.uk/-tapp2/ steganography/index.html**