UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN

“MARCAS DE AGUA EN AUDIO”

T E S I S

QUE PARA OBTENER EL TÍTULODE:

INGENIERO EN COMPUTACIÓN

P R E S E N T A:

MISAEL DELGADO SAUCEDO

DIRECTOR DE TESIS:
MAT. LUIS RAMÍREZ FLORES

MÉXICO **2011**

Universidad Nacional Autónoma de México
UNAM
Biblioteca Central
Dirección General de Bibliotecas de la UNAM

UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

A México por ser el mejor país para vivir, me siento muy orgulloso de ser mexicano.

A la Universidad Autónoma de México y a la Facultad de Estudios Superiores Aragón por su calidad de estudio y por formarme profesionalmente.

A Francisca Bautista Alcántara. Por tu entereza, sabiduría y cariño. La guía de mi familia. Te amo abuelita.

A José Luis Delgado Martínez. Porque eres la persona más honorable y trabajadora que he conocido, porque me enseñaste la ciencia, por tu amor y confianza. Te amo papá.

A Martha Saucedo Alcántara. Por tu amor y comprensión, por tus ganas de vivir y de sonreír, nunca dejes de sonreír, Te amo mamá.

A Luis Ommar Delgado Saucedo. Porque eres mi ejemplo y mi amigo, por tus consejos y tu humor, gracias por hacerlos cada día. Te amo hermano.

A Ana Karen delgado Saucedo. Porque con tan sólo verte me haces feliz, porque eres mi hermanita y mi amiga, porque con tus metas me has enseñado como lograr las mías. Te amo hermana.

A mi familia. Porque cada fin de semana me enseñan el valor y el amor de estar juntos y de apoyarnos. Los amo tíos, tías, primos, primas y sobrinos.

A Luis Ramírez Flores. Por tus consejos, vivencias y el gran apoyo que me ofreciste. Te quiero amigo.

A mis amigos de la universidad. Aldo, Isaac, Norberto, Bernardo, Miguel Ángel, Víctor, José, Isaías, Edgar, Daniel, Sonia, Elizabeth, Rocío, Paola, porque con su amistad y alegría me enseñaron a quererlos. Gracias amigos.

A mis amigos. Armando, Enrique, Francisco, Juan Carlos, Isaac, Rubén, Héctor, Miguel, Manuel, Luis, Edgar, Jennifer, Cecilia, Sonia, por ser parte de mi vida, por su gran apoyo y su amistad. Gracias amigos.

ÍNDICE GENERAL

RESUMEN

INTRODUCCIÓN

CAPÍTULO 1. ORIGENES DE LA ESCRITURA

1.1 La protoescritura.

1.2 La escritura cuneiforme

1.3 Los jeroglíficos egipcios

1.4 El alfabeto

1.5 Sistemas de escritura

CAPÍTULO 2. ESTEGANOGRAFÍA

2.1 Evidencias en la historia de la esteganografía

2.2 Una mirada detallada a la esteganografía

2.3 Codificando mensajes secretos en texto

2.4 Codificando mensajes secretos en imágenes

2.5 Codificación LSB

2.6 Codificando mensajes secretos en audio

CAPÍTULO 3. MARCAS DE AGUA

3.1 Áreas de aplicación

3.2 Áreas de investigación

CAPÍTULO 4. MARCAS DE AGUA EN AUDIO

4.1 Enmascaramiento frecuencial

4.2 Enmascaramiento temporal

4.3 Umbral de enmascaramiento

4.4 Modelo general de marca de agua digital

4.5 Decodificación y detección

4.6 Seleccionando algoritmos de audio de marca de agua

CAPÍTULO 5. APLICACIÓN DE MARCA DE AGUA CON EL MÉTODO "BIT MENOS SIGNIFICATIVO" (LSB)

5.1 Estructura de un archivo WAV

5.2 Extracción de datos del archivo WAV

5.3 Encriptación del mensaje con AES

5.4 Generación de números pseudo – aleatorios

5.5 Inserción de la marca de agua

5.5.1 Guardando los datos con la marca de agua en un archivo WAV

5.6 Lectura de la marca de agua

CAPÍTULO 6. UN CASO DE USO DE MARCA DE AGUA EN AUDIO

CONCLUSIONES

BIBLIOGRAFÍA

RESUMEN

El gran crecimiento de las comunicaciones ha facilitado el intercambio de archivos multimedia de manera casi instantánea, y esto ha conllevado a la necesidad de protegerlos, contra las copias y distribuciones ilegales. Actualmente la tecnología tiene el reto de brindar una solución integral a este hecho, pero no esta sola, ya que existen técnicas que se pueden aplicar para minimizar las vulnerabilidades de estos archivos en las comunicaciones del día de hoy.

INTRODUCCIÓN

En el primer capítulo muestra el invento más importante que ha tenido la humanidad, que es la escritura, y su desarrollo hasta convertirse en un alfabeto, esto es porque desde que inició la escritura la humanidad se ha desarrollado rápidamente, ya que los conocimientos no se perdían de generación en generación. Este es un paso muy importante en ocultar información, ya que desde la antigüedad hasta el día de hoy, no todos tienen acceso a cierta información.

Con lo anterior, el capítulo 2 muestra la esteganografía y sus técnicas usadas para la ocultación de información. Este capitulo da una visión de cómo inició la esteganografía y como eran sus técnicas, hasta de cómo hoy en día se usa y en que medios digitales se puede aplicar.

El capítulo 3 da una visión general de lo que es una marca de agua, sus técnicas de uso y en que medios digitales se aplican. También muestra lo que puede proteger, es decir, desde los derechos de autor hasta la detección de manipulación.

En el capítulo 4 muestra de lo que se trata esta tesis, la diferencia que existe entre la esteganografía y una marca de agua. También las técnicas y los métodos que se pueden aplicar a los archivos de audio.

En el capítulo 5 muestra una aplicación que utiliza el método del bit menos significativo (LSB, Least Significant Bit), para dar una visión de cómo se aplica y como se desarrolla este método utilizando el lenguaje PHP.

En el último capítulo, da un caso de uso de una marca de agua en audio, de cómo se puede aplicar en una caso que las compañías disqueras sufren hoy en día, que es el de las copias ilegales y su distribución.

CAPÍTULO 1

Orígenes de la escritura

"La escritura es la pintura de la voz"

François Voltaire

La liberación de las manos de las tareas de locomoción no sólo hizo posible la fabricación de herramientas, utensilios y la expansión del cerebro, sino que permitió observar de manera distinta, la disposición de los objetos que aparecen en el campo visual. La mano se convierte en un objeto más en dicho campo, pero con la diferencia de que se tiene sobre ella un control estricto, hasta que es capaz de llegar a producir una mímica sobre un objeto inexistente: así se crea la comunicación gestual simbólica.

Con la comunicación verbal se libera el sentido de la vista para transmitir más información. Escuchar la propia voz, significó para el homínido[1], un logro más sorprendente que ver su mano dibujando un gesto. El homínido que inventó el lenguaje, entendido como intercambio intencional de símbolos acústicos abstractos y no meros gestos, permitió, el avance de la cultura como fenómeno propio de la especie humana.

La necesidad que tiene el ser humano de comunicarse, generó el lenguaje gestual, el lenguaje hablado y el lenguaje mediante signos gráficos, hasta constituir un sistema de comunicación que está basado en símbolos gráficos convencionales, denominado escritura.

El nacimiento de la escritura constituye una hazaña tan revolucionaria como el dominio del fuego y el desarrollo de la agricultura, pues, al igual que estas otras dos, transformó profundamente la existencia humana. Gracias a la escritura, los seres humanos pudieron coordinar sus actividades con las de otros que vivían a gran distancia de ellos. Por otra parte, hizo posible la formación de sociedades

[1] Individuo perteneciente al orden de los primates superiores, cuya especie superviviente es la humana.

mucho mayores y más complejas que las conocidas hasta entonces: ciudades-estado, reinos, imperios. El dominio de la escritura permitió a los hombres desarrollar ideas y realizar cálculos mucho más complicados, abriendo consecuentemente el camino a la matemática y a la ciencia.

1.1 La Protoescritura[2]

Los investigadores se les hace muy complicado tener que interpretar ciertos símbolos encontrados en cuevas de la Era Glacial. ¿Qué puede significar, por ejemplo, el dibujo de una mano rodeada de círculos? Realmente no se sabe. Lo que sí sabemos es que a eso no se le puede llamar aún escritura. Es por eso que a estos símbolos les llamamos protoescrituras, debido a que se encuentran en el umbral de la escritura plena. Los primeros síntomas de escritura (las protoescrituras) surgieron ante la necesidad de contar mercancías, jornaleros, ganancias; o de contar días y ciclos lunares para diseñar calendarios.

Una de las primeras manifestaciones la descubrimos en la Era Glacial donde los hombres realizaban muescas en huesos con materiales diversos. Alexander Marshack[3], un escritor científico, advirtió en una fotografía de un hueso de 8.500 años descubierto en un yacimiento arqueológico del África Central, una docena de grupos de rayas, incluyendo de 3 a 21 rayas cada uno. Luego de profundizar sus investigaciones llegó a la conclusión de que las mismas podrían corresponder a los días del ciclo de la Luna.

Figura 1.1. Hueso arqueológico de África central.

[2] Del vocablo “Proto” que significa “antes”; antes de la escritura.
[3] Director de investigaciones del Museo Peabody de Etnología y Arqueología de la universidad de Harvard en 1963.

Independientemente de la razón, lo cierto es que tales marcas son el resultado de un sistema para almacenar información. Sin duda, esta información sólo sería comprensible para quien la registraba o, como máximo, para unos escasos allegados. Pero, incluso así, las marcas inscritas en los huesos constituyen lo que algunos especialistas consideran como la primera etapa de la protoescritura: el recurso mnemotécnico, es decir, el recurso destinado a auxiliar a la memoria para recordar algo.

Algo parecido las usaron las Haciendas Inglesas entre el año 1100 y 1834 a través de las “tarjas”, tablillas de madera en las que, además de anotaciones de las cantidades en cuestión escritas en su superficie, se realizaban muescas que, en función de su tamaño y grosor, reflejaban las cantidades escritas.

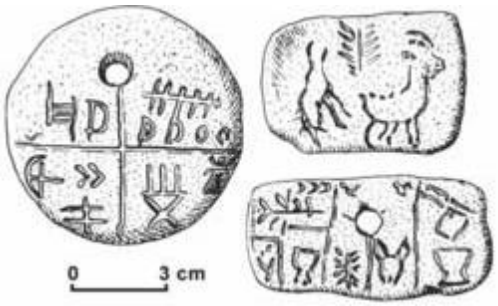

Figura 1.2. Las tabletas de Tărtăria (actual Rumanía), es ejemplo de protoescritura (5500 a. de N.E.)

Otro tipo distinto es la de los “quipus” incas, cuerdas donde se realizaban nudos para representar cantidades.

Pero quizá las muestras más importantes de protoescritura sean las de tipo económico, realizadas en tablillas de arcilla. Este tipo de protoescritura tuvo su antecedente en las “fichas” de arcilla usadas entre el 8000 y el 1500 a. de N.E., y que servían como apoyo a la mnemotecnia humana en transacciones económicas complejas, pues permitían operar aritméticamente con grandes cantidades. Realmente esto no es protoescritura, pero pronto estas fichas fueron introducidas en el interior de bolas huecas hechas también de arcilla que, a modo de sobre

esférico, albergaba un número de fichas inscrito en la superficie, y al mismo tiempo hacían más difícil la falsificación de las fichas, ya que ahora debían ir en el interior de las bolas. Esas inscripciones talladas en la superficie sí pueden considerarse protoescrituras.

1.2 La escritura cuneiforme[4]

La escritura apareció hace poco más de 5.000 años. Sin embargo sus raíces, como de tantos otros inventos, se hunden en un pasado mucho más lejano. El hombre llegó a la escritura tras lentas etapas anteriores: el desarrollo del lenguaje; el descubrimiento de la representación mediante imágenes; la necesidad de reforzar la memoria almacenando información; el darse cuenta de que se podían usar tales imágenes para satisfacer esta necesidad; y por último, el difícil proceso de ensayo y error para adaptar las imágenes a la representación de los sonidos del lenguaje.

Antes del año 3000 a. de N.E., en las orillas de los ríos Tigris y Éufrates (Mesopotamia), se asentaban poblaciones de campesinos, que vivían de su agricultura y ganadería en las tierras de Acad y Sumer. De este modo de vida, surgió la necesidad de llevar una contabilidad de las cabezas de ganado y de los productos de la agricultura, esto se llevaba acabo en los templos. Así, en el templo Sumerio de Uruk, se encontraron uno de los hallazgos más antiguos en materia de escritura. Estaba escrito en caracteres cuneiformes. La escritura cuneiforme desciende directamente del más antiguo sistema de escritura conocido, la escritura pictográfica, probablemente inventada por los sumerios en Mesopotamia hacia el año 3100 a. de N.E.

[4] Denominación relacionada a la forma que tenían los mismos, de la palabra latina cuneus, "cuña".

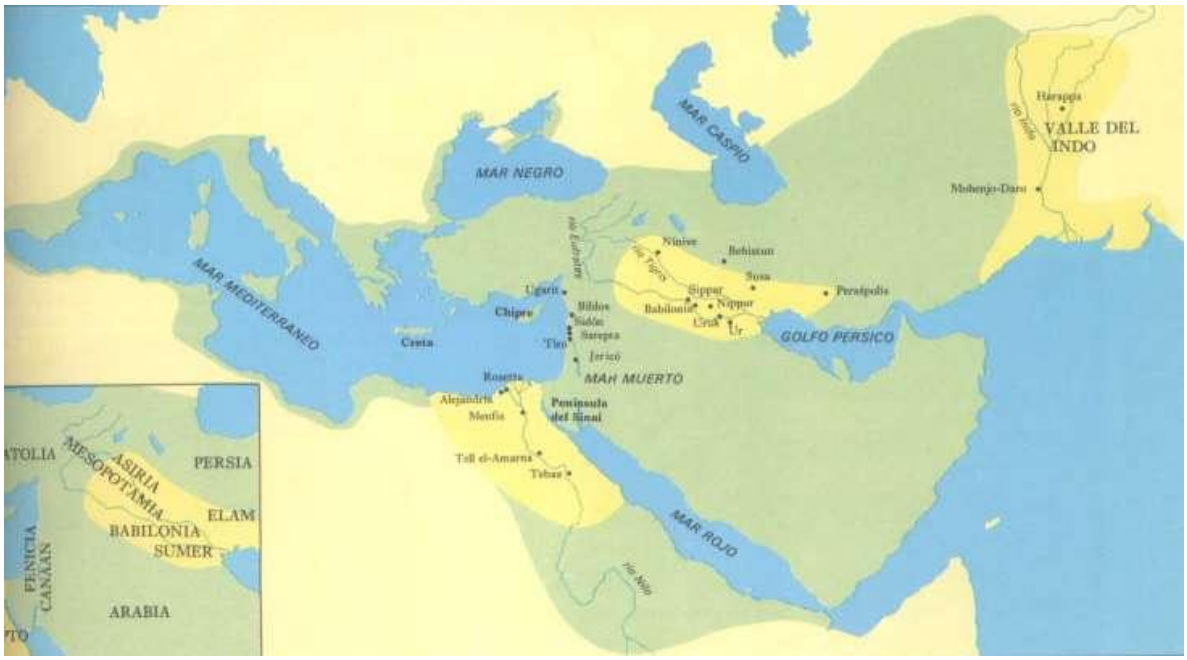

Figura 1.3. Este mapa refleja el impacto cultural que la escritura produjo, basado en su gran sencillez y en la facilidad de su aprendizaje.

En el segundo milenio antes de nuestra era, la enseñanza de la lectura y la escritura estaban circunscriptas a pequeños grupos de población (en amarillo más intenso en la figura 1.1) que se concentraban en las riveras del Nilo, del Indo, del Tigris y el Éufrates. Hacia el 400 antes de nuestra era, cuando ya el alfabeto fenicio y varios otros se habían desarrollado completamente, la escritura se había difundido por una zona (en color verde de la figura 1.1) que cubría no sólo el Próximo Oriente sino también las tierras que bordeaban el Mediterráneo. Las tierras y civilizaciones que promovieron el nacimiento y desarrollo de la escritura se localizan en el mapa del margen inferior izquierdo.

Los símbolos cuneiformes evolucionaron desde el pictograma[5], pasando por el ideograma[6], hacia el silabario[7], pero nunca se llegó a formar un alfabeto, hecho que sólo consiguieron las escrituras Ugras y la Persa antigua.

[5] Es un signo que representa esquemáticamente un objeto real.
[6] Es una representación gráfica de una idea o palabra.
[7] Es un conjunto de caracteres o símbolos que representan (o aproximan) sílabas que forman las palabras.

Los escritos cuneiformes se realizaban en tablillas, principalmente de arcilla fresca, pero también se han encontrado inscripciones en piedra e incluso en metales. No se tiene constancia de la materia del instrumental utilizado para fijar los signos en las tablillas, pero se cree que los punzones estaban hechos con cañas o con madera y que eran de tres clases: uno triangular para formar las cuñas, otro de punta hueca para hacer los clavos y un tercero de punta redonda para marcar cifras.

Figura 1.4. Las tablillas cuneiformes estaban escritas por las dos caras. Solían dividirse en columnas, o en líneas con trazos verticales u horizontales.

La primitiva escritura cuneiforme se realizaba de arriba a abajo, pero posteriormente rotó 90 grados y se convirtió en una dirección de izquierda a derecha. Como hemos dicho, los acadios[8] heredaron éste sistema de los sumerios y, obviamente, al hacerlo propio, lo modificaron dando lugar a varias lecturas de una misma escritura cuneiforme y, por lo tanto, a problemas de transcripción.

Los primeros descubrimientos de escrituras cuneiformes tuvieron lugar en las ruinas de la ciudad de Persépolis (hoy Irán), pero esos investigadores jamás sospecharon el significado de las cuñas. Fue en el año 1621 cuando Pietro Della Valle, un viajero italiano, dio cuenta de una inscripción de 413 líneas, hallada en una pared en las montañas de Behistun (oeste de Persia). En 1674, Jean Chardin agrupó algunos signos cuneiformes y descubrió que las inscripciones se componían de series de tres formas paralelas.

[8] Fue un gran reino de Mesopotamia formado a partir de las conquistas de Sargón de Acad.

Así, el desciframiento de los signos en la montaña de Behistun fue en continuo progreso, hasta que Carter Niebuhr descubrió que las tres formas paralelas que veía ante sí, no era más que un mismo texto escrito en tres tipos de escrituras diferentes, aunque por el momento desconocidas, y en 1777 publicó la auténtica transcripción de la roca de Behistun: Se trataba de la inscripción trilingüe de Darío I, Rey de Persia, escrita en caracteres cuneiformes de tres idiomas: persa, elamita y babilonio.

Figura 1.5. Tumba de Darío I (Naqs-i-Rustem, Persépolis)

Figura 1.6. Tumba de Ciro el Grande (Pasargada, Persia).

Una vez desarrollado completamente, el sistema cuneiforme posee más de 600 signos. Casi la mitad se emplearon como ideogramas o como sílabas, los restantes sólo fueron ideogramas. Algunos signos sirvieron como determinantes, conocidos por determinativos, que indicaban la clase a la que pertenecía la palabra (del tipo hombre, árbol, piedra). A lo largo de su existencia, el sistema era una mezcla de ideogramas y sílabas. Cuando se aplicaba a una lengua diferente, los ideogramas se podían emplear, porque se entendían al representar objetos.

Las transcripciones de la escritura cuneiforme nos han dado cuenta de todo lo que hoy conocemos sobre Asiria, Babilonia y el antiguo Oriente Próximo.

Figura 1.7. Estela de Hammurabi. Este relieve relata la presentación del rey ante el Dios solar Shama. Sobre ella está grabado el famoso Código de Hammurabi, cuya mitología escrita ha arrojado luz sobre la vida religiosa de la antigua Siria y ha obligado a reinterpretar ciertos aspectos de la Biblia.

Fecha	Evolución de la escritura pictográfica hasta la cuneiforme					
Significado	kú Comer	šah Cerdo	mušen Ave	gi Caña	šag Cabeza	kin Huerto
3000 a.C.						
2400 a.C.						
650 a.C.						

Tabla 1.1. Evolución de la escritura pictográfica

Un curioso ejemplo de una tablilla legible es la que se muestra en la Figura 1.8, procedente de Uruk. Se trata de una tablilla de contabilidad en la que vemos muchos cuadros en el anverso. Pues bien, dentro de cada cuadro, los

semicírculos representan números y los signos aislados refieren a nombres de personas.

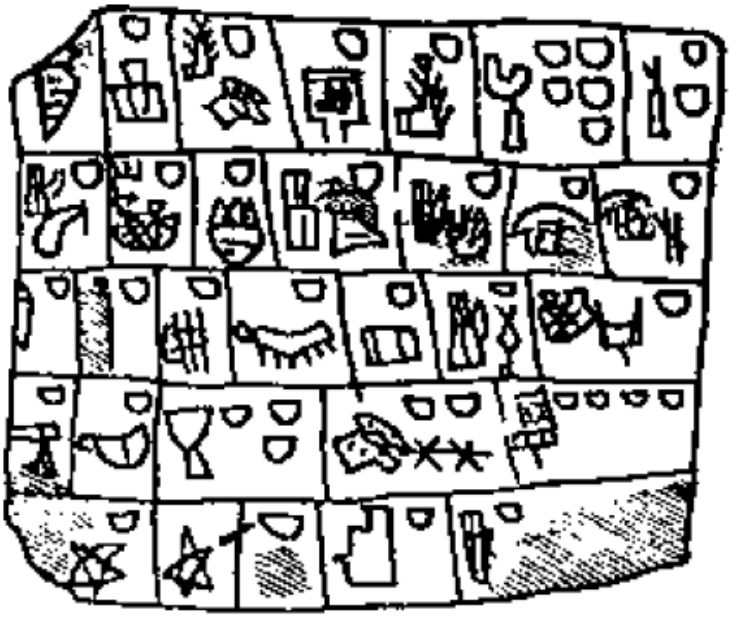

Figura 1.8. Tablilla de Uruk (Frente).

En el reverso, podemos leer la mercancía que se envía. Aquí se lee claramente “54 buey vaca”, es decir, “54 bueyes y vacas” o “54 reses.”

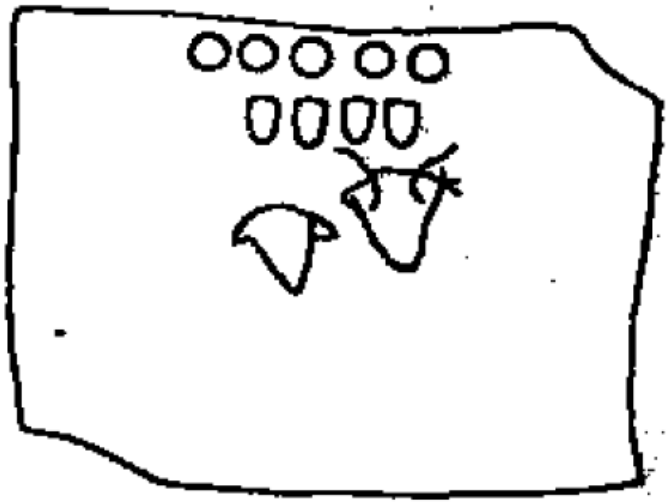

Figura 1.9 Tablilla de Uruk (Reverso).

Esta otra tablilla, también de Uruk, representa una transacción de cebada: Los “capuchones” y los círculos negros arriba, expresan la cantidad de productos: 235.000 litros.

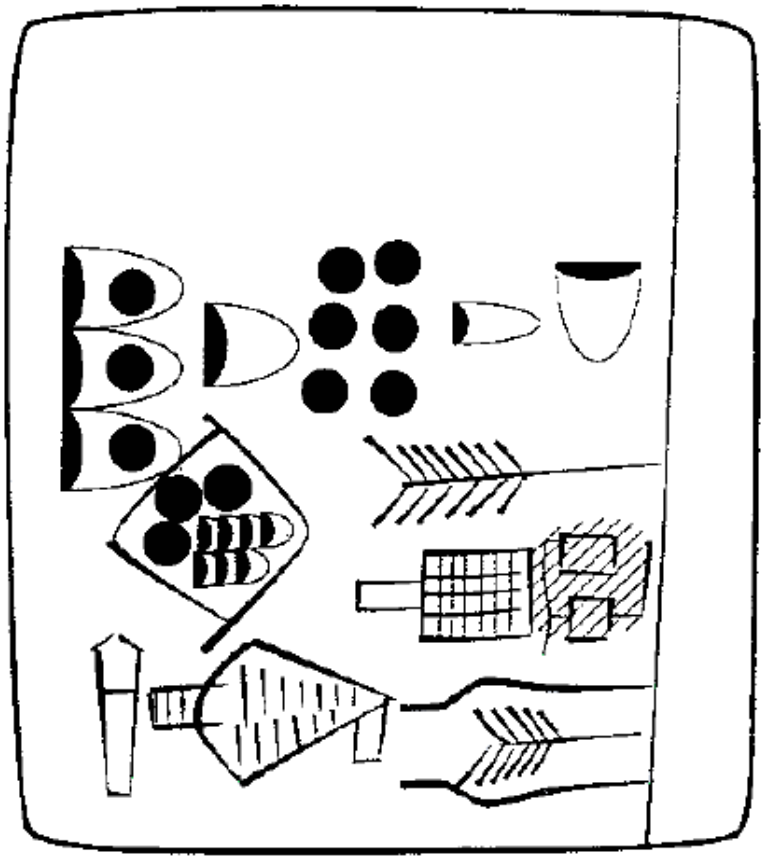

Figura 1.10. Tablilla de Uruk.

El “rombo” del centro-izquierda indica el periodo de tiempo que abarca la contabilidad: 37 meses. La espiga es el producto: cebada. Debajo de la espiga está el signo que representa la función del documento. El símbolo de abajo-izquierda es el nombre del funcionario: Kushim, y la “espiga encerrada” a su lado indica el uso de la cebada.

Con el tiempo se tendió a simplificar la escritura y a reducir el número de signos, sobre todo los polisémicos o polifónicos, para evitar la ambigüedad, con lo que se daba el primer paso para establecer el alfabeto, en el que cada signo representa un sonido, cosa que nunca había ocurrido en el sistema cuneiforme.

1.3 Los jeroglíficos Egipcios.

El desarrollo de la escritura no se dio cuando una imagen empezó a usarse para representar un objeto determinado, sino el sonido correspondiente a su nombre. Es la etapa de la escritura jeroglífica, así llamada por analogía con los jeroglíficos modernos: por ejemplo, la imagen de un "sol" y la de un "dado" significan "soldado". Mediante este sistema, los signos pictográficos empezaron a ser signos fonéticos.

Los jeroglíficos egipcios no parecen haber evolucionado a lo largo de los siglos: se encuentran de repente en el año 3100 a. de N.E., en la antesala del Egipto Dinástico, y progresará poco a lo largo de su existencia. En general, la cultura y la escritura egipcia despertaron el interés y la admiración de las civilizaciones posteriores, pero no fue hasta el Renacimiento donde prendió la verdadera llama decodificadora.

Es posible que la idea de escritura fuera importada por los egipcios desde Mesopotamia, a partir de la invención de la escritura cuneiforme. A partir de la escritura egipcia se desarrollaron simultáneamente a ésta, dos tipos de escritura cursiva: la hierática (usada por los sacerdotes) y la demótica (de uso documental). Los jeroglíficos podían escribirse tanto de izquierda a derecha como de derecha a izquierda, aunque solían adoptar la última dirección en la mayoría de los casos, excepto cuando por simetría convenía usar ambas (por ejemplo, en inscripciones idénticas flanqueando puertas).

La escritura jeroglífica egipcia consta de:

- **Signos monoconsonánticos:** los 24 signos consonánticos que reciben el nombre de “alfabeto”.
- **Signos biconsonánticos:** que representan sonidos como /ms/, /mr/, /sw/,…
- **Signos triconsonánticos:** que representan sonidos como /ndm/, /htp/,…

- **Complementaciones fonéticas:** o añadidos de uno o varios signos monoconsonánticos, colocados al final de la palabra, para que aclaren la pronunciación de ésta.
- **Determinativos:** logogramas que se suman al final de los fonogramas para aclarar el significado de una palabra en su contexto concreto. La mayoría suelen ser de naturaleza pictográfica.

Lo que nos ha llegado de escritura egipcia es fundamentalmente inscripciones en piedra y loza, y la escritura sobre papiro:

Lo escrito sobre piedra y loza es generalmente referente a nombres de faraones y títulos reales; de lo cual lo más destacado es quizá la piedra Rosetta, conservada en el Museo Egipcio de Londres. De lo escrito sobre papiro conservamos mucho menos, puesto que no ha sobrevivido al paso de los años (no tampoco a los escrutinios de las bibliotecas donde se archivaban). El papiro era fruto de la unión de finísimas tajadas de caña, pegadas con su propio jugo. Un cómodo material, más fácil de manipular, que ha valido para escribir "el Libro de los Muertos", quizá la obra más destacada en este soporte, que describe las ceremonias y ritos que han de seguirse para que el difunto alcance la vida eterna.

1.4 El Alfabeto[9]

Frente a los sistemas de escritura antes expuestos, la creación del alfabeto supuso una innovación de consecuencias formidables para el desarrollo de las escrituras y de la cultura misma, una auténtica revolución dentro de la propia revolución que había sido el nacimiento de la escritura.

El alfabeto trata de representar cada sonido por medio de un solo signo, lo que se consigue pocas veces, excepción hecha del coreano (que es el más perfecto) y,

[9] Del griego, formada a partir de *alpha* y *beta,* nombre de las dos primeras letras de su abecedario. El alfabeto es una serie de signos escritos, que cada uno representa un sonido o más de uno que se combinan para formar todas las palabras posibles de una lengua dada.

en menor grado, de los silabarios japoneses. Los alfabetos son algo distinto a los silabarios, pictogramas e ideogramas. En un silabario un solo signo representa una sílaba (secuencia de fonemas, entre dos y cuatro, que se emiten sin pausa). Por ejemplo, el japonés posee dos silabarios completos —el *hiragana* y el *katakana*— inventados para complementar los caracteres que poseían de origen chino. Un sistema pictográfico representa por medio de dibujos los objetos que así lo permiten, por ejemplo, el dibujo de un sol significa la palabra *sol.* Un sistema ideográfico emplea la combinación de varios pictogramas para representar lo que no se puede dibujar, como las ideas y los verbos de significación abstracta. Así si se combinan los pictogramas chinos *sol* y *árbol* representan la palabra del punto cardinal *Este.* Casi todos los alfabetos poseen entre veinte y treinta signos, aunque el rokotas, de las islas Salomón, sólo contiene once letras, mientras que el khmer cuenta nada menos que con setenta y cuatro letras.

Los primeros sistemas de escritura son de carácter pictográfico, ideográfico o una combinación de los dos; entre éstos están la escritura cuneiforme de los babilonios y los asirios, la escritura jeroglífica de los egipcios, los símbolos de la escritura china, japonesa y los pictogramas de los mayas. Lo que distingue a estos sistemas de un silabario o de un alfabeto es que el signo deja de representar un objeto o una idea y pasa a representar un sonido. Normalmente, el sonido es el sonido inicial de la palabra hablada indicada por el pictograma original. Así en el semítico temprano, un pictograma que representaba una *casa,* pasó a ser la escritura de la *b,* primera letra de la palabra *beth* que en este idioma es como se decía *casa.* El símbolo primero significó *casa,* luego la idea del sonido *b* y más tarde es la letra *b,* tal y como ha llegado al alfabeto español.

El alfabeto es otra interrogante que se abre inevitablemente en el tema de la escritura. Sabemos que los griegos fueron quienes introdujeron la idea del alfabeto pero, ¿Fueron ellos realmente los pioneros?, ¿Por qué era tan necesaria la invención del alfabeto?, ¿Acaso para agilizar las operaciones comerciales? De ser así, ¿Por qué no hay indicios de contabilidad en las primitivas escrituras alfabéticas griegas?

Los primeros indicios de escritura alfabética fueron hallados en antiguas minas del Sinaí, que habían pertenecido a los egipcios. Petrie, el autor del hallazgo, observó que la escritura, por su pequeño número de caracteres, parecía ser alfabética, y que plasmaba el idioma semítico.

¿Qué hacía la escritura semítica en estas minas egipcias? Como se supo más tarde, los egipcios se habían valido de los cananeos como esclavos para explotar dichos yacimientos, por lo que es fácil deducir que los mineros habían aprendido el sistema en Canaán antes de ser apresados. Así, todo apuntaba a que la escritura alfabética había nacido en Canaán, a partir de escrituras protocananeas semi-pictográficas: las letras tomaban el nombre del pictograma que representaban antaño (la primera letra, que tiempo atrás se asemejaba a un buey, se llamaba “aleph”- buey -). Siendo los cananeos, habitantes de una región de paso para egipcios, babilonios, hititas y cretenses; y siendo, como eran, comerciantes natos; parece lógico que necesitasen valerse de un sistema rápido, sencillo y sin ambigüedades.

Con el tiempo, la idea de escritura se fue contagiando a otras regiones. En el siglo XIV a. de N.E., los habitantes de Ugarit, mercaderes en su mayoría, persuadidos por el descubrimiento cananeo, adoptaron el cómodo sistema alfabético para transcribir su idioma, con un total de 30 signos cuneiformes. Esta forma de escritura desapareció por la crisis de Ugarit en el 1200 a. de N.E. Tras otras invenciones como el Lineal A (sistema de escritura silábico que tuvo un lapso de uso que va desde el siglo XVIII a. de N.E. al XV a. de N.E., donde el sentido de la escritura es horizontal de izquierda a derecha) o la escritura pseudo-jeroglífica. La idea del alfabeto parece resurgir de nuevo en zonas de Israel. Quienes rescataran el sistema alfabético del desuso serán los fenicios, comerciantes viajeros, que a partir del siglo XI a. de N.E. Comienzan a usar una escritura alfabética, inspirados en los antiguos cananeos. Contaba con 22 caracteres, ninguno de ellos vocálico.

Más tarde, los griegos comerciantes apreciaron la comodidad que podía permitir un sistema alfabético: es por eso que tomaron los caracteres fenicios con un

nombre aproximado al de éstos (“alfa” en lugar de “aleph”; “beta” en lugar de “beth”). Además, convirtieron cinco consonantes débiles en vocales.

La adopción del alfabeto fenicio por los griegos entre el 1100 al 800 a. de N.E. No significa que antes fueran iletrados: como ya hemos visto se valían anteriormente del Lineal B (sistema de escritura silábico que tuvo un lapso de uso que va desde el siglo XVI a. de N.E. al XI a. de N.E., donde el sentido de la escritura es horizontal de izquierda a derecha). Respecto a la razón a la que responde ese cambio de sistema tan repentino, los expertos discrepan. Unos defienden que se basa en razones comerciales y otros, los más románticos, consideran que se debe al deseo de algún culto coetáneo a Homero de conservar la belleza de la Ilíada y la Odisea en un sistema apropiado para escribir poesía épica: sin duda, el sistema fenicio con la introducción de las vocales era ideal.

Así, los etruscos tomarían más tarde el sistema alfabético de los griegos, y los romanos a su vez de los etruscos, expandiéndose así el sistema alfabético ideado por los fenicios a lo largo de Europa y, más tarde, a todo el mundo.

1.5 Sistemas de escritura

Sistemas Incompletos

Los sistemas incompletos se usan para anotaciones, o son mecanismos nemotécnicos que recuerdan hechos significativos o expresan significaciones generales. Estos sistemas, que también reciben el nombre de subescrituras, incluyen la escritura pictórica (o pictográfica), la ideográfica y la que usa objetos marcados y no marcados, como mecanismos nemotécnicos.

Estos sistemas se caracterizan por una gran ambigüedad, dado que no existe correspondencia entre los signos gráficos y la lengua que tratan de representar. La finalidad de un pictograma, un ideograma o un objeto es la de traer a la mente una imagen o una sensación que antes se ha expresado por medio del lenguaje. Éste y no otro era el procedimiento que seguía la escritura pictórica de algunos pueblos

indígenas norteamericanos antes de la colonización, donde cualquiera puede *leer* aunque no conozca la lengua.

De todas maneras, si se trata de interpretar la escritura de un sistema incompleto sin tener conocimiento previo de esa cultura, se corre el peligro de no comprender íntegramente su significado, o de realizar una interpretación errónea. Los pictogramas son los sistemas de escritura más primitivos.

Sistemas Completos

Un sistema completo es aquél que es capaz de expresar en la escritura todo cuanto formule su lengua. Se caracterizan por una correspondencia más o menos estable entre los signos gráficos y los elementos de la lengua que transcriba. Tales elementos pueden ser palabras, sílabas o fonemas (unidad mínima de una lengua que distingue una realización de otra). Así pues, estos sistemas se clasifican en ideográficos (también llamados morfemáticos), silábicos y alfabéticos. Dado que cada signo gráfico representa un elemento de la lengua, hace falta conocer esa lengua para comprender el significado de lo que escribió su autor. Ahora bien, eso no significa que un sistema de escritura esté ligado únicamente a una sola lengua; de hecho, son fácilmente transferibles de una lengua a otra. Lo único que significa es que, a diferencia del pictográfico, ningún sistema completo puede leerse si el lector no comprende la lengua que allí está representada.

Sistemas Ideográficos o Morfemáticos

Se caracterizan porque sus signos, que se llaman ideogramas, representan palabras completas. En algunas ocasiones los signos representan toda una serie de palabras derivadas, y en otras un solo signo representa varias palabras separadas y distintas. En un sistema ideográfico puro estas ambigüedades quedan sin resolver. Sin embargo, para resolverlas existen unos signos determinados que aseguran la lectura correcta. Esos signos se usan como indicadores fonéticos y semánticos, y se suelen llamar complementos fonéticos y determinativos. Los determinativos son los que indican la clase o la categoría

gramatical a la que pertenece la palabra que representa el ideograma. Los determinativos son también ideogramas, pero no se leen, sino que sirven para expresar una clase semántica, como dioses, países, pájaros, peces, verbos de acción, verbos que significan proceso, objetos de madera, de piedra, y así sucesivamente. Los complementos fonéticos tienen un uso parecido, pero muestran de forma más específica cómo se pronuncia toda o parte de la palabra que representa el ideograma. Por ejemplo, dentro de la escritura del español, el ideograma 2 se lee dos. Sin embargo, cuando se escribe el ordinal, hay que añadir el complemento fonético o y el ideograma más el complemento 2.º, se lee segundo, si el complemento fonético se combina con el determinativo que expresa femenino ª, el logograma se transforma en 2.ª y se lee segunda. En este ejemplo se emplean los signos con una finalidad fonética (y no ideográfica). En otras palabras, el signo º funciona no para traer a la mente una idea y la palabra con la que se asocie, sino que trae a la mente un sonido que forma parte de la palabra representada por el ideograma completo. Los indicadores fonéticos surgen a partir de unos ideogramas que tuvieron el mismo significado que el sonido que representan. A este procedimiento se le llama transferencia fonética. Los indicadores fonéticos tampoco se leen, sólo sirven para facilitar la lectura de los ideogramas básicos.

Se ha visto hasta aquí un sistema en el que los elementos de una lengua se representan únicamente por medio de los ideogramas. Ahora bien, esta escritura resulta adecuada para muchos nombres y verbos simples y primitivos, pero no para los adjetivos y los adverbios que suelen ser palabras derivadas, ni tampoco para los pronombres o los nombres propios, y mucho menos puede representar los matices que añaden las terminaciones de caso o de la conjugación verbal. Por lo tanto, según lo que se definió anteriormente, no es un sistema de escritura completo ya que no transcribe todo lo que expresa su lengua. En resumen, si no cumple con este requisito, un sistema ideográfico no será completo, por mucho uso que haga de los indicadores semánticos y fonéticos.

Sistemas Silábicos

Para superar las deficiencias de la escritura ideográfica, se empleaba el principio de transferencia fonética. Cuando se utilizan signos que representan sonidos, sílabas en este caso, se pueden escribir todas las palabras que no era posible hacerlo con la escritura ideográfica. Además, cuando se añaden los signos silábicos a las raíces, es posible representar morfemas, es decir las terminaciones de caso o las de la conjugación verbal. Hay que destacar que deben leerse e interpretarse porque son elementos de la lengua escrita, frente a los indicadores fonéticos.

Un sistema mixto, el ideosilábico, es el primer paso para uno completo. Una vez alcanzada la capacidad para expresarlo todo, el problema se plantea ante la disyuntiva de reducir la ambigüedad o hacer más económico el sistema de escritura (número de signos necesarios para escribir cualquier realización). El problema reside en que se requiere un elevado número de signos, porque el número de palabras que tiene una lengua es también elevado. El segundo paso consiste en reducir el número de signos imprescindibles y eso se puede conseguir si se agrupan en uno sólo todas las palabras de significado parecido, o en emplear el mismo signo para palabras distintas, pero aun así, este sistema necesita unos quinientos o seiscientos signos. Además, la ambigüedad es mucha, a menos que se empleen indicadores, lo que significa sacrificar su ventaja principal, que consiste en tener menos signos por cada realización. Por otro lado, el número de signos que precisa un sistema silábico puro pocas veces supera los doscientos. Frente a la escritura ideográfica, la silábica ofrece una ventaja adicional, no hay que interpretarla puesto que las palabras se escriben sin ambigüedad fonética. La desventaja consiste en que de promedio, el sistema necesita más signos para escribir cada realización. En su forma más sencilla, un sistema silábico está formado por signos de vocal más consonante y signos para las vocales aisladas.

El siguiente paso consiste en reducir la lista de sílabas a signos que representen sólo consonante más vocal, sin diferenciar las vocales. Así se equipara el número

de signos al número de sonidos consonánticos de la lengua, pero se aumenta la ambigüedad, porque el lector debe suplir el sonido vocálico correcto. Dado que se trata de escribir sílabas, los signos necesarios para escribir cada realización son tantos como los de la escritura silábica pura, que además expresa cada una de las vocales. El sistema silábico reducido necesita muchos menos signos y cada uno puede ser más sencillo. Sin embargo, mucha gente considera que esta forma de escribir es un sistema alfabético, o más adecuadamente semialfabético, puesto que no indica cada fonema aislado.

Sistemas Alfabéticos

El último paso hacia una escritura completamente alfabética consiste en escribir por separado los sonidos vocálicos de los consonánticos, lo que precisa de unos cuantos signos más, pero elimina la ambigüedad de tener que suplir las vocales al leer. Por tanto hay más signos para escribir cada realización, aunque el sistema completo necesite menos signos y más sencillos. Puesto que cada uno representa un fonema, la palabra así escrita es su transcripción fonética y no hay que sustituir ningún sonido al leerla.

Estos sistemas trazan la teoría y los procedimientos de escritura, pero hoy por hoy no existen sistemas de escritura que sean una forma pura. Existen elementos de uno y otro tipo incorporados a alguna de las formas que conocemos; un ejemplo de ello es el número de logogramas que son necesarios en los modernos sistemas alfabéticos

CAPÍTULO 2

Esteganografía

" El primer párrafo es el último disfrazado"

Richard Peck

La esteganografía o Stego como se refieren a menudo en la comunidad científica, literalmente significa, escritura cubierta; la cuál se deriva de la lengua griega. La esteganografía es definida por Markus Kahn[10] como: " La esteganografía es el arte y la ciencia de la comunicación, de una manera en la cual se oculta la existencia de la comunicación. En contraste con la criptografía, donde se le permite al enemigo detectar, interceptar y modificar los mensajes sin violar ciertas premisas de la seguridad garantizadas por un sistema criptográfico, el objetivo de la esteganografía es ocultar mensajes dentro de otros mensajes inofensivos de una manera que no permite a cualquier enemigo detectar incluso que hay un segundo mensaje".

En un mundo digital, la esteganografía y la criptografía[11] están destinadas para proteger la información de entidades indeseadas. La esteganografía y la criptografía son medios excelentes por los cuales se puede lograr esto, pero ninguna tecnología es perfecta y ambas pueden ser vulnerables. Es por esta razón que la mayoría de los expertos sugieren usar ambas para agregar capas múltiples de seguridad.

La esteganografía se puede utilizar en una gran cantidad de formatos de datos en el mundo digital de hoy. Los formatos de datos más populares usados son: .bmp, .doc, .gif, .jpeg, .mp3, .txt y .wav. Principalmente debido a su renombre en Internet y la facilidad de empleo de las herramientas esteganográficas que utilizan estos formatos de datos. Estos formatos son también populares debido a la facilidad

[10] Autor del libro "Steganography"

[11] Son las técnicas utilizadas para cifrar y descifrar información utilizando técnicas matemáticas que hagan posible el intercambio de mensajes de manera que solo puedan ser leídos por las personas a quienes van dirigidos.

relativa por la cual los datos redundantes o ruidosos pueden ser removidos de éstos, y sustituirlos por un mensaje oculto.

Las tecnologías esteganográficas son parte importante del futuro de la seguridad de Internet y de la privacidad en los sistemas abiertos. La investigación de la esteganografía se conduce sobre todo por la carencia de fuerza en los sistemas criptográficos y el deseo de tener secreto completo en un ambiente de sistemas abiertos. Muchos gobiernos han creado leyes que limitan la fuerza de sistemas criptográficos o los prohíbe totalmente. Esto desafortunadamente deja a la mayoría de la comunidad de Internet relativamente débil y, muchas veces, los algoritmos de encriptación son frágiles o no hay ninguno. Aquí es adonde viene la esteganografía. La esteganografía se puede utilizar para ocultar datos importantes dentro de otro archivo, de modo que solamente las entidades previstas para conseguir el mensaje sepan que existe un mensaje secreto. Para agregar capas múltiples de seguridad y ayudar a subsidiar "la criptografía contra la ley", problemas mencionados previamente, es una buena práctica utilizar la criptografía junto con la esteganografía. Pero ni la criptografía ni la esteganografía se consideran "soluciones" para la privacidad de sistemas abiertos, pero al usar ambas tecnologías, pueden proporcionar una cantidad muy aceptable de privacidad para cualquier persona que se conecta a Internet y para la comunicación sobre estos sistemas abiertos.

2.1 Evidencias en la historia de la esteganografía.

Las apariciones más tempranas de la esteganografía datan del historiador griego Herodotus, en su crónicas conocidas como "Historias", alrededor del 440 a. de N.E. Herodotus registró dos historias de las técnicas esteganográficas que se usaban durante este tiempo en Grecia. La primera indica que el Rey Darius de Susa afeitó a uno de sus presos y escribió un mensaje secreto en su cuero cabelludo. Cuando el pelo del preso creció, lo enviaron con el yerno Aristogoras de los reyes de Miletus, desapercibido. La segunda historia también vino de Herodotus, que dice que un soldado llamado Demeratus necesitó enviar un

mensaje a Esparta, en el cual Xerxes se proponía invadir Grecia. En ese entonces, el medio de escritura era texto escrito en tabletas cubiertas de cera. Demeratus quitó la cera de la tableta, escribió el mensaje secreto en la madera, cubrió de nuevo la tableta con la cera para hacer que pareciera como una tableta en blanco y finalmente envió el documento sin ser detectado.

Los romanos utilizaron las tintas invisibles, que fueron basadas en sustancias naturales tales como zumos y leche de fruta. Esto se lograba calentando el texto, y así se revelaba su contenido. Las tintas invisibles han llegado a ser mucho más avanzadas y todavía actualmente están en uso. Una variación curiosa del concepto de tinta invisible apareció en el siglo XVI, cuando el científico Italiano Giovanni Porta describió como esconder un mensaje dentro de un huevo cocido, que constaba en hacer una mezcla de vinagre y alumbre y pintar el huevo con el mensaje, esto hacía que el huevo absorbiera la tinta y que el mensaje sólo pudiera ser leído si se pelará el huevo.

Durante los siglos XV y XVI, muchos escritores incluyendo Juan Trithemius (autor de Steganographia) y Gaspari Schotti (autor de Steganographica) escribieron técnicas esteganográficas tales como de codificación para texto, tintas invisibles, y los mensajes ocultos en música.

Entre 1883 y 1907, el desarrollo se puede atribuir a las publicaciones de Auguste Kerckhoff (autor de Cryptographic Militaire) y Charles Briquet (autor de Les Filigranes). Estos libros tratan, sobre todo, de criptografía, pero ambos se les puede atribuir al fundamento de algunos sistemas esteganográficos y más perceptiblemente a las técnicas de marca de agua.

Durante los tiempos de la primera y segunda guerra mundial, significativos avances en la esteganografía dieron lugar. Los conceptos tales como cifras nulas (tomando la 3a letra de cada palabra en un mensaje inofensivo para crear un mensaje oculto), la sustitución de imágenes y micropuntos o microdot (que toma datos como texto o imágenes y los reducen de tamaño sobre un trozo de papel) fueron introducidos como grandes técnicas esteganográficas.

En el mundo digital de hoy, de 1992 al presente, la esteganografía se está utilizando en todo el mundo sobre sistemas informáticos. Se han creado muchas herramientas y tecnologías que se aprovechan de viejas técnicas esteganográficas tales como cifras nulas, cifrado en imágenes, audio, vídeo y microdot. Con la investigación que se está consiguiendo ahora, veremos muchos usos para la esteganografía en un futuro próximo.

2.2 Una mirada detallada a la Esteganografía.

Para comenzar, veremos en que consiste la comunicación secreta perfecta, hablando teóricamente. Para ilustrar este concepto, utilizaremos tres caracteres ficticios nombrados Amy, Bret y cristal. Amy quiere enviar un mensaje secreto (m) a Bret, usando (r) un mensaje inofensivo para crear la cubierta (c) que se va a enviar a Bret, sin el aumento de suspicacia. Amy entonces cambia el mensaje inofensivo para crear (c) un stego-objeto (s), incrustando el mensaje secreto (m) en el mensaje inofensivo (c), usando una stego-llave (k). Amy debe entonces poder enviar el stego-objeto (s) a Bret sin ser detectado por Crystal. Bret entonces podrá leer el mensaje secreto (m) porque él sabe la stego-llave (k) usada para encajarla en el mensaje cubierto (c).

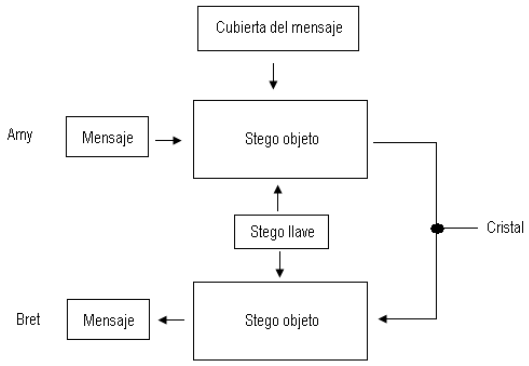

Figura 2.1 Esquema de una comunicación secreta perfecta.

Mientras que Fabián A.P. Petitcolas[12] precisa: “En un sistema perfecto, una cubierta normal no debe ser distinguible de un stego-objeto, ni por un ser humano ni por una computadora que busca patrones estadísticos”. En la práctica, sin embargo, éste no es siempre el caso. Para incrustar datos secretos en la cubierta de un mensaje, la cubierta debe contener una suficiente cantidad de datos redundantes o de ruido. Esto es porque las aplicaciones del proceso de incrustación de la esteganografía, substituyen realmente estos datos redundantes por el mensaje secreto. Esto limita los tipos de datos que se pueden utilizar con la esteganografía.

En la práctica, hay básicamente tres tipos de protocolos esteganográficos usados. Son esteganografía pura, esteganografía con llave secreta y esteganografía con llave pública.

La esteganografía pura se define como el sistema esteganográfico que no requiera el intercambio de una cifra, tal como una stego-llave. Este método de esteganografía es de los menos seguros para comunicarse secretamente, porque el remitente y el receptor pueden confiar solamente en la presunción de que no hay otras entidades esperando este mensaje secreto. Usando sistemas abiertos tales como Internet, sabemos que éste no es el caso en lo absoluto.

La esteganografía con llave secreta se define como el sistema esteganográfico que requiere el intercambio de una llave secreta (stego-llave) antes de la comunicación. Este tipo de esteganografía toma la cubierta del mensaje e incrusta el mensaje secreto dentro de él, usando una llave secreta (stego-llave). Solamente las entidades que saben la llave secreta pueden invertir el proceso y leer el mensaje secreto. A diferencia de la esteganografía pura, donde hay un canal de comunicación invisible, la esteganografía con llave secreta intercambia la llave, y hace que sea más susceptible la intercepción. La ventaja de la esteganografía con llave secreta es que, incluso si se intercepta, sólo los partidos que saben la llave secreta pueden extraer el mensaje secreto.

[12]Investigador de la universidad de Portland, Oregon.

La esteganografía con llave pública se define como un sistema esteganográfico que utiliza una llave pública y una llave privada para asegurar la comunicación entre las entidades que quieran comunicarse secretamente. El remitente utilizará la llave pública durante el proceso de la codificación y solamente la llave privada, que tiene una relación matemática directa con la llave pública, puede descifrar el mensaje secreto. La esteganografía con llave pública proporciona una manera más robusta de implementar un sistema esteganográfico, porque puede utilizar una tecnología mucho más robusta e investigada en criptografía sobre la llave pública. También tiene niveles de seguridad múltiples en que, las entidades indeseadas deben primero sospechar el uso de la esteganografía en un mensaje, y entonces, tendrían que encontrar una manera de romper el algoritmo usado por el sistema de llave pública antes de que pudieran interceptar el mensaje secreto.

2.4 Codificando mensajes secretos en texto.

Codificando mensajes secretos en texto puede ser una tarea muy desafiante. Esto es porque los archivos de texto tienen una pequeña cantidad de datos redundantes a substituir por un mensaje secreto. Otra desventaja de usar esteganografía en texto, es la facilidad de alterar el texto por las entidades indeseadas, apenas cambiando el texto o cambiando el formato del texto (de .TXT a .PDF, etc.). Hay métodos numerosos por los cuales se puede lograr la esteganografía en texto, menciono los siguientes:

- **Codificación Cambio de Línea (line-shift):** Implica realmente el cambiar de puesto cada línea de texto verticalmente, para arriba o abajo, cerca de 3 centímetros. Dependiendo de si la línea, fue para arriba o hacia abajo, se compararía a un valor que se podría codificar en un mensaje secreto.

- **Codificación Cambio de Palabra (Word-shift):** trabaja más o menos de la misma manera que la anterior, solamente utilizamos los espacios entre las palabras horizontales para comparar un valor para el mensaje oculto. Este

método de codificación es menos visible que el anterior, pero requiere que el formato de texto soporte el espacio variable.

- **Codificación de característica específica:** implica codificar mensajes secretos en el texto, cambiando ciertas características del texto, tales como longitud, vertical/horizontal, de letras tales como b, d, T, etc. Éste método de codificación es el más difícil de interceptar, pues cada tipo de formato de texto tiene una gran cantidad de características que se pueden utilizar para codificar el mensaje secreto.

Estos tres métodos de codificación basados en texto, requieren el archivo original o el conocimiento del formato original para poder descifrar el mensaje secreto.

2.5 Codificando mensajes secretos en imágenes.

Estos tipos de codificación, es en gran medida los más ampliamente utilizados de todos los métodos en el mundo digital de hoy. Esto es porque puede aprovecharse del campo limitado del sistema visual humano. Casi cualquier texto plano, texto cifrado, imagen y cualquier otro medio, puede ser codificado en un flujo de bits que puede ser ocultado en una imagen digital. Con el continuo crecimiento de los gráficos en computadoras y la investigación de la esteganografía basada en imágenes, este campo continuará creciendo a un paso muy veloz.

Antes de entrar en las técnicas de codificación para las imágenes digitales, hago una breve explicación de la arquitectura de la imagen digital y las técnicas de compresión de imagen digital.

Como Duncan Sellars[13] explica: “Para una computadora, una imagen es un arreglo de números que representa intensidades de luz en varios puntos, o pixeles. Estos pixeles componen la trama de datos de las imágenes.”

[13] Autor del libro “*An Introduction to Steganography*”.

Al ocupar imágenes digitales para el uso con la esteganografía, los archivos de imágenes de 8 y 24 bits por pixel, son típicos. Ambos tienen ventajas y desventajas. Las imágenes de 8 bits son un gran formato a utilizar debido a su tamaño relativamente pequeño. La desventaja es que solamente 256 colores pueden ser utilizados, y esto puede ser un problema potencial durante la codificación. Generalmente se utiliza una gama de colores en la escala de grises al ocupar imágenes de 8 bits, por ejemplo .GIF, porque su cambio gradual en color será más difícil de detectar después de que la imagen se haya codificado con el mensaje secreto. Las imágenes de 24 bits ofrecen mucha más flexibilidad cuando éstas son utilizadas para la esteganografía. Una gran cantidad de los colores (sobre los 16 millones) pueden ser utilizados más allá del sistema visual humano (HVS por sus siglas en inglés, “human visual system”), que hace muy difícil de detectar un mensaje secreto. La otra ventaja es que una cantidad mucho más grande de datos ocultos se puede codificar en imágenes digitales de 24 bits, en comparación con una imagen digital de 8 bits. La desventaja principal de las imágenes digitales de 24 bits, son su gran tamaño (generalmente en el orden de los Megabytes, MB), las hacen más sospechosas que las imágenes digitales de 8 bits, mucho más pequeñas (generalmente en el orden de los Kilobytes, KB), cuando éstas son enviadas sobre un sistema abierto, tal como la Internet.

La compresión de imágenes Digitales es una buena solución para las imágenes digitales grandes, tales como las imágenes de 24 bits. Hay dos tipos de compresión usados en las imágenes digitales, lossy (con pérdidas) y lossless (sin pérdidas). La compresión lossy por, ejemplo .JPEG, reduce considerablemente el tamaño de una imagen digital, quitando el exceso de datos de la imagen y calculando una aproximación cercana de la imagen original.

La compresión lossy se utiliza generalmente con imágenes digitales de 24 bits para reducir su tamaño, pero conlleva una desventaja importante. Las técnicas de compresión lossy aumentan la posibilidad que el mensaje secreto, sin comprimir, pierda partes de su contenido, debido al hecho de que la compresión lossy quita lo que considera como exceso de datos de la imagen. Las técnicas de compresión

lossless, mantienen la imagen digital original intacta, sin pérdida. Es por esta razón que esta técnica de compresión es escogida para usos esteganográficos. Los ejemplos de las técnicas de compresión lossless son .GIF y .BMP. La única desventaja de la compresión de imágenes losless es que no hacen un trabajo muy bueno en la compresión del tamaño de los datos de la imagen.

Ahora veremos un par de técnicas de codificación para imágenes digitales, que se han popularizado hoy en día. Estas técnicas son: LSB (por sus siglas en inglés, Least Significant Bit, que significa bit menos significativo), máscara y filtros.

2.6 Codificación LSB

La codificación LSB, es en gran medida la técnica más popular de codificación usada para las imágenes digitales. Usando el bit menos significativo de cada octeto (8 bits) en una imagen para un mensaje secreto, se puede almacenar 3 bits de datos en cada pixel en imágenes de 24 bits y 1 bit en cada pixel para las imágenes de 8 bits. Como se puede observar, se puede almacenar mucha más información en archivos de imagen de 24 bits. Dependiendo de la gama de colores usados para la cubierta de la imagen (por ejemplo, todo gris), es posible tomar 2 LSB de un octeto sin que el sistema visual humano pueda notar la diferencia. El único problema con esta técnica es que es muy vulnerable a los ataques tales como: cambios y formato de la imagen (es decir, cambiando de .GIF a .JPEG).

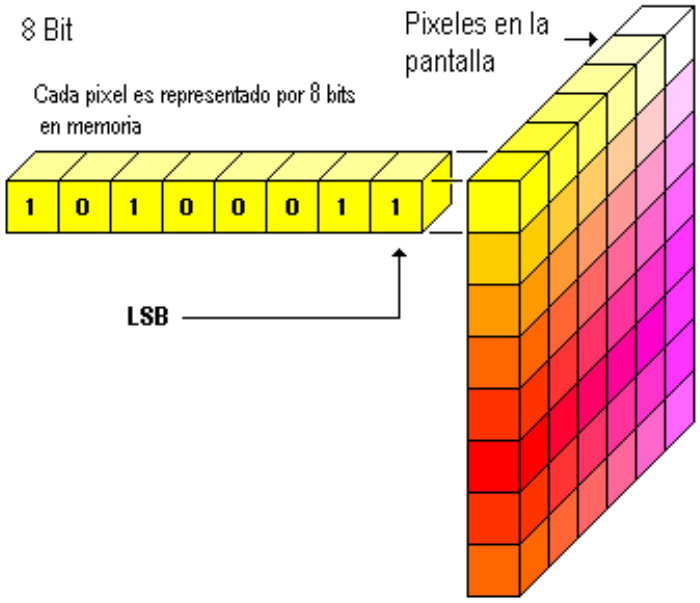

Figura 2.2. LSB en imágenes de 8 bits.

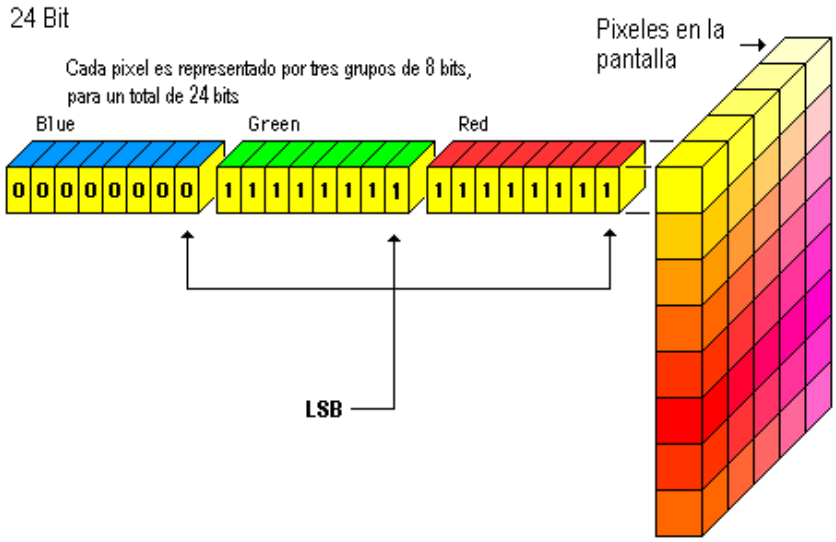

Figura 2.3. LSB en imágenes de 24 bits.

Las técnicas de enmascaramiento y de filtración para la codificación de imágenes digitales tal como marcas de agua (es decir - integrando un logo de las compañías en el contenido de una imagen) es más popular entre técnicas de compresión

lossy, por ejemplo .JPEG. Esta técnica enmascara los datos secretos sobre los datos originales en comparación a esconder información dentro de los datos. Algunos expertos sostienen que ésta es definitivamente una forma de ocultar información, pero no es técnicamente esteganografía. La gran ventaja de técnicas de enmascaramiento y de filtración, es que son inmunes a la manipulación de la imagen y hacen el uso posible de aplicaciones robustas.

2.6 Codificando mensajes secretos en audio.

Los mensajes secretos en audio es la técnica más desafiadora para utilizar, en cuanto a esteganografía se refiere. Esto es porque el sistema auditivo humano tiene tal rango dinámico sobre el cual se puede escuchar. Para poner esto en perspectiva, el sistema auditivo humano es tan sensible en rango y en frecuencias que hacen extremadamente difícil agregar o quitar datos de la estructura de datos original. La única debilidad en el sistema auditivo humano viene de intentar distinguir sonidos (los sonidos ruidosos ahogan a los sonidos tenues) y éste es el qué se debe explotar para codificar mensajes secretos en audio sin detección.

Hay dos conceptos a considerar antes de elegir una técnica de codificación para audio. Son el formato digital y el medio de transmisión del audio. Hay tres formatos de audio digital principales típicamente funcionando. Son muestreo y cuantificación, taza de muestreo temporal y muestreo perceptivo.

- **Muestreo y cuantificación:** es una arquitectura de 16 bits, que es el numero de bits de datos usados para representar la señal analógica, usada en formatos de audios populares, por ejemplo .WAV y. AIFF.

- **La taza de muestreo temporal:** utiliza frecuencias seleccionables (en los KHz) para muestrear el audio. Generalmente, cuanto más alto es la taza de muestreo, es más alto el espacio para guardar los datos.

- **El muestreo perceptivo:** cambia las estadísticas del audio, codificando solamente las partes que el oyente percibe, así se mantiene el sonido claro,

pero cambiando la señal. Este formato es utilizado por el más popular audio digital en Internet que es el MP3 (ISO MPEG).

El medio de transmisión (trayectoria que el audio lleva del remitente al receptor) debe también ser considerado al codificar mensajes secretos en audio. W. Bender[14] introduce cuatro medios posibles de transmisión:

1. Digital end to end: de máquina a máquina sin modificación.

2. Increased/decreased resampling: la taza de muestreo se modifica pero sigue siendo digital.

3. Analog and resampled: la señal se cambia a análoga y se vuelve a muestrear a una diferente taza.

4. Over the air: la señal se transmite en radiofrecuencias y se vuelve a muestrear desde un micrófono.

Ahora veremos tres de los métodos de codificación más populares para ocultar datos dentro del audio. Son la codificación LSB (por sus siglas en inglés, Least Significant Bit, que significa bit menos significativo), codificación de fase y espectro ensanchado.

La codificación del bit menos significativo (LSB) embebe datos secretos en el bit menos significativo del archivo de audio. La capacidad del canal es de 1KB por segundo por kilohertz (44 Kbps para una secuencia muestreada a 44 kilohertz). En este método es fácil incorporar datos pero es muy susceptible a la pérdida de los datos debido al ruido del canal o a volver a muestrear.

La codificación de fase substituye la fase de un segmento de audio con una referencia de fase que representa los datos ocultos.

[14] Autor del artículo “Techniques for Data Hiding”, IBM Systems Journal.

La extensión del espectro codifica el audio sobre casi todo el espectro de la frecuencia. Entonces transmite el audio sobre diversas frecuencias que variarán dependiendo de qué método de extensión de espectro se utiliza. La secuencia directa de extensión de espectro (DSSS) es un método que separa la señal multiplicando la señal de la fuente por una cierta pseudo secuencia al azar conocida como “Chip”. La taza de muestreo entonces se utiliza como la taza de la pseudo secuencia (Chip) para la comunicación de la señal audio. Las técnicas de codificación del espectro son los medios más seguros por los cuales se puede enviar mensajes ocultos en audio, pero puede introducir ruido al azar al audio creando pérdida de los datos.

Hay muchas aplicaciones para la esteganografía, algunas buenas y otras malas, que nos lleva a la sección final de la esteganografía, en la que vamos a ver: el estegoanálisis. El estegoanálisis es el arte y la ciencia de detener o detectar el uso de todas las técnicas esteganográficas mencionadas anteriormente.

En el estegoanálisis, el objetivo es poder comparar el objeto de cubierta (cubierta del mensaje), el stego-objeto (la cubierta del mensaje con los datos ocultos incrustados en ella) y cualquier parte posible de la stego llave (método de cifrado) en un esfuerzo por interceptar, analizar y / o destruir la comunicación secreta.

Como en su libro, Fabien A. P. Petitcolas[15], hay seis protocolos generales utilizados para atacar el uso de la esteganografía.

1. Stego only attack - Solo el objeto stego está disponible para su análisis.

2. Known cover attack - el objeto de cubierta original y el objeto stego están disponibles para análisis.

3. Known message attack - el mensaje oculto está disponible para compararse con el objeto stego.

[15] Autor del libro “Information Hiding: Techniques for Steganography and Digital Watermarking.”

4. Chosen stego attack - la herramienta stego (algoritmo) y el objeto stego están disponibles para su análisis.

5. Chosen message attack - tiene un mensaje seleccionado y genera un objeto stego para futuros análisis.

6. Known stego attack - la herramienta stego (algoritmo), el mensaje de la cubierta y los objetos stego están disponibles para su análisis.

Siendo que estegoanálisis es un tema muy amplio y que merece una investigación propia, voy a cerrar esta investigación del estegoanálisis por mostrar al lector un breve ejemplo de cómo alguien podría detectar el uso de herramientas esteganográficas al cambiar el bit menos significativo (LSB) de una imagen con el fin de integrar los datos secretos en su interior.

En general, en las imágenes de mapa de bits (. BMP) se sabe y tienen características predecibles. Una de estas características es la probabilidad de colores duplicados. Las imágenes de mapa de bits obtienen su color de una tabla de colores central, que por su naturaleza tienen pocos o no hay colores duplicados. Cuando los datos ocultos se incrustan en el bit menos significativo (LSB) de una imagen de mapa de bits, incrementa el número de colores duplicados de forma espectacular. En general, cualquier imagen de mapa de bits con más de cincuenta colores duplicados debe despertar la sospecha de que hay datos incorporados en ella.

CAPÍTULO 3

Marcas de Agua

"¿Sabes dibujar sonidos?

No comprendí bien qué quería decir y le pregunté a Herger, y después de hablar un poco más, me di cuenta finalmente que se refería a la escritura."

Los devoradores de cadáveres, Michael Crichton.

El rápido desarrollo de Internet y la revolución de la información digital provoca cambios significativos en la sociedad global, que van desde la influencia en la economía mundial a la manera de como la gente hoy en día se comunica. Las redes de comunicación de banda ancha y los datos multimedia en un formato digital, ya sea imágenes, audio ó vídeo, abrió muchos retos y oportunidades para la innovación. El software fácil de utilizar y los precios de los dispositivos digitales (por ejemplo, cámaras fotográficas digitales, videocámaras, CD portátiles, reproductores de MP3, reproductores de DVD, grabadoras de DVD y CD, laptops, PDA), han hecho posible que los consumidores de todo el mundo puedan crear, editar e intercambiar datos multimedia. Las conexiones de banda ancha y la transmisión de datos, facilitará a la gente distribuir grandes archivos multimedia y hacer copias digitales idénticas de ellos.

Los archivos multimedia digitales no sufren ninguna pérdida de calidad debido a los procesos de copia múltiple, tales como de audio analógico y cintas de VHS. Además, el soporte de grabación y las redes de distribución de multimedia analógica son más caros. Las ventajas de los medios de comunicación digital al contrario de las analógicas se transforman en desventajas con respecto a la gestión de derechos intelectuales, porque la copia ilimitada sin pérdida de fidelidad puede causar una pérdida económica considerable para los titulares de derechos de autor. La facilidad de modificación de contenidos y una reproducción perfecta en el dominio digital, han promovido la protección de la propiedad intelectual y la

prevención de la manipulación no autorizada de datos multimedia y se han convertido en un importante tema de investigación tecnológica.

El uso razonable de los datos multimedia, combinado con una rápida entrega de datos multimedia para los usuarios con distintos dispositivos se está convirtiendo en un tema difícil e importante. Los métodos tradicionales para la protección de los derechos de autor de datos multimedia ya no son suficientes. Los sistemas de protección contra copia de datos multimedia analógica ya han sido fácilmente eludidos. En los sistemas digitales es aún más fácil debido a la disponibilidad de plataformas de procesamiento multimedia en general, por ejemplo, una computadora personal. Un mecanismo simple de protección se basaba en insertar información en la cabecera del archivo (header) y este mecanismo es inútil porque la información de encabezado se puede quitar fácilmente por un simple cambio de formato de datos, que no afecta la fidelidad.

La Marca de agua digital ha sido propuesta como un nuevo método alternativo para hacer valer los derechos de propiedad intelectual y proteger sus medios digitales de la manipulación. Se trata de un proceso de integración, en una señal huésped, de la firma digital perceptualmente transparente, que lleva un mensaje acerca del autor. La firma digital se llama la marca de agua digital. La marca de agua digital contiene datos que pueden ser utilizados en diversas aplicaciones, incluyendo la gestión de derechos digitales, el control de difusión y de manipulación de pruebas. A pesar de la percepción transparente, la existencia de la marca de agua es revelada cuando se transmite a través de un detector de marca de agua adecuado.

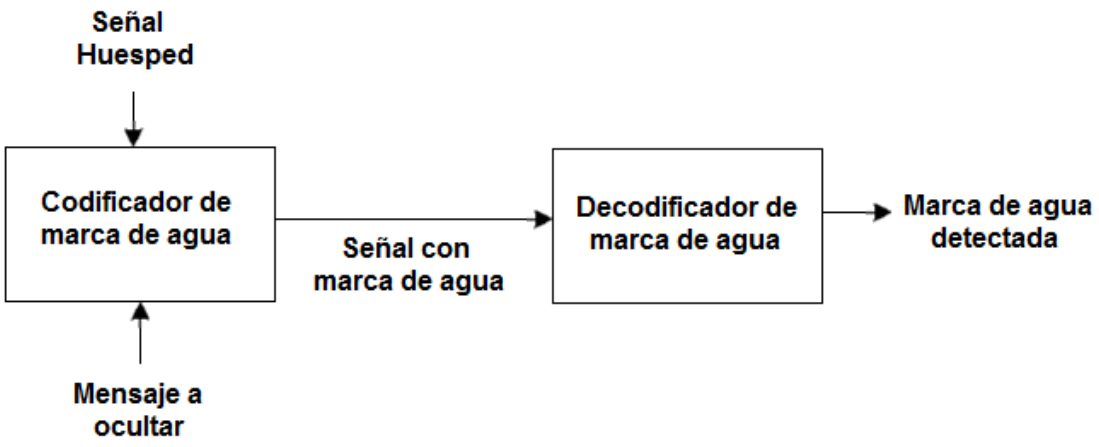

Figura 3.1. Un diagrama de bloques del codificador.

La Figura 3.1 da una visión general del sistema de marca de agua. Una marca de agua, que generalmente consiste en una secuencia de datos binarios, se inserta en la señal huésped en el bloque watermark embedder. Así, el bloque watermark embedder dispone de dos entradas, una es el mensaje de marca de agua (generalmente acompañado de una clave secreta) y la otra es la señal huésped (por ejemplo, una imagen, un clip de vídeo, una secuencia de audio, etc.). La salida del bloque watermark embedder es la señal de marca de agua. La señal de marca de agua es transmitida y posteriormente presentada al detector de marca de agua. El detector determina si la marca de agua está presente en la señal multimedia, y si es así, ¿Qué mensaje está codificado en el mismo? El área de investigación de marcas de agua está estrechamente relacionada con los ámbitos de ocultación de información y la esteganografía. Los tres campos tienen un considerable solapamiento y muchas soluciones técnicas comunes. Sin embargo, existen algunas diferencias filosóficas fundamentales que influyen en los requisitos y por lo tanto en el diseño de una solución técnica en particular. El ocultamiento de información (o de ocultación de datos) es un área más general, que abarca una gama más amplia de los problemas que la marca de agua. El término ocultar se refiere al proceso de hacer la información imperceptible o de mantener la existencia de información en secreto.

Por lo tanto, podemos definir los sistemas de marcas de agua como los sistemas en los que el mensaje oculto está relacionado con la señal huésped y no a los sistemas en que el mensaje no está relacionado con la señal huésped. Por otro lado, los sistemas para insertar mensajes en las señales huéspedes se puede dividir en sistemas esteganográficos, en donde la existencia del mensaje se mantiene en secreto, y los sistemas no esteganográficos, en donde la presencia del mensaje embebido no tiene que ser secreto. Esta diferencia la podemos observar en la siguiente tabla, que muestra 4 categorías de sistemas de información oculta.

	Señal huésped dependiente del mensaje	Señal huésped independiente del mensaje
Mensaje Oculto	Comunicación secreta	Marca de agua esteganográfica
Mensaje Conocido	Marca de agua no esteganográfica	Comunicación embebida pública

Tabla 3.1. Cuatro categorías de sistemas de ocultación de información.

3.1 Áreas de aplicación.

La marca de agua digital es considerada como una comunicación imperceptible, robusta y segura de los datos relativos a la señal huésped, que incluye la inserción y en la extracción de la señal huésped. El objetivo fundamental es que la información de la marca de agua embebida soporte modificaciones no intencionales e intentos de eliminación intencional. El reto principal es el diseño de una marca de agua a integrar de manera que pueda ser extraída con seguridad en un detector de marca de agua. La importancia relativa de las propiedades mencionadas depende significativamente de la aplicación para la que está diseñado el algoritmo. Algunas áreas de aplicación son las siguientes:

Protección de propiedad.

En las aplicaciones de protección de propiedad, una marca de agua que contiene información de propiedad se incorpora a la señal de multimedia huésped. La marca de agua, conocida solamente por el titular del derecho de autor, se espera que sea muy robusta y segura (es decir, para sobrevivir modificaciones comunes de procesamiento de señal y ataques intencionales), que permite al propietario demostrar la presencia de esta marca de agua en caso de litigio para demostrar su propiedad.

Prueba de propiedad.

Es aún más exigente para el uso de marcas de agua no sólo en la identificación de la propiedad de derechos de autor, sino como una prueba real de la propiedad. El problema surge cuando un atacante utiliza un software de edición para reemplazar la nota de copyright original con la suya, y a continuación afirma tener los mismos derechos de autor. En el caso de los primeros sistemas de marca de agua, el problema era que el detector de marca de agua está fácilmente disponible para los atacantes. Como es de suponerse, si alguien que puede detectar una marca de agua, probablemente puede quitarla también. Por lo tanto, debido a que un atacante puede obtener fácilmente un detector, puede quitar la marca de agua del propietario y sustituirla por la suya. Para alcanzar el nivel de la seguridad necesaria para la prueba de la propiedad, es indispensable restringir la disponibilidad de los detectores. Cuando un atacante no tiene el detector, la eliminación de una marca de agua puede ser extremadamente difícil. Sin embargo, incluso si una marca de agua no puede ser eliminada, un atacante podría tratar de sobreponer su marca de agua. Como es de suponerse, un atacante, utilizando su propio sistema de marca de agua, podría ser capaz de hacer que parezca como si sus datos de marca de agua estuvieran presentes en la señal original huésped de los propietarios. Este problema se puede resolver mediante una ligera modificación del problema. En lugar de una prueba directa de la propiedad mediante la inserción, por ejemplo "Dave posee esta imagen" como

firma de marca de agua en la imagen huésped, el algoritmo en cambio, trataría de probar que la imagen del atacante se deriva de la imagen de marca de agua original. Este algoritmo proporciona evidencia indirecta de que lo más probable es que el propietario real de la imagen en disputa, es el que tiene la versión de la que derivan las demás marcas de agua.

Autenticación y detección de manipulación.

En las aplicaciones de autenticación de contenidos, un conjunto de datos secundarios se incrusta en la señal multimedia huésped y se utiliza más tarde para determinar si se ha manipulado la señal. La eliminación de la marca de agua no es una preocupación ya que no hay tal motivo, desde el punto de vista del atacante. Sin embargo, forjar una marca de agua de autenticación válida en una señal huésped no autorizada o alterada debe ser prevenida. En las aplicaciones prácticas, es también deseable localizar (en tiempo o dimensión espacial) y discriminar las modificaciones no intencionales (por ejemplo, las distorsiones que incurra debido a la compresión MPEG, o cualquier tipo de compresión) de la alteración del contenido en sí. La detección se debe realizar sin la señal huésped original, ya sea porque el original no está disponible o su integridad aún no se ha establecido. Este tipo de detección de marca de agua se suele llamar “detección a ciegas”.

Huellas dactilares (fingerprinting).

Los datos adicionales incorporados por la marca de agua en las aplicaciones de huellas digitales se utilizan para localizar el autor o los destinatarios de una copia particular de archivo multimedia. Por ejemplo, la marca de agua lleva un numero de serie o un número de identidad (id) que se insertan en diferentes copias de CD de música o DVD antes de distribuirlos a un gran número de destinatarios. Los algoritmos implementados en las aplicaciones de huellas dactilares deben mostrar gran robustez frente a ataques intencionales y procesamiento de señales, tales como la compresión con pérdida (Compresión lossy) o filtración (filtering). Las huellas dactilares también requieren buenas propiedades de anti colusión de los

algoritmos, es decir, no es posible integrar más de un número de identificación (id) para el archivo multimedia, de lo contrario el detector no será capaz de distinguirla.

Control de emisión.

Una gran variedad de aplicaciones de marcas de agua en audio están en el ámbito de la radiodifusión. La marca de agua es un método alternativo de identificación para un control de transmisión activa. Tiene la ventaja de ser embebido dentro de la propia señal huésped multimedia en lugar de la explotación de un segmento particular de la señal de emisión. Por lo tanto, es compatible con los equipos de radiodifusión, incluidos los canales de comunicación digitales y analógicos. El principal inconveniente es que el proceso de incrustación es más complejo que una simple puesta de datos en los encabezados del archivo. Existe también una preocupación, especialmente por parte de los propietarios, y esta es que la marca de agua puede introducir distorsiones que degradan la calidad visual o de audio del archivo multimedia.

Control de copia y control de acceso.

En la aplicación de control de copia, la marca de agua embebida representa un ejemplar de control de copia o una política de control de acceso. Un detector de marca de agua es generalmente integrado en un sistema de grabación o de reproducción, como en la propuesta del algoritmo de control de copia de DVD[16], o durante el desarrollo de "Iniciativa de Música Digital Segura" (SDMI, Secure Digital Music Initiative)[17]. Después de que una marca de agua se ha detectado y el contenido decodificado, el control de copia o la política de control de acceso es impuesta al hardware o software, tal como habilitar o deshabilitar el módulo de grabación. Estas aplicaciones requieren algoritmos de marcas de agua resistente a ataques intencionales y a modificaciones de procesamiento de señales, capaz de realizar una "*detección a ciegas*".

[16] Bloom J, Cox I, Kalker T, Linnartz J, Miller M & Traw C (1999) Copy protection for dvd video. IEEE volumen 87 número:7.

[17] Craver S & Stern J (2001) Lessons learned from sdmi. IEEE International Workshop on Multimedia Signal Processing, Cannes, France.

Portador de información.

La marca de agua embebida en esta aplicación se espera que tenga una alta capacidad y debe ser detectado y descifrado usando un algoritmo de "*detección a ciegas*". Si bien la robustez frente al ataque intencional no es necesaria, un cierto grado de robustez frente al tratamiento común, como la compresión MPEG, se puede desear. Una marca de agua pública incorporada en el huésped multimedia puede ser utilizada como un enlace a una base de datos externa que contienen cierta información adicional acerca de los archivos multimedia en sí, tales como información sobre el copyright y las condiciones de concesión de licencias. Una aplicación interesante es la transmisión de metadatos junto con multimedia.

3.2 Áreas de investigación.

Los algoritmos de marca de agua se caracterizan por una serie de definiciones de propiedades. Seis de ellas, que son más importantes para los algoritmos de audio de marcas de agua, representan subáreas de investigación. La importancia relativa de una subárea en particular depende de la aplicación, y en muchos casos la interpretación de una propiedad de marca de agua varía con la aplicación.

Percepción de transparencia.

En la mayoría de las aplicaciones, el algoritmo de la marca de agua tiene que insertar los datos sin afectar la calidad percibida de la señal de audio huésped. La fidelidad del algoritmo de marcas de agua se define generalmente como una semejanza perceptual entre la secuencia de audio original y la marca de agua. Sin embargo, la calidad del audio con marca de agua suele degradarse, ya sea intencionalmente por un atacante o accidentalmente en el proceso de transmisión, antes de que una persona lo perciba. En ese caso, es más adecuado definir la fidelidad de un algoritmo de marca de agua como una similitud entre la percepción del audio de la marca de agua y el audio original en el punto en que son presentados a un consumidor.

Tasa de bits de la marca de agua.

La tasa de bits de la marca de agua embebida es el número de bits incrustados dentro de una unidad de tiempo y generalmente se da en bits por segundo (bps). Algunas aplicaciones de audio de marcas de agua, como el control de copia, requieren la inserción de un número de serie o identificación del autor, con la tasa de bits media de hasta 0,5 bps. Para una marca de agua de control de emisión, la velocidad es mayor, causada por la necesidad de la incorporación de una firma de identificación de un comercial en el primer segundo del comienzo del clip de difusión, con una tasa de bits media de hasta 15 bps.

Robustez.

La robustez del algoritmo se define como la capacidad del detector de marca de agua para extraerla después de manipulaciones comunes de procesamiento de señales. Las aplicaciones normalmente requieren la robustez en presencia de un conjunto predefinido de modificaciones de procesamiento de señales, de modo que una marca de agua puede ser extraída de forma fiable en la parte de detección. Por ejemplo, en el control de emisión de radio, una marca de agua sólo necesita sobrevivir a las distorsiones causadas por el proceso de transmisión, incluida la compresión dinámica y un filtro de paso bajo, ya que la detección de marca de agua se realiza directamente desde la señal de emisión.

Detección de marca de agua a ciegas e informada.

En algunas aplicaciones, un algoritmo de detección puede utilizar la señal original para extraer la marca de agua de la secuencia de audio marcada previamente. A menudo, mejora significativamente el desempeño del detector, el audio original se substrae de la copia de marca de agua, resultando en la secuencia de marca de agua. Sin embargo, si el algoritmo de detección no tiene acceso al audio original (detección a ciegas), disminuye considerablemente la cantidad de datos que pueden estar escondidos en la señal huésped. El proceso completo de inserción y extracción de la marca de agua se modela como un canal de comunicaciones de

marca de agua donde se distorsiona debido a la presencia de fuertes interferencias y efectos del canal. Una interferencia es causada por la presencia del audio original, y los efectos del canal corresponden a las operaciones de procesamiento de señal.

Seguridad.

El algoritmo de marca de agua debe ser seguro en el sentido de que un adversario no debe ser capaz de detectar la presencia de datos embebidos, y mucho menos de eliminarlos. La seguridad del proceso de marca de agua se interpreta de la misma manera como el de las técnicas de seguridad de encriptación y no puede romperse a menos que el usuario autorizado tenga acceso a una clave secreta que controla la incrustación de marcas de agua. Un usuario no autorizado no debería extraer los datos en una cantidad razonable de tiempo, incluso si sabe que la señal original contiene una marca de agua y está familiarizado con el algoritmo de inserción. Los requisitos de seguridad varían de acuerdo con la aplicación y los más estrictos son en las aplicaciones de comunicaciones encubiertas, y, en algunos casos, los datos son encriptados antes de la inserción en el audio huésped.

Complejidad computacional y costo.

La aplicación de un sistema de audio de marca de agua es una tarea tediosa, y depende de la aplicación del negocio involucrado. El principal problema desde el punto de vista técnico es la complejidad computacional de los algoritmos de inserción y detección. Por ejemplo, en el control de emisión, la inserción y la detección deben hacerse en tiempo real, mientras que en las de protección de derechos de autor, el tiempo no es un factor crucial para una aplicación práctica. Una de las cuestiones económicas es el diseño de insertores y detectores, que pueden ser implementados como hardware o plug-in[18] de software, y la diferencia

[18] Un plug-in es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy especifica. Esta aplicación adicional es ejecutada por la aplicación principal e interactúan por medio de su API (Interfaz de Programación de Aplicaciones).

de potencia de procesamiento de diferentes dispositivos (Laptop, PDA, teléfono móvil, etc.).

El proceso fundamental en cada sistema de marcas de agua puede ser modelado como una forma de comunicación donde se transmite un mensaje del codificador al receptor de marca de agua. El proceso de la marca de agua es visto como un canal de transmisión a través del cual el mensaje de marca de agua está siendo enviado, con la señal huésped. En la figura 1.2, se presenta una asignación general de un sistema de marcas de agua en un modelo de comunicación. Después que la marca de agua se inserta, ésta suele ser distorsionada después de los ataques. Las distorsiones de la señal de marca de agua son, al igual que el modelo de comunicaciones de datos, el ruido aditivo.

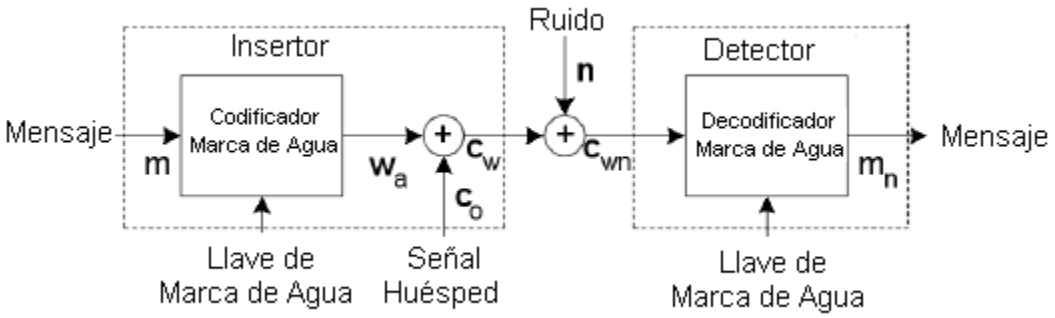

Figura 3.2. Un sistema de marcas de agua y un modelo equivalente de comunicaciones.

CAPÍTULO 4

Marcas de Agua en Audio

"La música empieza donde se acaba el lenguaje"

Ernest Theodor Amadeus Hoffmann

Las marcas de agua de las señales de audio son más difícil en comparación con las de imágenes o secuencias de vídeo, debido a la amplia gama dinámica del sistema auditivo humano (HAS, Human Auditory System) en comparación con el sistema visual humano (HVS, Human Visual System). El HAS percibe sonidos en un rango de potencia superior a $1x10^9$ y una gama de frecuencias superiores a $1x10^3$. La sensibilidad del HAS al Ruido Gaussiano Blanco Aditivo[19] (AWGN, Additive White Gaussian Noise) es alta, este ruido en un archivo de sonido se puede detectar debajo de los 70 dB del nivel del ambiente.

Por otro lado, frente a su amplio rango dinámico, el HAS tiende a enmascarar los sonidos más débiles por los más fuertes. Además, el HAS no es sensible a un cambio de fase relativa constante en una señal de audio estacionaria.

La percepción auditiva se basa en el análisis de banda crítica en el oído interno, donde la transformación se lleva a cabo a lo largo de la membrana basilar. El espectro de potencia de los sonidos recibidos no está representado en una escala de frecuencia lineal, sino en las bandas de frecuencias limitadas llamadas bandas críticas. El sistema auditivo es generalmente modelado como un banco de filtros de paso de banda, que consiste en la superposición de filtros de paso de banda con anchos de banda de alrededor de 100 Hz para las bandas con una frecuencia central y por debajo de 500 Hz, y hasta 5000 Hz para bandas colocadas en las frecuencias altas. Si la mayor frecuencia se limita a 24.000 Hz, 26 bandas críticas tienen que ser tomadas en cuenta.

[19] El término ruido blanco aditivo Gaussiano (AWGN, Additive White Gaussian Noise) se refiere cuando finalmente el ruido se combina con la señal deseada y es un importante factor limitante en la transmisión de información.

Dos propiedades del HAS predominantemente utilizadas en los algoritmos de marcas de agua, son la frecuencia (simultánea) de enmascaramiento y enmascaramiento temporal. El concepto de usar los agujeros de la percepción del HAS es tomado de la codificación de audio de la banda ancha (por ejemplo, la compresión MPEG 1 Layer 3, normalmente llamado mp3). En los algoritmos de compresión, los agujeros se utilizan con el fin de disminuir la cantidad de los bits necesarios para codificar la señal de audio, sin causar una distorsión de la percepción del audio. Por otra parte, en los escenarios de ocultación de información, las propiedades de enmascaramiento se utilizan para integrar bits adicionales en un flujo de bits existentes, sin generar ruido audible en la secuencia de audio utilizado para ocultar los datos.

4.1 Enmascaramiento Frecuencial.

El enmascaramiento frecuencial es la disminución de la sonoridad de un tono a una cierta frecuencia, en presencia de otro tono simultáneo a una frecuencia diferente. Es decir, cuando el oído es expuesto a dos o más sonidos de diversas frecuencias, existe la posibilidad que uno de ellos camufle a los demás y por tanto, que éstos no se oigan.

Este fenómeno perceptivo puede explicarse de manera simplificada considerando como varía la excitación de la membrana basilar del oído según la frecuencia. Esta membrana vibra, en función de la tonalidad, más cerca o más lejos de la ventana oval. A más frecuencia, tonos agudos, el máximo desplazamiento de la membrana basilar es más cercano a la ventana oval que a un tono de baja frecuencia, un tono grave. Esto explica porque este fenómeno no es simétrico. Un tono grave enmascara a uno agudo con más facilidad. Y se pueden dar los siguientes casos:

Frecuencias diferentes con la misma amplitud.

Cuando hay dos tonos en la misma amplitud, se puede comprobar que el tono grave enmascara mucho más al tono agudo. El enmascaramiento no es simétrico respecto la frecuencia.

Baja frecuencia enmascara a alta frecuencia

Cuando un tono grave tiene más nivel de amplitud, aún se hará más claro el enmascaramiento de tonos agudos. En este caso el tono agudo es totalmente enmascarado por el grave.

Alta frecuencia no enmascara a baja frecuencia

En este caso, pese a aumentar la amplitud del tono agudo respecto al tono grave, el primero casi no afecta a al segundo, por tanto el enmascaramiento del tono grave es mínimo.

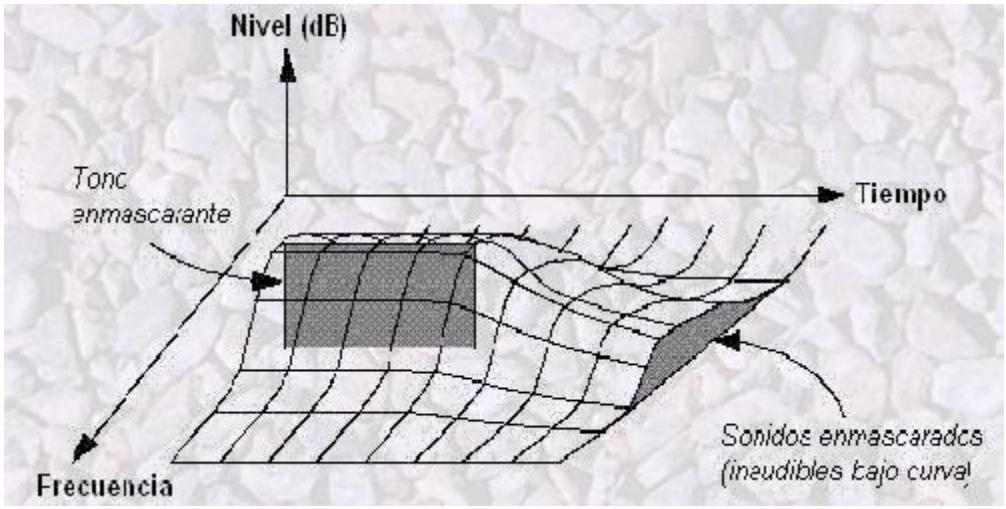

Figura 4.1. Enmascaramiento.

4.2 Enmascaramiento Temporal

El enmascaramiento temporal sucede cuando dos estímulos sonoros llegan a nuestro oído de forma cercana en el tiempo. El estímulo *enmascarante* hará que el otro, *enmascarado*, resulte inaudible. Dada esta situación, el tono más intenso tiende a enmascarar el tono más débil.

Según el instante de tiempo en el que se produce el estímulo enmascarante con respecto al instante en que se produce el enmascarado, se puede distinguir entre *Post-enmascaramiento* y *Pre-enmascaramiento*.

Post-enmascaramiento

Se da cuando es el tono de mayor amplitud el que sucede con antelación en el tiempo al de menor amplitud, percibiéndose tan sólo el primer estímulo. Este fenómeno se produce cuando ambos sonidos llegan al oído en un intervalo de tiempo de entre 30 y 60 ms aproximadamente. Esto se debe a que una vez percibido el tono fuerte, el oído necesita un cierto periodo de adaptación.

Pre-enmascaramiento

Si se produce primero un estímulo suave y posteriormente un tono intenso, este último enmascarará igualmente al de menor amplitud, siempre cuando estén separados en el tiempo por una diferencia menor de entre 5 y 10 ms. Como este fenómeno se presenta incluso antes de que aparezca el tono enmascarante, implica que se trata de un proceso más complejo que el Post-enmascaramiento.

La explicación a esta anticipación se encuentra en que la información que llega a la corteza auditiva del cerebro humano se procesa por ráfagas. Así mismo se sabe que el cerebro procesa los sonidos fuertes más rápido que los débiles, facilitándose de esta forma el pre-enmascaramiento.

Como se puede observar en la figura 4.2, la efectividad del enmascaramiento disminuye de forma exponencial para ambos casos de enmascaramiento temporal.

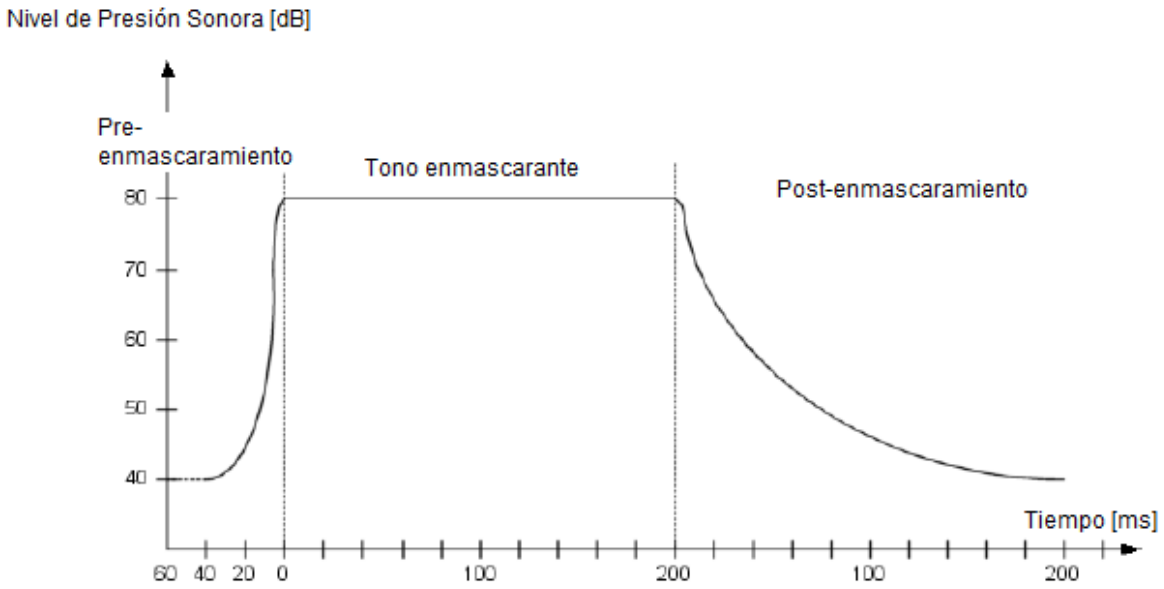

Figura 4.2. Pre – enmascaramiento.

4.3 Umbral de Enmascaramiento

En psicoacústica, el umbral de enmascaramiento es el nivel de presión sonora (SLP) de un sonido de prueba necesario para que éste sea apenas audible en presencia de una señal enmascarante. Este nivel depende también en gran medida de la frecuencia y de las características del enmascarado y del enmascarador. El efecto aparece normalmente entre tonos muy cercanos en frecuencia.

Que no sea audible implica ciertas ventajas en el mundo de las transmisiones. En cuanto a codificación de audio, por ejemplo, implica la posibilidad de pasar por alto dicho tono consiguiendo así una mejor compresión ó, en su alternativa, la codificación con menos peso, es decir, menos bits y por consiguiente reducir el tamaño del fichero resultante.

Habitualmente no se trabaja con un solo tono sino con varios de forma simultánea. Así que para una sola frecuencia se tienen más de una posible señal enmascaradora. Para estas situaciones se calcula el umbral de enmascaramiento global. Éste se cuantifica en base a un espectro de alta resolución de la señal

(habitualmente de audio) a partir de una Transformada rápida de Fourier (FFT) de 512 ó 1024 puntos. En primera instancia se calculan los umbrales individuales teniendo en cuenta el nivel de señal, el tipo de enmascarador (ya sea señal ó ruido) y la banda de frecuencias (hay frecuencias inaudibles para el oído humano). Posteriormente se suman todos los umbrales añadiéndose el umbral de tranquilidad, de esta forma se asegura que el umbral de enmascaramiento total no estará nunca por debajo de este último. Finalmente se puede calcular el SMR (Signal to Mask Ratio). La anterior operación es la que se lleva a cabo en codificación de audio.

En la figura 4.3 se muestra el caso de tener un tono a 1kHz. Se puede observar el umbral de tranquilidad o silencio debajo del cual ningún sonido es perceptible. Ahora bien, al sobreponer el tono este nivel varia alrededor de la frecuencia central del enmascarador haciendo más difícil oír las posibles frecuencias cercanas a éste.

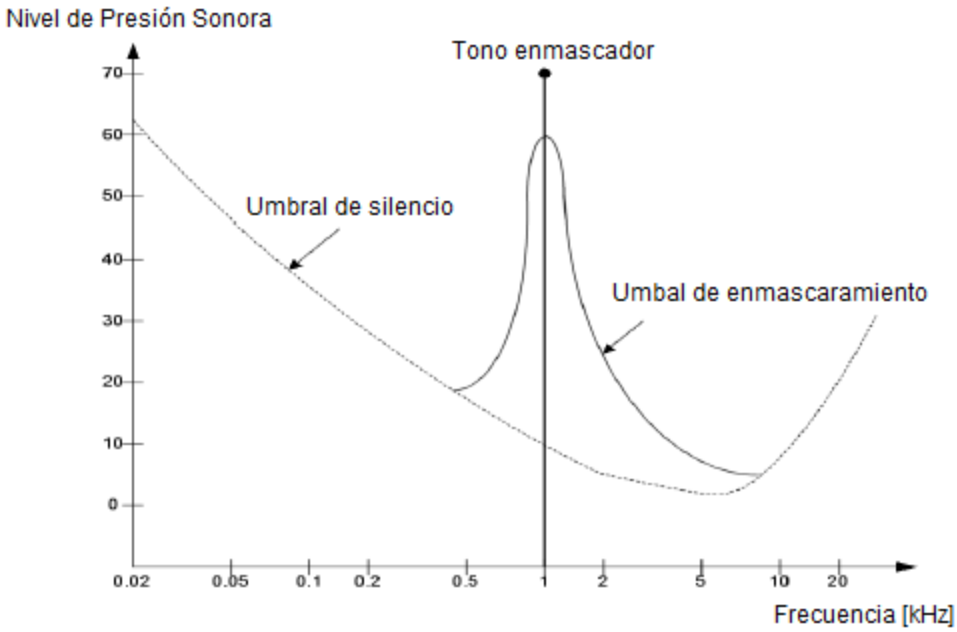

Figura 4.3. Umbral de enmascaramiento.

Aplicaciones del umbral de enmascaramiento

Una aplicación del umbral de enmascaramiento la encontramos en las codificaciones de audio que usa MPEG. En estos esquemas se incluye el bloque denominado 'modelo psicoacústico'. Éste está comunicado con el banco de filtros y el bloque de cuantificación (ó asignación de bits). El modelo psicoacústico es el encargado de analizar las muestras que provienen del banco de filtros calculando, para cada banda, el nivel de enmascaramiento. El procedimiento, como ya se ha comentado en el párrafo anterior, se lleva a cabo mediante un FFT. Según la capa de MPEG en la que estemos trabajando se utilizan más o menos puntos. A partir de todos los distintos niveles se calcula el SMR que se pasa al cuantificador. El cuantificador es el encargado de asignar más o menos bits a cada uno de los bloques frecuenciales teniendo en cuenta el SMR. El bloque con máxima relación señal-enmascaramiento se codificará con el máximo de bits posible mientras que el que tenga la peor relación con el mínimo, llegando a ser cero bits.

En definitiva, el cálculo del umbral de enmascaramiento es tenido en cuenta por ciertos códecs de audio con tal de no codificar muestras que, al fin y al cabo, si se codificaran, serian igualmente inaudibles para el oído humano. De este modo se utilizan menos bits y en consecuencia se reduce el tamaño del archivo de audio consiguiendo así una mejor compresión

4.4 Modelo general de marca de agua digital.

La figura 4.4 da una visión general del modelo general de la marca de agua digital. Un mensaje de marca de agua "*m*" se incrusta en la señal huésped "*x"* para producir la señal de marca de agua "*s"*. El proceso de incrustación depende de la llave K y deberá cumplir el requisito de transparencia perceptual, es decir, la diferencia de calidad subjetiva entre "*x*" y "*s*" (denotada como distorsión embebida d_{emb}) debe estar por debajo del umbral de diferencia perceptible. Antes de que la detección de marcas de agua y el proceso de decodificación se lleve a cabo, "*s*" es usualmente modificada, ya sea intencionalmente o no. Las modificaciones intencionales son usualmente referidas como ataques, un ataque produce una

Distorsión de Ataque d_{att} a un nivel perceptualmente aceptable. Después de los ataques, un extractor de marca de agua recibe la señal de ataque r.

El proceso de extracción de la marca de agua consiste en dos sub-procesos, en primer lugar, el decodificador descifra un mensaje con marca de agua usando la llave K, y, en segundo lugar, la detección de marca de agua, es decir, la prueba de hipótesis entre:

Hipótesis H0: los datos recibidos "*r*" no es una marca de agua con la llave K.

Hipótesis H1: los datos recibidos "*r*" es una marca de agua con la llave K.

Dependiendo de la aplicación, el detector realiza una detección informada o a ciegas. El término ataque requiere algunas aclaraciones. La señal "*s*" puede ser modificada sin la intención de afectar la marca de agua integrada (por ejemplo, la compresión dinámica de la amplitud de audio antes de la radiodifusión). ¿Por qué es este tipo de proceso de señal se llama un ataque? La primera razón es para simplificar la notación del modelo general de marca de agua digital. La otra, una razón aún más importante, es que cualquier proceso de señal común menoscaba una marca de agua, esto será un método potencial aplicado por los adversarios que, intencionadamente intenten quitar la marca de agua. Los algoritmos deben estar diseñados para soportar el peor de los posibles ataques de una distorsión de ataque d_{att}, lo que podría ser incluso una operación común de procesamiento de señales (por ejemplo, la compresión dinámica, filtrado de paso bajo, etc.).

La separación entre la decodificación y la detección durante la extracción de la marca de agua deben estar claramente definidos. Por lo tanto, es importante diferenciar entre comunicar un mensaje de marca de agua "*m*" (embeber y decodificar una marca de agua digital) y verificar si los datos recibidos "*r*" es una marca de agua o no (detección de marca de agua). A primera vista, la decisión entre la hipótesis H0 y H1 (detección de marca de agua) aparece como un caso especial de decodificación de un mensaje binario $m \in \{0,1\}$. Este no es el caso, ya que la señal marcada y la señal recibida, tienen alguna composición especial para

m = 0 y otra estructura especial para m = 1. Sin embargo, en la hipótesis H0 del problema de detección, los datos recibidos pueden tener cualquier estructura o, equivalentemente, sin estructura alguna.

La importancia de la llave K tiene que ser enfatizado. Las marcas de agua incorporadas deben ser protegidas contra la detección, el descifrado, la eliminación o modificación, y la modificación de atacantes. Kerckhoff afirma que la seguridad de un sistema de cifrado ha de residir sólo en la llave de un sistema, así que tiene que ser aplicado en la seguridad de un sistema de marcas de agua. Por lo tanto, se debe suponer que los algoritmos de inserción y extracción son de conocimiento público, pero sólo ciertas partes conocen la llave adecuada para recibir y modificar la información incluida. La llave K se considera un número entero grande, con una longitud de palabra de 64 bits a 1024 bits.

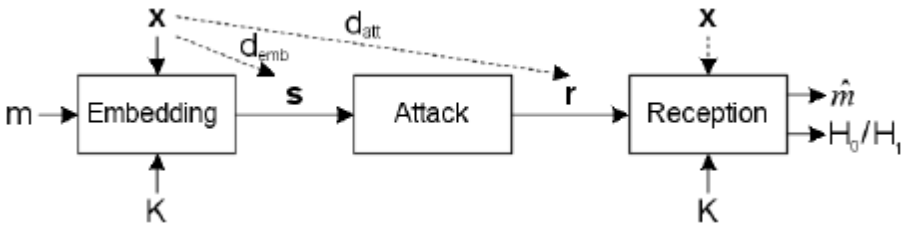

Figura 4.4. Modelo general de marca de agua digital.

4.5 Decodificación y detección.

El objetivo final de cualquier marca de agua es un algoritmo de extracción confiable. En general, la fiabilidad de extracción de un régimen específico se basa en las características de los datos originales, en la distorsión embebida d_{emb} y en la distorsión de ataque d_{att}. La fiabilidad de extracción de una marca de agua normalmente se analiza para los diferentes niveles de distorsión ataque d_{att}, características de datos y la distorsión embebida d_{emb}. Diferentes medidas de fiabilidad se utilizan para descifrar y detectar marcas de agua.

Decodificador de Marca de Agua

En la evaluación del desempeño de la decodificación, se considera como un problema de comunicación. Un mensaje "*m*" se incrusta en la señal huésped "*x*" y debe ser descifrable de la señal "*r*". Las tasas de error sólo se pueden lograr utilizando códigos de corrección de errores[20]. Para un escenario practico de corrección de errores en la codificación, el mensaje de marca de agua es generalmente codificado en un vector "*b*" de longitud "*Lb*" con elementos binarios "$b_n = 0, 1$". Por lo general, "*b*" es llamado el mensaje de marca de agua, y el mensaje decodificado llamado "$b^{\wedge}$". La fiabilidad de decodificación de "*b*" puede ser descrito por la probabilidad de error de palabra[21] (WEP, Word Error Probability):

$$P_w = P_r(m \neq m^{\wedge}) = P_r(b \neq b^{\wedge})$$

o por la probabilidad de error de bit[22] (BEP, Bit Error Probability):

[20] En teoría de la información y la teoría de codificación con aplicaciones en ciencias de la computación y las telecomunicaciones, la detección y corrección de errores o de control de errores son técnicas que permiten la entrega confiable de datos digitales a través de los canales de comunicación fiables. Algunos canales de comunicación están sujetos a canal de ruido, y por lo tanto los errores pueden ser introducidos durante la transmisión de la fuente a un receptor. técnicas de detección de errores permiten detectar este tipo de errores, mientras que la corrección de errores permite la reconstrucción de los datos originales.

Las definiciones generales de los términos son los siguientes:

* Detección de error es la detección de los errores causados por el ruido u otras discapacidades durante la transmisión desde el transmisor al receptor.

* Corrección de errores es la detección de errores y la reconstrucción de los datos originales, sin errores.

[21] Se define la probabilidad de error de palabra como la probabilidad de descifrar a favor de una palabra clave "x" cuando la palabra clave "X" se transmitió, entre todas las palabras posibles que se transmitieron.

La tasa de error de la palabra, es la probabilidad de selección en el decodificador de una palabra clave diferente a la que se transmitió.

[22] En la transmisión digital, el número de errores de bits es el número de bits recibidos de una corriente de datos a través de un canal de comunicación que han sido alterados debido al ruido, interferencias, distorsiones o errores en los bits de sincronización.

$$P_b = \sum_{n=1}^{l_b} p_r(b_n \neq \hat{b_n})$$

El WEP y BEP se pueden calcular para determinados modelos estocásticos de todo el proceso de marca de agua, incluidos los ataques. Las probabilidades de error previstos pueden ser confirmados experimentalmente por un gran número de simulaciones con diferentes realizaciones de la llave de marca de agua "*K*", la señal huésped "*x*", los parámetros de ataque y un mensaje de marca de agua "*m*". El número de eventos de error de medida dividido por el número de los hechos observados definen los índices de la tasa de error, la tasa de error de la palabra (WER), y la tasa de error de bit (BEP).

Los límites de rendimiento pueden ser obtenidos con métodos tomados de la teoría de la información. Por ejemplo, el tipo de marca de agua máxima que puede recibir, en principio, sin errores, está determinado por la información mutua $I(r|m)$ entre el mensaje transmitido "*m*" marca de agua y los datos recibidos "r" y esta dado por:

$$I(r|m) = h(r) - h(r|m)$$

donde $h(r)$ es la entropía diferencial de "*r*", y $h\,(r|m)$ es la entropía diferencial de "*r*" condicionada por el mensaje transmitido "*m*". $I(r|m)$ sólo se puede lograr para un número infinito de elementos de datos. Para un número finito de elementos de datos, una probabilidad de error de palabra P_w o una probabilidad de error de bit P_b diferentes a cero, son inevitables.

La capacidad de canal "*C*" de un canal de comunicación específica se define como el máximo de información mutua $I(r;m)$[23] sobre todos los sistemas de transmisión

---

La tasa de error o la relación de error de bit (BER, Bit Error Rate) es el número de errores de bit dividido por el número total de bits transferidos durante un intervalo de tiempo estudiado. BER es una medida de rendimiento sin unidades, a menudo se expresa como un número porcentual.

[23] Hacer referencia a la teoría de la información de Shannon
http://www.scholarpedia.org/article/Mutual_information

con una potencia de transmisión limitada a un valor fijo. La capacidad de marca de agua "*C*" se define en consecuencia con una ligera modificación específica. El análisis de la capacidad proporciona un buen método para comparar los límites de desempeño de los diferentes escenarios de comunicación. Dado que aún no existe una solución para el problema general de marca de agua, suele ser analizada dentro de ciertas limitaciones a la integración y ataques. Además, para diferentes escenarios, la capacidad de marca de agua puede depender de diferentes parámetros (dominio de integración, parámetros de ataque, etc.).

Detección de marca de agua.

La detección de la marca de agua se define como la decisión de si los datos recibidos "*r*" es una marca de agua (como en la hipótesis en el punto 4, H1) o no (como en la hipótesis en el punto 4, H0).

En general, ambas hipótesis no se puede separar perfectamente. Por lo tanto, definimos la probabilidad Pf_p (falsos positivos) en el caso de aceptar H1 cuando H0 es verdadera y la probabilidad Pf_n de aceptar H0 cuando H1 es verdadera (falso negativo).

En muchas aplicaciones, la prueba de hipótesis deben ser diseñados para garantizar un número limitado de probabilidad de falsos positivos, por ejemplo, $Pf_p < 10^{-12}$ para la detección de marca de agua en el contexto de la protección de copia de DVD. Otra opción para la evaluación de la detección de marca de agua es la investigación de la probabilidad de detección de error total P_e, que mide los dos tipos de errores posibles.

4.6 Seleccionado algoritmos de audio de marca de agua

Los algoritmos de marca de agua se han desarrollado principalmente para las imágenes digitales y secuencias de vídeo, el interés e investigación en audio comenzó un poco más tarde. En los últimos años, varios algoritmos para la inserción y extracción en las secuencias de audio se han presentado. Todos los

algoritmos desarrollados toman ventaja de la propiedad de percepción del sistema auditivo humano (HAS, Human Auditory System) con el fin de añadir una marca de agua en una señal huésped de una manera perceptivamente transparente. Una amplia gama de técnicas de inserción va desde el bit menos significativo (LSB) hasta varios métodos de espectro (spread spectrum).

Bit menos significativo (LSB)

Esta es la manera más sencilla de esconder información en un archivo de audio. Sustituyendo el bit menos significativo de cada punto de muestra con un mensaje, permite ocultar una gran cantidad de información. La figura 5 ilustra como el mensaje “HEY” es codificado en una muestra con calidad de CD de 16-bit usando el método LSB.

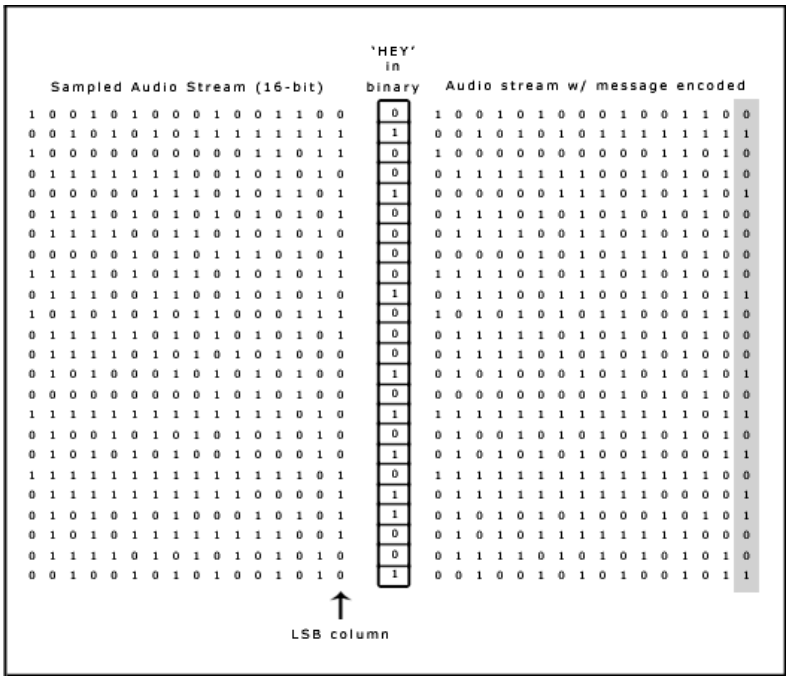

Figura 4.5. Ocultamiento LSB

En el LSB, la velocidad de transmisión de datos ideal es 1 kbps por 1kHZ. En algunas implementaciones, los dos últimos bits menos significativos de una muestra son reemplazados con dos bits del mensaje a ocultar. Esto aumenta la cantidad de datos que se pueden ocultar, pero también aumenta la cantidad de

ruido al archivo de audio. Por lo tanto, se debe tener en cuenta el tipo de aplicación que se requiere implementar. Por ejemplo, el ruido de un archivo de sonido grabado en una calle congestionada disfrazaría mejor el ruido agregado con LSB. Por otra parte, el mismo ruido seria fácilmente detectable en un archivo de sonido que contiene una pieza de música clásica.

Para extraer el mensaje oculto, el receptor debe acceder a la secuencia de índices de muestras usados en el proceso de ocultamiento. Normalmente, la longitud del mensaje secreto es menor que el numero total de muestras en un archivo de sonido. Es posible decidir como se escogerán el subconjunto de muestras que contendrán el mensaje secreto y comunicar esa decisión al receptor. Una técnica trivial consiste en empezar en el principio del archivo de sonido y realizar el proceso LSB hasta que el mensaje haya sido ocultado completamente, dejando intactas las muestras restantes. Esto crea un problema de seguridad, debido a que la primera parte del archivo de sonido tendrá propiedades estadísticas diferentes a la segunda parte que no fue modificada. Una solución a este problema es completar el mensaje secreto con bits aleatorios para que su longitud sea igual al número total de muestras. Ahora el proceso de ocultamiento modifica muchas más muestras que las requeridas para la transmisión del mensaje secreto, lo que aumente la probabilidad de que un atacante sospeche de la comunicación secreta.

Un enfoque más sofisticado es usar un generador de números pseudos-aleatorios para esparcir el mensaje dentro del archivo de sonido de una manera aleatoria. Receptor y emisor deben compartir una clave secreta usada como semilla en un generador de números pseudos-aleatorios para crear una secuencia aleatoria de índices de muestras.

Codificación de Fase (*Phase Coding*)

La codificación de fase aborda las desventajas de los métodos esteganográficos de inducción de ruido. Se basa en que los *componentes de fase* del sonido son menos perceptibles al oído humano que el ruido. En vez de introducir

perturbaciones, esta técnica codifica los bits del mensaje como desplazamiento de fase en el espectro de fase de una señal digital.

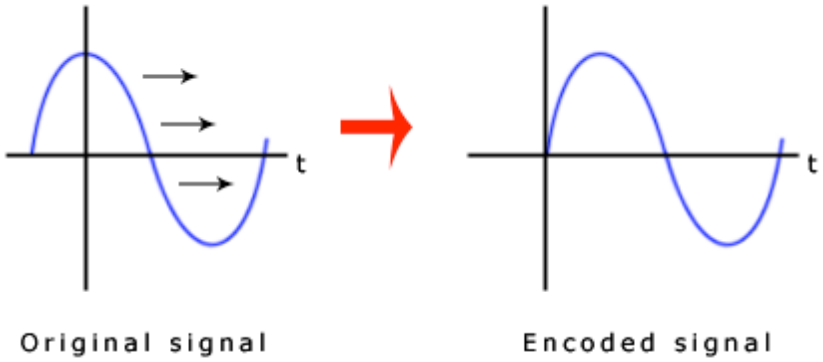

Figura 4.6. Codificación de Fase.

La codificación de fase es explicada en el siguiente procedimiento:

1. La señal original es dividida en segmentos más pequeños para igualar la longitud del mensaje a ser codificado.
2. Una Transformada de Fourier Discreta (DFT) es aplicada a cada segmento para crear una matriz de las fases y magnitudes de la transformada de Fourier.
3. Diferencias de fase entre segmentos adyacentes son calculadas.
4. Desplazamiento de fases entre segmentos consecutivos son fácilmente detectadas. En otras palabras, las fases absolutas de los segmentos pueden ser cambiadas pero la diferencia de fase relativa entre segmentos adyacentes debe ser preservada. Por lo tanto el mensaje secreto es insertado únicamente en el vector de fase del primer segmentos como se muestra a continuación:

$$phase\_new = \begin{cases} \pi/2 & if \quad message \quad bit = 0 \\ -\pi/2 & if \quad message \quad bit = 1 \end{cases}$$

5. Una nueva matriz de fase es creada usando la nueva fase del primer segmento y las diferencias originales de fase.
6. Usando la nueva matriz de fase y la matriz de magnitud original, la señal de sonido es reconstruida aplicando la inversa DFT y luego concatenando los segmentos de sonido.

Para extraer el mensaje secreto, el receptor debe conocer la longitud de segmento. El receptor puede usar la DFT para obtener las fases y extraer la información.

Una desventaja asociada con la codificación de fase es la baja tasa de transmisión de datos debido a que el mensaje secreto es codificado únicamente en el primer segmento de la señal. Esto puede ser resuelto incrementando la longitud del segmento inicial. Sin embargo, esto cambiaria la relación de fase entre cada componente de frecuencia de una manera más drástica, permitiendo que la codificación sea más fácil de detectar. Como resultado, el método de codificación de fase es usado sólo cuando se necesita esconder una pequeña cantidad de datos.

Codificación en el Eco (*Echo Hiding*)

En Codificación en el Eco, información es añadida en un archivo de sonido introduciendo un eco en la señal discreta. Como el método Espectro Ensanchado, permite alta velocidad de transmisión de datos y provee una robustez superior en comparación con los métodos de inducción de sonido.

Para esconder los datos satisfactoriamente, tres parámetros del eco son modificados de la señal original: amplitud, velocidad de caída, y retardo *(offset)*. Estos tres parámetros son establecidos por debajo del umbral de audición humana. El *offset* es modificado para representar el mensaje binario a ser ocultado.

Si se produce sólo un eco de la señal original, sólo un bit de información puede ser codificado. Por lo tanto, la señal original es dividida en bloques antes de que el proceso de codificación empiece. Una vez concluido el proceso de codificación, los bloques son concatenados para crear la señal final.

Como ejemplo, ocultaremos el mensaje “HEY” usando Codificación en el Eco*:*

Primero la señal es dividida en bloques, y luego a cada bloque se le asigna un bit del mensaje secreto.

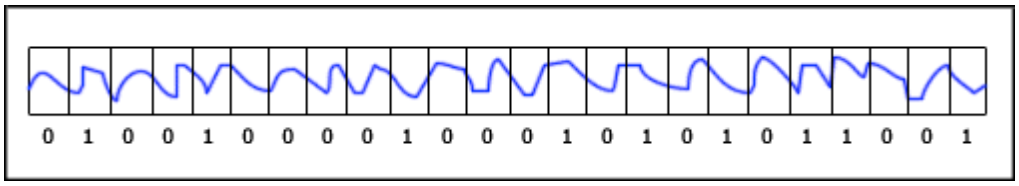

Figura 4.7. División de la señal de audio en bloques.

Por último los bloques son combinados para producir la señal final.

Usando esta implementación de Codificación en el Eco puede resultar en una señal en donde se nota la mezcla de ecos, lo que incrementa el riesgo de detección. Una segunda implementación de Codificación en el Eco resuelve este problema. Primero una señal de eco es creada a partir de la señal original completa usando un valor binario 0 como valor de *offset.* Luego una señal de eco es creada a partir de la señal original completa usando un valor binario 1 como valor de *offset.* Para combinar estos dos ecos, dos mezcladores de señal son usados. Los mezcladores de señal tienen un valor de uno o cero dependiendo del bit de la señal original (ver Figura 11). En nuestro ejemplo usando el mensaje “HEY”, obtenemos los dos mezcladores de señal siguientes:

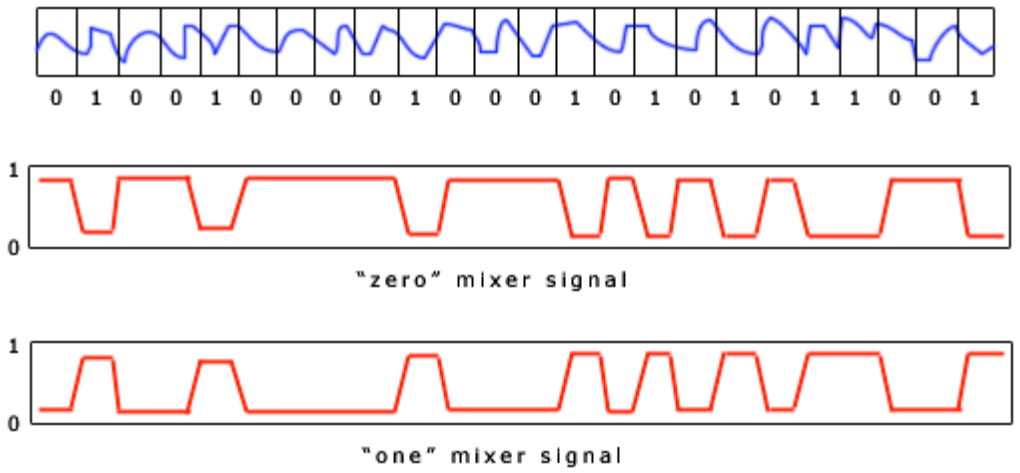

Figura 4.8. Codificación en el Eco I.

La señal de eco “1” es entonces multiplicada por el mezclador de señal “one” y la señal de eco “zero” es multiplicada por el mezclador de señal “zero”. Luego los dos resultados son unidos para obtener una señal final menos abrupta que la obtenida con la primera implementación de Codificación en el Eco. Esto ocurre por lo que dos mezcladores son complemento del otro, produciendo transiciones más suaves entre ecos.

El siguiente diagrama resume la segunda implementación de Codificación en el Eco:

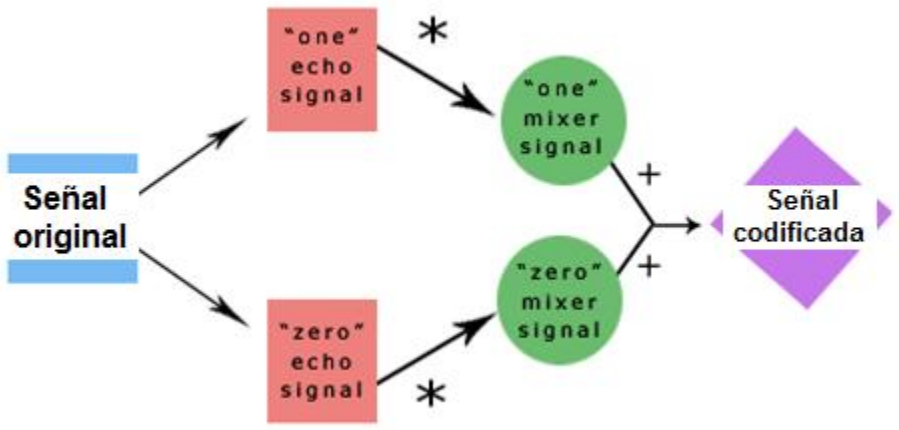

Figura 4.9. Codificación en el Eco II.

Propagación de espectro (*Spread Spectrum)*

El método básico de *spread spectrum* (SS) trata de esparcir lo máximo posible la información secreta a través de el espectro de la señal de audio. Este método es análogo a la implementación de codificación LSB aleatoria. Sin embargo, a diferencia de la codificación LSB, el método SS esparce el mensaje usando un código que es independiente de la señal de audio, lo que hace que la señal de audio use un excesivo ancho de banda, mucho más de lo que es realmente requerido para su transmisión.

Dos versiones de SS pueden ser usadas: secuencia directa y salto de frecuencia. En secuencia directa, el mensaje secreto es esparcido por una constante llamada *chip rate* y luego modulada con una señal pseudo-aleatoria. Luego se intercala con el mensaje de cubierta. En salto de frecuencia, el espectro de frecuencia del archivo de audio es alterado de tal manera que salte rápidamente entre frecuencias. El siguiente diagrama ilustra el sistema SS aplicado específicamente a la esteganografía de audio:

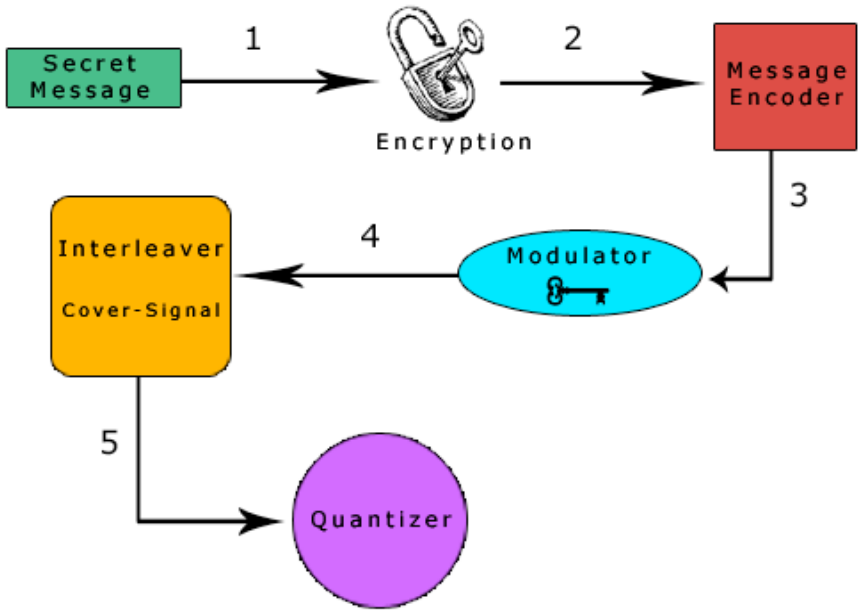

Figura 4.10. Espectro Ensanchado.

1. El mensaje secreto es cifrado usando una clave simétrica.
2. El mensaje cifrado es codificado usando un código de corrección de error de baja frecuencia. Este paso aumenta la robustez del sistema.
3. El mensaje cifrado es modulado usando una señal pseudo-aleatoria generada usando una segunda clave simétrica como semilla.
4. La señal resultante del paso 3 es intercalada con el mensaje de cubierta.
5. La señal final es *cuantificada* para crear un nuevo archivo de audio que contiene el mensaje.
6. La inversa del proceso sirve para extraer el mensaje.

El método SS tiene el potencial de desempeñarse mejor en algunas áreas que LSB, codificación de paridad, y codificación de fase, debido a que ofrece una tasa de transmisión de data moderada y al mismo tiempo mantiene un alto nivel de robustez contra técnicas de eliminación. Sin embargo, al igual que LSB y codificación de paridad, SS pueden introducir ruido en un archivo de sonido.

CAPÍTULO 5

Aplicación de Marca de Agua con el método “Bit Menos Significativo” (LSB)

“La música es un eco del mundo invisible”

Giuseppe Mazzini

La siguiente aplicación demuestra el uso de la Marca de Agua con el método “Bit Menos Significativo” o ”Least significant bit”, por sus siglas en inglés, en un archivo de sonido, específicamente formato .WAV. Como en capítulos anteriores se muestra, este método intercambia el primer bit de un byte de información, como se muestra en la siguiente figura:

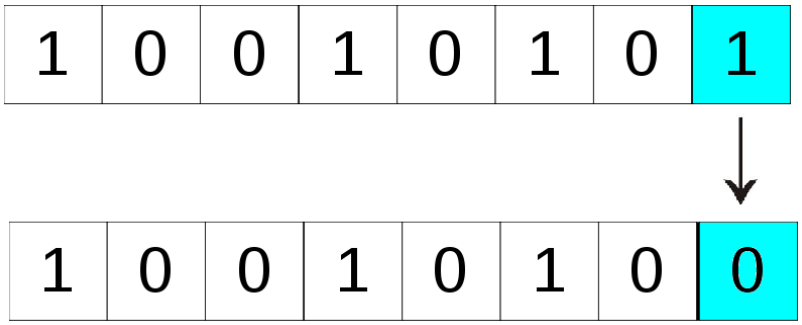

Figura 5.1. Cambio de 1 a 0 en el bit menos significativo.

A continuación presento un diagrama con las señales que intervienen para la inserción de la Marca de Agua. El bloque Watermark Embedder es el que se encarga de la extracción de los datos de audio y la inserción de los datos del mensaje oculto:

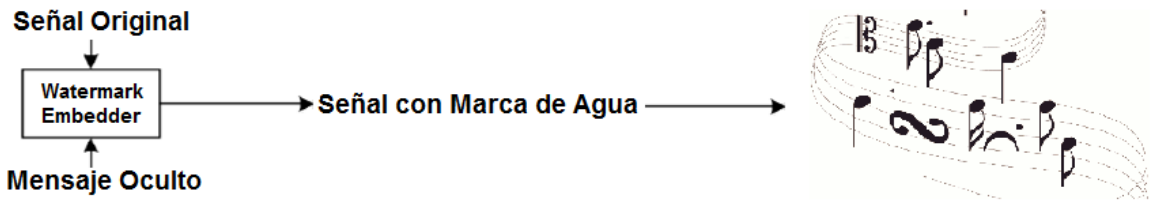

Figura 5.2. Señales que intervienen en la Marca de Agua.

Como sugieren los expertos, utilizar capas múltiples de seguridad disminuye el riesgo o vulnerabilidad de la Marca de Agua en caso de ataque. En esta aplicación utilicé como capa de seguridad al esquema de cifrado AES[24] (Advanced Encryption Standard), que es un algoritmo de cifrado para asegurar el material sensible, en este caso nuestro mensaje que vamos a insertar. En el siguiente diagrama a bloques representa los pasos que seguí para la obtención de datos e inserción de la Marca de Agua sin la capa de seguridad mencionada anteriormente, esto es, porque quiero presentar como es el diagrama sin esta capa, ya que solo es un método que ayuda a la seguridad de la marca de Agua:

[24] También conocido como Rijndael, es un esquema de cifrado por bloques adoptado como un estándar de cifrado por el gobierno de los Estados Unidos. Era usado para información secreta (Top Secret) de los Estados Unidos. Para romperlo se necesitarían 2^{120} operaciones mediante el ataque de fuerza bruta.

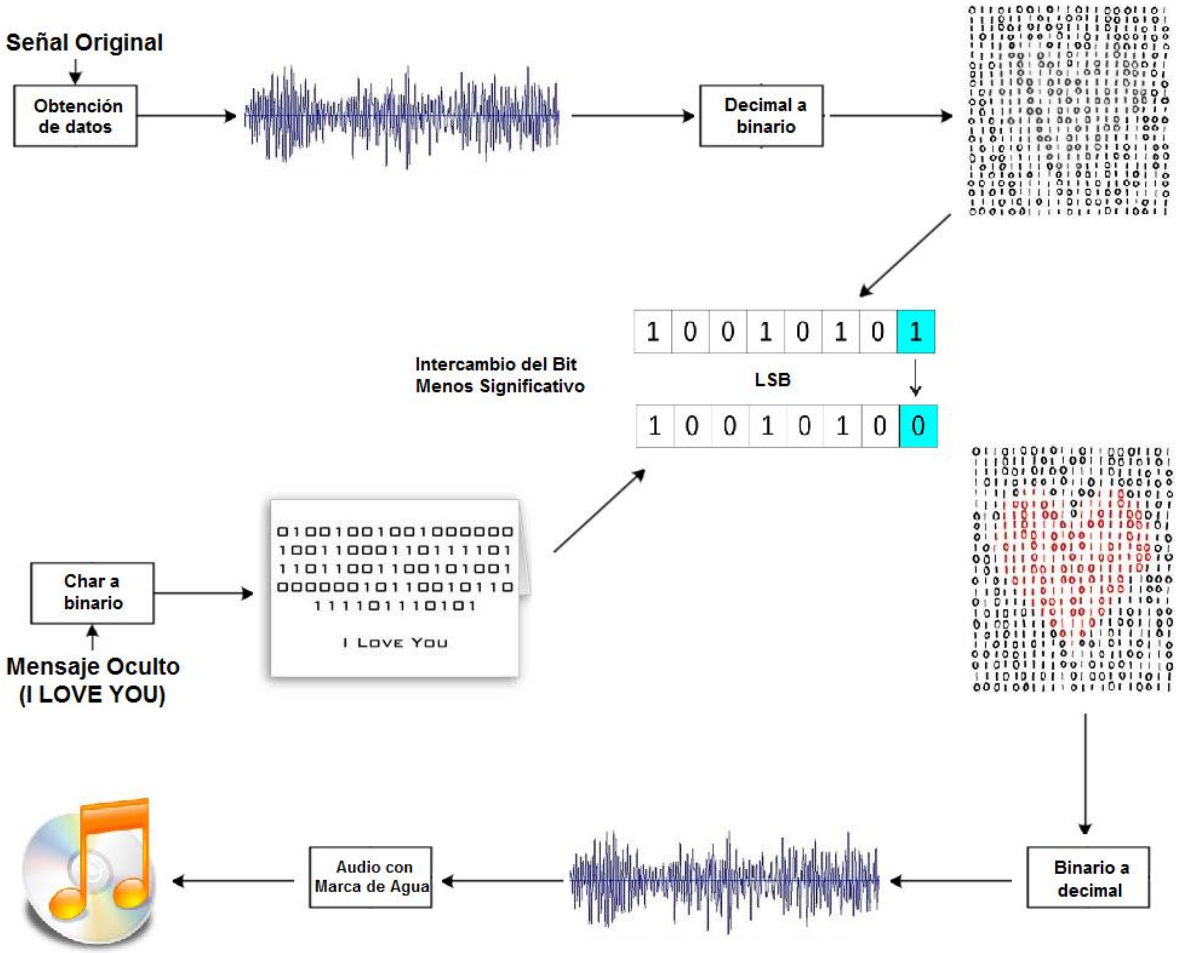

Figura 5.3. Proceso de lectura de datos e inserción de Marca de Agua.

La Figura 5.4 muestra el diseño que implementé con la capa de seguridad con AES, que incrementa la seguridad de la Marca de Agua:

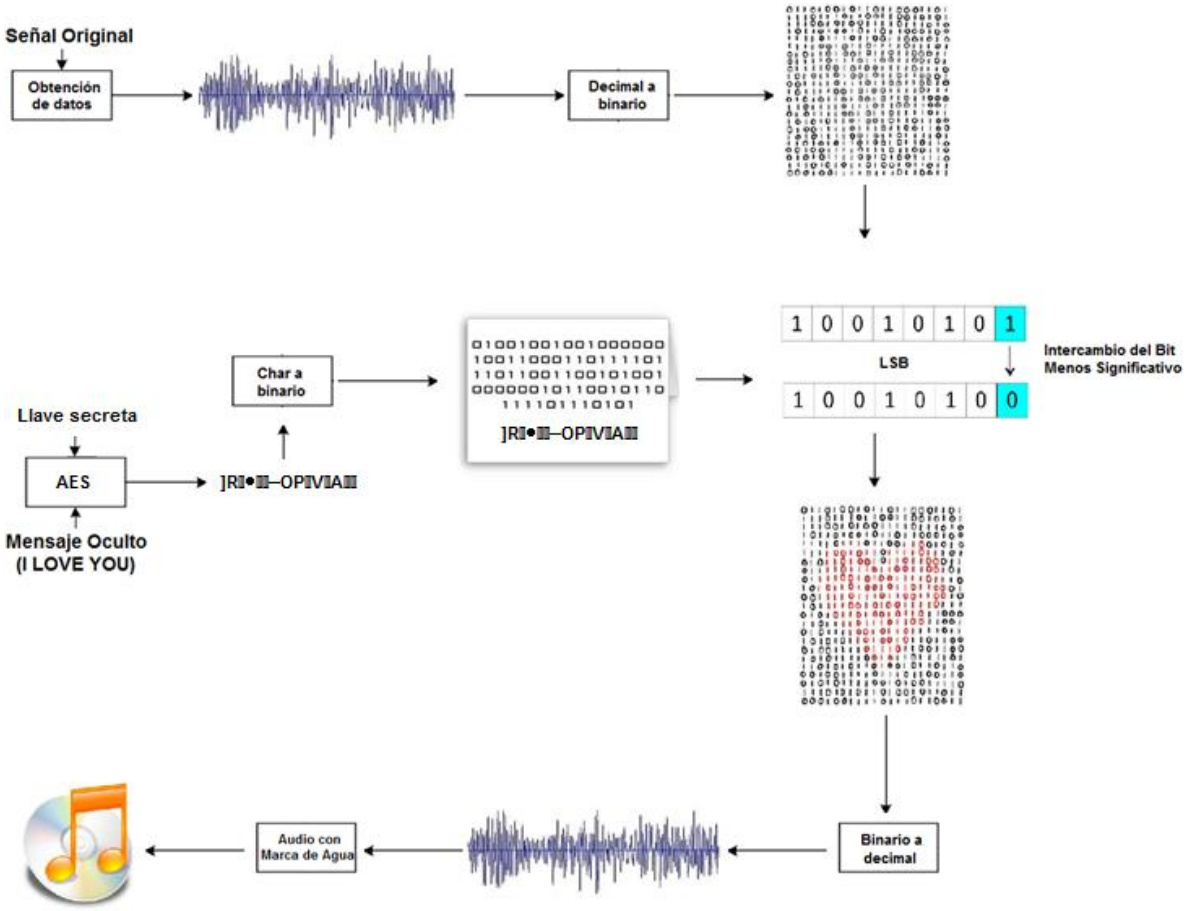

Figura 5.4. Proceso de inserción de Marca de Agua con capa de seguridad.

Para esta aplicación utilicé los siguientes lenguajes de programación:

- PHP[25] *Hypertext Pre-processor.*
- HTML[26] HyperText Markup Language (Lenguaje de Marcado de Hipertexto).
- Javascript[27].

Utilicé estos lenguajes Web, ya que son portables y se pueden utilizar perfectamente para realizar una Marca de Agua.

Siguiendo el proceso de la Figura 5.4 desarrollaré cada uno de los bloques para realizar una Marca de Agua.

5.1 Estructura de un archivo WAV

El formato de archivo WAV ó WAVE es el formato de archivo nativo de Windows para almacenar datos de audio digital. Se ha convertido en uno de los formatos de audio digital más soportados en el PC, debido a la popularidad de Windows y el gran número de programas para la plataforma.

Un archivo WAVE se compone de un fragmento “WAVE” que a su vez tiene dos sub bloques, un fragmento “fmt” que especifica el formato de datos y un fragmento “data” que contiene los datos reales de la muestra.

En la siguiente tabla se describe la estructura de un archivo WAV, cuyo conocimiento y manejo es clave para la implementación de la aplicación de una Marca de Agua. Este formato es uno de los más utilizados para almacenar sonidos, se trata de obtener las muestras en un arreglo y manejar estos datos de

[25] PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Se usa principalmente para la interpretación del lado del servidor (*server-side scripting*).
[26] HTML es un lenguaje de marcado predominante para la elaboración de páginas web.

[27] JavaScript es un lenguaje de programación interpretado. Se define como orientado a objetos, se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.

audio para hacer la inserción de la Marca de Agua. La sencillez de este formato lo hace ideal para el tratamiento digital de audio, y con el lenguaje PHP podemos manipularlo debidamente.

Bytes	Nombre del campo	Contenido usual	Propósito/Descripción
00 – 03	**ChunkID**	RIFF	Bloque de identificación
04 – 07	**ChunkSize**	Datos	Entero largo. Tamaño del fichero en bytes, incluyendo cabecera. 36 + SubChunk2Size, o más precisamente: 4 + (8 + SubChunk1Size) + (8 + SubChunk2Size)
08 – 11	**Format**	WAVE	Otro identificador.
12 – 15	**Subchunk1ID**	fmt	Otro identificador.
16 - 19	**Subchunk1Size**	16, 0, 0, 0	16 para la PCM. Este es el tamaño del resto de este Sub fragmento.
20 - 21	**AudioFormat**	1, 0	PCM = 1 (es decir, la cuantificación lineal) Los valores distintos a 1 indican algún tipo de compresión.
22 - 23	**NumChannels**	1, 2,…	Mono = 1, Estéreo = 2, etc.
24 – 27	**SampleRate**	Datos	Frecuencia de muestreo (muestras/segundo).
28 – 31	**ByteRate**	Datos	Número medio de bytes/segundo. Este valor indica la cantidad de bytes del archivo WAVE que se transmitirán a un convertidor D/A (Digital a Analógico) por segundo para poder reproducir el archivo. $\frac{SampleRate * NumChannels * BitsPerSample}{8}$
32 – 33	**BlockAlign**	Datos	Alineamiento de bloque. El número de bytes por parte de la muestra. Este valor no se ve afectada por el número de canales.

			$\frac{NumChannels * BitsPerSample}{8}$
34 – 35	**BitsPerSample**	8, 16, 24, 32	Este valor especifica el número de bits utilizados para definir cada muestra.
36 – 39	**Subchunk2ID**	data	Marcador que indica el comienzo de los datos de las muestras.
40 - 43	**Subchunk2Size**	Datos	Número de bytes muestreados.
*	**Data**	Datos	Contiene los datos de audio digital de la muestra.

Tabla 5.1. Formato WAV.

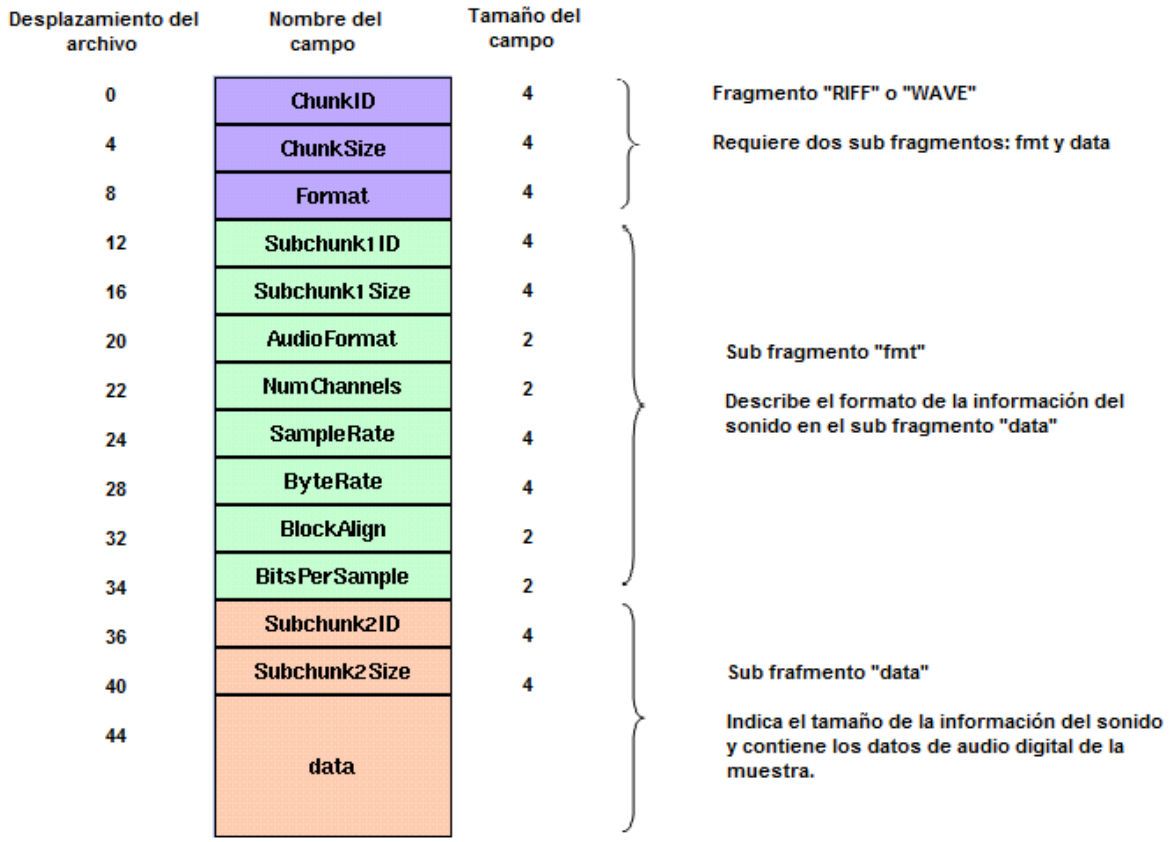

Figura 5.5. Muestra la forma canónica del formato WAV.

5.2 Extracción de datos del archivo WAV

Conociendo el formato WAV podemos obtener sus datos para dar comienzo a la inserción de la Marca de Agua.

Primeramente abrimos el archivo de sonido y leemos los primeros 44 bytes de información del archivo, los cuales corresponden a las cabeceras donde reside el tipo y el formato del fragmento de datos.

```
$handle = fopen ($_FILES['file']['tmp_name'], "r");
$cabeceras[] = fread ($handle, 4);
$cabeceras[] = fread ($handle, 4);
$cabeceras[] = fread ($handle, 4);
$cabeceras[] = fread ($handle, 4);
$cabeceras[] = fread ($handle, 4);
$cabeceras[] = fread ($handle, 2);
$cabeceras[] = fread ($handle, 2);
$cabeceras[] = fread ($handle, 4);
$cabeceras[] = fread ($handle, 4);
$cabeceras[] = fread ($handle, 2);
$cabeceras[] = fread ($handle, 2);
$cabeceras[] = fread ($handle, 4);
$cabeceras[] = fread ($handle, 4);
```

La función “fopen” abre el archivo y con “fread” recorre el archivo dependiendo los bytes que queremos obtener, así obtenemos los 44 bytes y lo guardamos en un arreglo, si vemos detenidamente, seguimos la forma canónica en la Figura 5.5. en el tamaño del campo. Seguidamente obtenemos los datos de audio digital.

```
while(!feof($handle)){
    $data[] =fgetc($handle);
}
```

Con el ciclo “while” recorremos el archivo desde el byte 44 hasta el fin del archivo, el cual se muestra con “feof” (file – end – of - file) y obtenemos cada byte con “fgetc”. Hasta aquí tenemos los datos de audio en un arreglo para manipularlos, pero los datos no están normalizados como queremos, ya que son bytes en código ASCII por la función “fgetc” la cual nos devuelve un char, así que tenemos que

convertir cada byte en ceros y unos, es decir, 8 bits. Esto lo hacemos con la siguiente función que realicé:

```
function asc2bin ($ascii){
    $cont=0;
    while ( strlen($ascii) > 0 )
    {
        $byte = "";
        $i = 0;
        $byte = substr($ascii, 0, 1);
        while ( $byte!= chr($i) ) { $i++; }
        $byte = base_convert($i, 10, 2);
        $byte = str_repeat("0", (8 - strlen($byte)) ) . $byte;
        $ascii = substr($ascii, 1);
        $binary[$cont]= $byte;
        $cont++;
    }
    return $binary;
}
```

Esta función toma cada carácter y lo asigna a la variable $byte = substr($ascii,0,1), posteriormente obtiene el número que representa el carácter ASCII con la siguiente expresión: while($byte != chr($i)) { $i++; }; convertimos el número decimal que representa ese carácter a binario con la función base_convert($i, 10, 2), después llenamos de ceros el carácter binario, si es que lo necesita, por ejemplo:

Tomando en cuenta la tabla ASCII de la Figura 5.6. Si un byte de audio digital que obtuvimos representa el carácter #, representa en decimal el número 35 de la tabla ASCII, después el número 35 lo convertimos en binario y obtenemos el binario “100011”, como observamos el binario que obtuvimos es de 7 bits, necesitamos normalizarlo a un byte, es decir, 8 bits, así que con la función str_repeat(“0”, (8 – strlen($byte))).$byte logramos normalizar y obtenemos el byte “0100011”. Finalmente, le sustraemos el carácter leído substr($ascii, 1), lo guardamos en un array y se repite para todos los elementos no normalizados del array de datos de audio digital.

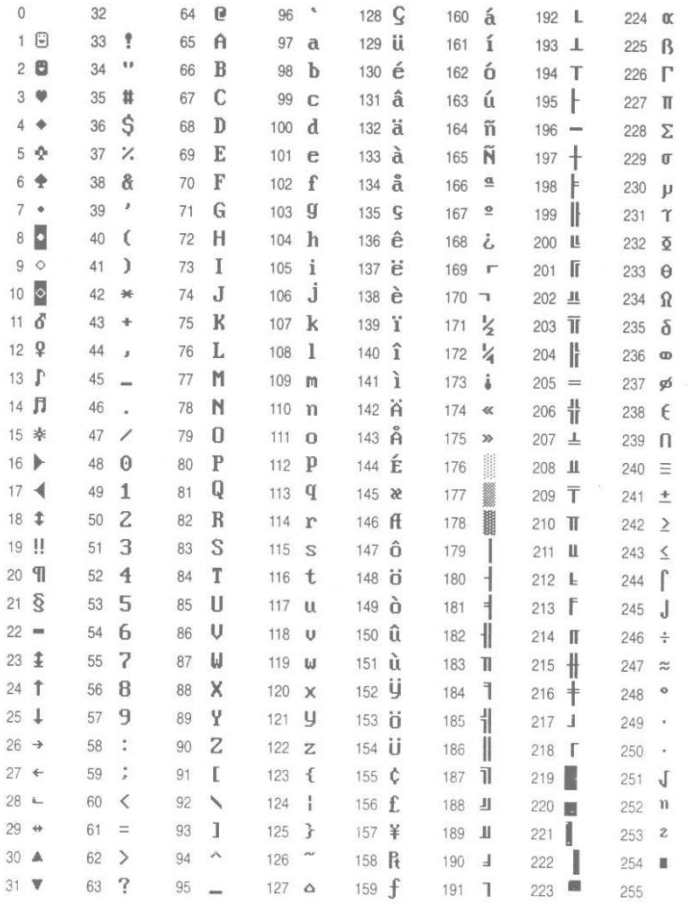

Figura 5.6. Tabla ASCII.

5.3 Encriptación del mensaje con AES

Este es el proceso para agregarle una capa de seguridad a la Marca de Agua, que es encriptar el mensaje y normalizar cada carácter encriptado en ceros y unos, como lo hicimos con los datos de audio. Para encriptar el mensaje ocupe una clase en PHP que hace el encriptado[28], esta clase necesita la llave (key) con la cual se va a encriptar el texto o mensaje y, obviamente, el texto o mensaje.

```
include("./AES.class.php");
$key = "FES-Aragon-UNAM.";
$txt = "I LOVE YOU";
$aes = new AES($key);
$txtCifrado=$aes->encrypt($txt);
```

[28] La clase AES se puede descargar desde esta página http://www.phpaes.com/.

Incluimos la clase AES.class.php, la instanciamos y le pasamos como parámetro la variable “$Key” que es nuestra llave, y en su método “encrypt” le pasamos la variable “$txt” que es nuestro mensaje que vamos a encriptar y a insertar como Marca de Agua. La clase AES nos regresa un texto cifrado.

Este texto contiene la frase “I LOVE YOU” cifrada con la llave “FES-Aragon-UNAM.”. Ahora tenemos que convertir estos caracteres en código binario, recordemos que cada carácter es un byte de información, así que tendremos 8 bits por cada carácter, esta conversión queda de la siguiente forma:

5.4 Generación de números pseudo-aleatorios

La generación de estos números es para tener una distribución pseudo-aleatoria de nuestro mensaje cifrado en los datos de audio. Para obtener estos números utilicé el algoritmo “Generador de congruencia lineal o LCG (Linear Congruential Generator)” que es un generador de números pseudo-aleatorios. La secuencia de números pseudo-aleatorios son producidos a partir de la siguiente relación de recurrencia:

$$x_{n-1} = (Ax_n + B) mod\ M$$

En esta expresión *A*, *B* y *M* tienen ciertos valores fijos, en tanto que x_0 es un número inicial o semilla. Este tipo de generador es computacionalmente rápido y de fácil implementación; un LCG produce una secuencia de valores que se repiten con un período que a lo mucho es M-1, y si los valores de A y B no son los apropiados, con frecuencia dicho período es mucho menor. Por lo anterior, nosotros tenemos que obtener números dentro del rango de nuestros datos de audio, por ejemplo, si nuestro arreglo es de 1500, tenemos que obtener números dentro de ese rango; así el periodo de nuestro LCG debe cumplir una secuencia

no repetida del tamaño de nuestro mensaje binario cifrado, es decir, si nuestro arreglo de datos de audio es de 1500 y nuestra cadena de bits de nuestro mensaje binario cifrado es de 128, el algoritmo debe ser capaz de obtener 128 números aleatorios en el rango de 1500.

La siguiente función que hice demuestra el funcionamiento para obtener los números aleatorios.

```
function algoritmoLCG($semilla,$modulo){
    $r=0;
    $semilla2=0;
    $semilla= $semilla*8;

    while($r<$semilla)
        {
            $semilla = ((13*$semilla)+25)%$modulo;
            if($semilla == $semilla2)
                {
                    echo "repetido ".$x;
                    break;
                }
            else
                {
                    $aleatorios[]=$semilla2 = $semilla;
                }
            $r++;
        }

    return $aleatorios;

}
```

A esta función le pasamos como parámetro los valores de la semilla y el modulo, utilizo como semilla el tamaño de la cadena de nuestro mensaje binario cifrado, para obtener por medio del ciclo while los números aleatorios que necesitamos, si algún número es repetido, no nos sirve, así que verificamos si algún número se repite por medio de la estructura if.

5.5 Inserción de la Marca de Agua

La inserción se realiza teniendo todos los datos normalizados, que son:

1. Los datos de audio del archivo wav en binario.
2. Los datos del mensaje cifrado en binario.
3. Los números aleatorios para la inserción.

Como se explica en el comienzo del capítulo, la inserción se llevará a cabo cambiando el bit menos significativo por el bit de la cadena cifrada, de acuerdo con el número de byte designado por el número aleatorio.

La siguiente función hace este cambio:

```
function algoritmoLSB($datos,$cifradoBin,$alea){

    for($n=0;$n<count($alea);$n++){

        $byte = $datos[$alea[$n]];
        $bit7 = substr($byte[0],0,7);
        $bit1 = substr($byte[0],-1);

        if($bit1==$cifradoBin[$n]){
            $aux= $bit7.$bit1;
            $datos[$alea[$n]]= $aux ;
        }else{
            $aux = $bit7.$cifradoBin[$n];
            $datos[$alea[$n]]= $aux ;
        }
        unset($byte);

    }

    return $datos;
}
```

La función recibe como parámetros los datos de audio del archivo, la cadena cifrada en binario y los números pseudo-aleatorios. Primeramente hace un ciclo definido por la cantidad de números aleatorios que obtuvimos, en este ejemplo obtuvimos de la cadena cifrada 128 bits, y por ende, 128 números aleatorios. Elegimos el byte de acuerdo con nuestros números pseudo-aleatorios y lo asignamos a la variable $byte = $datos[$alea[$n]]. Sustraemos los primeros 7 bits y lo guardamos en la variable $bit7 = substr($byte, 0, 7), esto es porque el que nos interesa es el último, de izquierda a derecha, seguidamente obtenemos el bit que nos interesa y lo asignamos a la variable $bit1 = substr($byte, -1). Verificamos si el

bit menos significativo de ese byte es igual o diferente al bit de nuestra cadena cifrada if($bit1 == $cifradoBin[$n]), si es igual, dejamos idéntico el byte $aux = $bit7.$bit1 y lo agregamos en el mismo índice al arreglo de datos de audio $datos[$alea[$n]] = $aux, ya que si tenemos de nuestra cadena cifrada un “1” y el bit menos significativo es igualmente un “1” representa ese bit de información de nuestra cadena cifrada, en cambio, si es diferente lo cambiamos por el bit de nuestra cadena cifrada $aux = $bit7.$cifradoBin[$n] y lo agregamos en el mismo índice al arreglo de datos de audio $datos[$alea[$n]] = $aux, si existe este caso, aquí es donde sucede el realmente el cambio del bit menos significativo.

5.5.1 Guardando los datos con la Marca de Agua en un archivo WAV

Por último lo que hay que hacer es convertir los nuevos datos de audio en ASCII, ya que los normalizamos para manipularlos, y guardar la Marca de agua como archivo WAV.

La siguiente función cambia una cadena binaria en su respectivo valor ASCII:

```
function bin2asc($binario){
     $out = '';
     for ($i = 0, $len = strlen($binario); $i < $len; $i += 8)
     {
        $out .= chr(bindec(substr($binario,$i,8)));
     }
     return $out;
}
```

Esta función toma como parámetro un valor binario, en este caso un byte que es como lo normalizamos. Tomamos ese byte substr($binario, $i, 8), lo cambiamos a su valor decimal bindec(substr($binario, $i, 8)), obtenemos su valor ASCII y se lo asignamos a la variable $out .= chr(bindec(substr($binario, $i, 8))).

El siguiente código se encarga de crear el archivo y guardarlo:

```
$create = fopen("aplicacion1-Watermark.wav","w");
if($create == false){
  die("No se ha podido crear el archivo.");
}
$cabeceras2= implode("",$cabeceras);
$datos = implode("",$data);
$write = $cabeceras2.$datos;
fwrite($create,$write);
fclose($create);
```

Abrimos un puntero a un archivo con fopen, le pasamos como parámetros el nombre del nuevo archivo y el modo de este archivo, en este caso “w” ya que queremos hacer un archivo de sólo escritura. Tomamos el array de las cabeceras, el cual contiene los primeros 44 bytes de nuestro archivo original, y lo convertimos en cadena $cabeceras2 = implode(“”, $cabeceras) ya que es como lo vamos a insertar en nuestro nuevo archivo, e igualmente al array de datos $datos = implode(“”, $data). Ya teniendo estas dos cadenas de datos, las unimos $write = $cabeceras2.$datos y las insertamos en el archivo fwrite($create, $write).

5.6 Lectura de la Marca de Agua

Para leer una marca de agua tenemos que seguir los pasos que se indican en la sección **2** de este capítulo. Obteniendo las cabeceras y los datos de audio del archivo, se tiene que generar los números aleatorios con los que insertamos la Marca de agua, esto se hace como se muestra en la sección **4** de este capítulo. Seguidamente extraemos los datos mediante los números aleatorios, y obtenemos los bytes que tienen la Marca de Agua mediante la siguiente función:

```
function extraeData($data,$aleatorios){
    for($j=0;$j<count($aleatorios);$j++){
    $f.=$data[$aleatorios[$j]];
    }
    return $f;
}
```

Esta función recibe los datos de audio del archivo y los números aleatorios como parámetro. Tomamos la cantidad de los números aleatorios, que es la cantidad de

bytes que vamos a obtener de los datos de audio, y hacemos un ciclo for($j=0; $j<count($aleatorios); $j++). Obtenemos los datos del array de los datos de audio en el índice de los números aleatorios $f .= $data[$aleatorios[$j]], con esto obtenemos los datos de la Marca de Agua.

Como sabemos que la función fgetc nos regresa caracteres ASCII, tenemos que normalizar los datos como lo hicimos en la sección **2** para poder manipularlos.

```
function asc2bin ($ascii){
    $cont=0;
    while ( strlen($ascii) > 0 )
    {
        $byte = "";
        $i = 0;
        $byte = substr($ascii, 0, 1);
        while ( $byte!= chr($i) ) { $i++; }
        $byte = base_convert($i, 10, 2);
        $byte = str_repeat("0", (8 - strlen($byte)) ) . $byte;
        $ascii = substr($ascii, 1);
        $binary[$cont]= $byte;
        $cont++;
    }
    return $binary;
}
```

Teniendo los datos normalizados, es decir, cada uno de los datos que extrajimos en binario, lo siguiente es obtener su bit menos significativo de cada dato, esto lo hacemos con la siguiente función:

```
function extraeLSB($binaryDataWatermark){
    $n=0;

    for($j=0;$j<count($binaryDataWatermark);$j++){
        $byte = $binaryDataWatermark[$j];
        $bitLSB = $byte[7];
        if($n==7){
           $returnLSB.=$bitLSB." ";
           $n=0;
        }
        else{
            $returnLSB.=$bitLSB;
            $n++;
        }

    }
    return $returnLSB;
}
```

Con el ciclo for recorremos cada uno de los datos que extrajimos, ya convertidos en binario for($j=0; $j<count($binaryDataWatermark); $j++). Extraemos el último bit que es el bit menos significativo $bitLSB = $byte[7] y hacemos una cadena en bloques de 8 bits, es decir, un bloque de 8 bits seguido de un espacio, como la siguiente cadena:

```
01011101 01111111 01010010 10001010...
```

Esto lo hacemos porque cada bloque de 8 bits, significa un carácter ASCII del mensaje cifrado con AES en su forma binaria, y nos facilita convertir la cadena en un arreglo para manipular cada carácter ASCII mediante la función explode()[29] de PHP:

```
$mensajeCifradoBinario = explode(" ",$LSB);
```

Teniendo el arreglo debemos que convertir cada elemento, que esta en bits, en su valor ASCII para obtener el mensaje cifrado que nos produjo la clase que use en la sección **3** mediante la siguiente función:

[29] Esta función convierte una cadena en un arreglo mediante un limitador, que en este caso es un espacio en blanco.

```
function mensajeBin2Asc($mensajeCifradoBinario){
    for($j=0;$j<count($mensajeCifradoBinario);$j++){
        $returnMsg.= bin2asc($mensajeCifradoBinario[$j]);

    }
    return $returnMsg;
}
```

Esta función realiza la conversión de 8 bits en su representación ASCII, con ayuda de la función “bin2asc” que utilice en la sección **5.1,** obtenemos la siguiente cadena:

]R OP V A

Como podemos observar, la cadena es la misma que obtuvimos al cifrar la frase “I LOVE YOU” con la llave “FES-Aragon-UNAM.” en la sección **3**. Si nosotros no hubiéramos encriptado esta frase, cualquier atacante de nuestra Marca de Agua, sencillamente hubiera obtenido el mensaje que tratamos de ocultar, pero en cambio, el atacante obtendría esta frase cifrada la cual sin la llave de encriptación sería imposible obtener dicho mensaje.

Por último, debemos desencriptar dicha cadena para obtener el mensaje oculto, la obtenemos utilizando la clase AES que ocupe en la sección **3**, con el siguiente código logramos esto:

```
$msgCifradoASCII = mensajeBin2Asc($mensajeAscii);

include("./AES.class.php");
$key = "FES-Aragon-UNAM.";
$aes = new AES($key);
$desencriptado =$aes->decrypt($msgCifradoASCII);
```

A nuestra clase de encriptación AES le pasamos a su método decrypt nuestra cadena que obtuvimos mediante la función mensajeBin2Asc y obtenemos el mensaje que insertamos como Marca de Agua.

I LOVE YOU

A continuación se muestran imágenes de esta aplicación que realicé.

En la Figura 5.7 vemos los datos que necesitamos para insertar la Marca de Agua, los cuales son:

- Nuestro mensaje que vamos a insertar.
- La llave con la cual vamos a encriptar.
- El archivo WAV que utilizaremos para insertar el mensaje como una Marca de Agua.

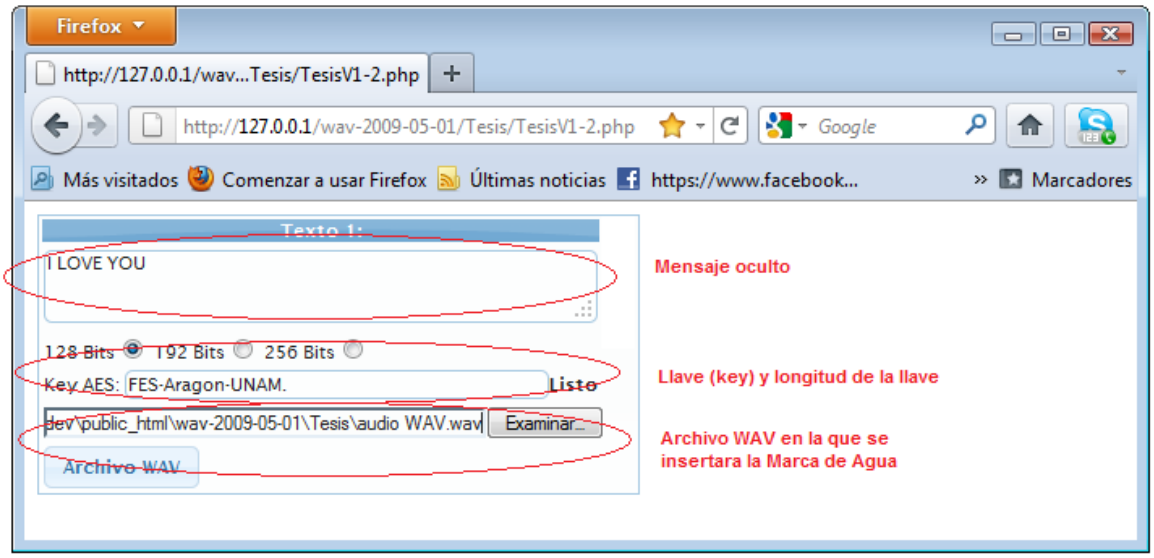

Figura 5.7. Inserción de datos.

La Figura 5.8 muestra nuestro mensaje encriptado, su representación binaria, la cual cada bit es el que vamos a cambiar por el bit menos significativo del arreglo de datos de audio dictados por los números aleatorios.

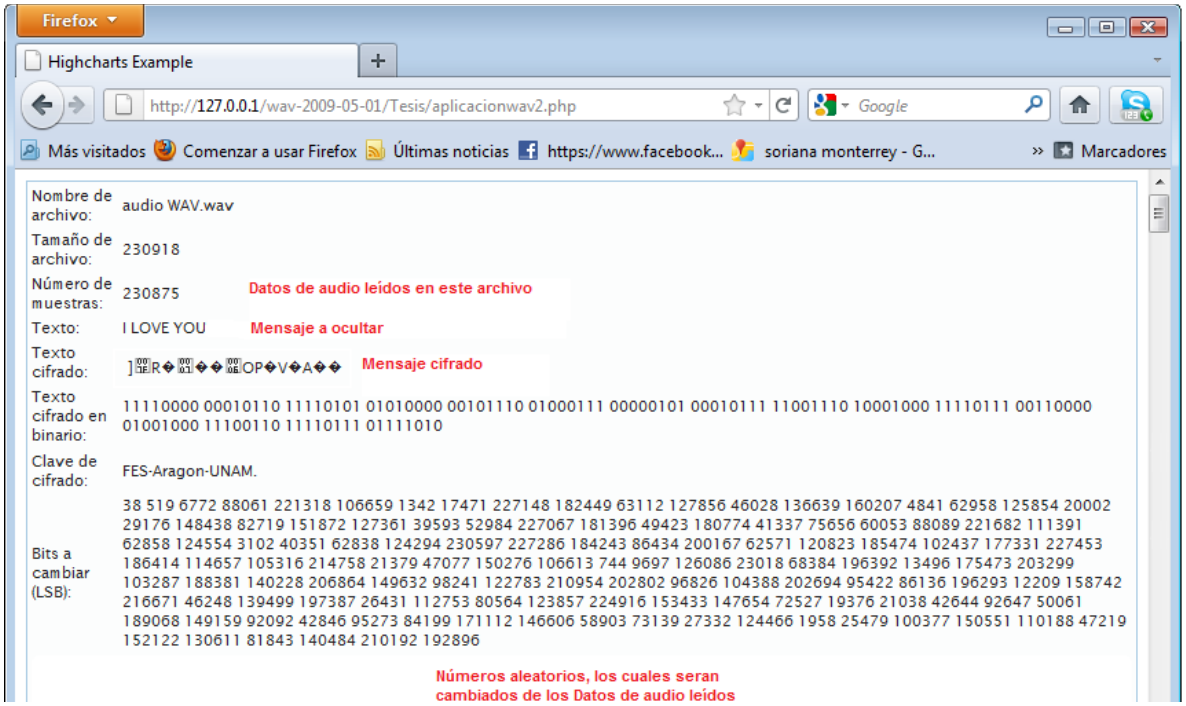

Figura 5.8. Información para la inserción de la marca de agua.

La Figura 5.9 nos muestra la onda de audio de la señal original y la señal con la Marca de Agua, lo hice así para observar los cambios que se realizaron al modificar un bit, en este caso el bit menos significativo.

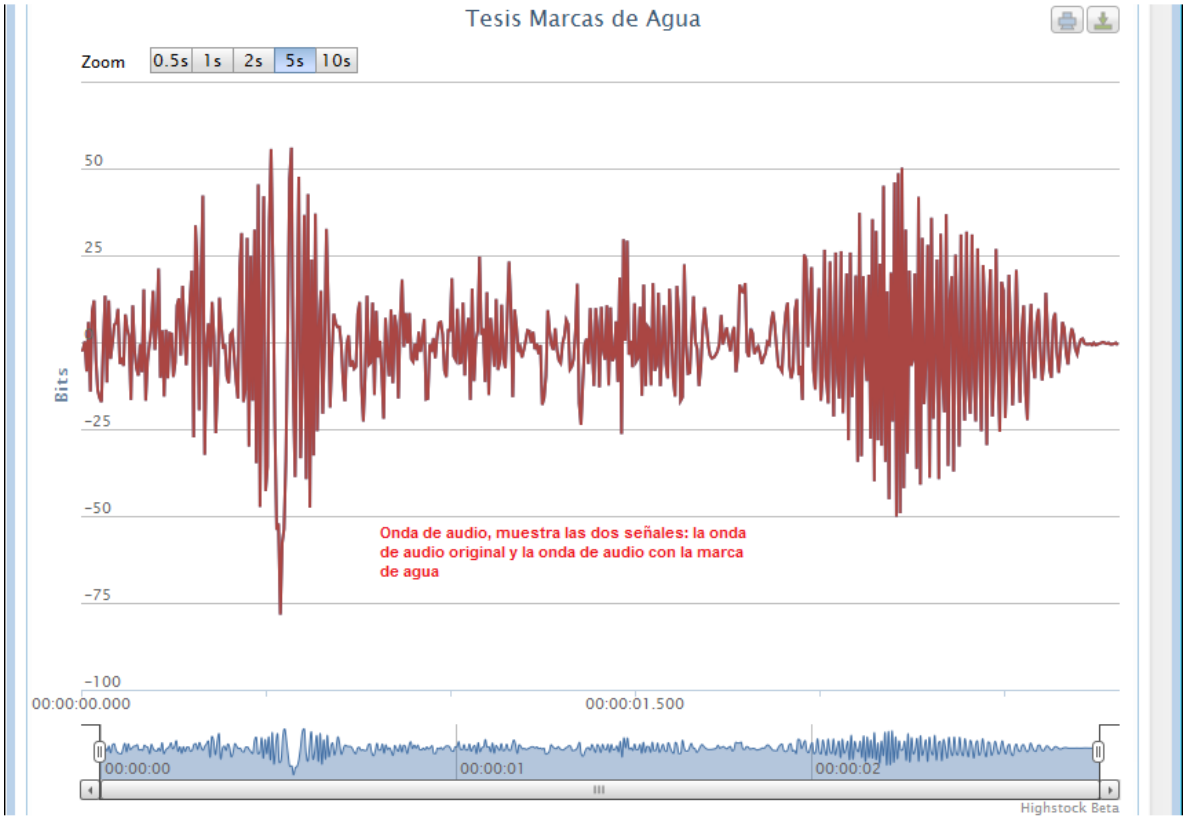

Figura 5.9. Onda de audio.

La Figura 5.10 nos muestra ese cambio, al hacer un zoom al la gráfica observamos que el byte 20002, el cual obtuvimos de los números aleatorios, cambió. La señal tenía como representación decimal -1, y al aplicar el cambio por el bit menos significativo, resulto ser cambiado por -2. Esto se debe, por ejemplo, si teníamos este byte: “01010101” y al aplicar el cambio resulto: “01010100”, al graficar dicho cambio, la gráfica tendrá que cambiar. Este cambio es insignificante o no significativo, ya que sólo se altera un bit, que en este caso el 85 por 84, es decir, un dígito. Si tomaríamos el más significativo resultaría el cambio muy significante, ya que si cambiaríamos el byte: “01010101” por este: “11010101”,

estaría cambiando el número 85 por 213, que en cuestión de datos de audio, es enorme.

Figura 5.10. Onda de audio original y con marca de agua.

CAPÍTULO 6

Un caso de uso de Marca de Agua

Las Marcas de agua es un buen método para incluir información dentro de piezas musicales, así que podemos utilizarlas de muchas formas como en el capítulo **3** muestra. Dos de las áreas de aplicación de una Marca de Agua me interesaron mucho, ya que con ellas conjuntamente se puede hacer un sistema de seguridad para las compañías de música y la protección de propiedad de los autores.

Estás áreas son la protección de propiedad y huellas dactilares (fingerprinting). La protección de propiedad indica que la marca de agua tendrá información esencial acerca del autor y de la obra, y será robusta para asegurar modificaciones. Las huellas dactilares en cambio se evocan en localizar el autor o los destinatarios mediante un id.

Planteamiento del problema

Durante los últimos 10 años, la piratería surgió como una amenaza directa al negocio de la música en México. Este problema masivo (el producto ilegalmente copiado es distribuido mayormente por vendedores informales) ha llegado a ser un problema muy serio debido a la falta de aplicación de leyes efectivas de derechos de propiedad intelectual por las autoridades Mexicanas. Para que el negocio prospere, las disqueras necesitan invertir en producciones locales de artistas Mexicanos. Las compañías disqueras necesitan tener la oportunidad de competir en un medio sano para poder recuperar su inversión y consecuentemente producir más música Mexicana.

Sin embargo, la piratería ha inhibido cualquier oportunidad. La situación de la piratería ha alcanzado su punto máximo. La producción a través de CD, DVD y la tecnología lo facilita mucho, no como en tiempos anteriores que los productos eran los discos de Vinil y los casetes, en los que se requerían instalaciones industriales mas sofisticadas.

El negocio de las disqueras es impredecible y de riesgo, ya que solamente 10% de las grabaciones recuperan sus inversiones. Cuando se enfrenta a niveles altos de piratería, la industria trata de minimizar el riesgo disminuyendo nuevas inversiones que requieren tiempo y dinero para salir al mercado.

Siete de cada diez CD vendidos en México son piratas. En total más de 122 millones de unidades de producto pirata fueron vendidos en el año 2006 contra aproximadamente 50 millones de unidades legítimas. México continúa siendo uno de los países con mayor índice en piratería de fonogramas.

Las pérdidas directas de la industria se encuentran en el rango de los $400 millones de dólares. Este impacto no considera el daño adicional que se le hace a negocios relacionados con la industria, como son los estudios de grabación, artistas, autores, arreglistas, artistas gráficos, impresores y músicos. Durante 2006 las ventas en unidades descendieron un 25.3%% respecto a 2005, y un 12.9% en valores.

Si no fuera poco, la fuga de material protegido por la misma compañía discográfica aparece en lugares asegurados por las autoridades por copia ilegal. Hay muchos casos donde en el lugar de la detención, se encuentran "masters[30]" originales de las compañías disqueras, lo que se supone que hay vínculos entre los ejecutivos o empleados de las compañías y los que se encargan de copiar y difundir dichas copias ilegales.

Propuesta de solución

Mi propuesta se enfoca hacia el proceso de creación de un CD, ya que ahí es donde se realizan los "masters" y las copias que son vendidas al los usuarios.

La producción de un CD de música consta de cuatro fases

1. Pre-Mastering

[30] Un máster de grabación es una grabación original de la que se hacen las copias editadas.

Aquí se reciben los datos que hay que incluir en el CD en diferentes soportes (pueden facilitarse un disco duro o un CD, y se llama master de grabación). Posteriormente los datos se comprueban bit a bit. Hay que asegurarse de que la estructura de los sectores corresponda con el tipo de CD ROM que se pretende fabricar. Una vez realizado lo anterior se puede pasar a la siguiente fase, no sin guardar antes una imagen de los datos en un disco duro para que sirva de referencia en las etapas subsecuentes como control de calidad.

Figura 6.1 Consola para grabar audio.

Figura 6.2. Dispositivos de almacenamiento.

2. Mastering

Comienza con la impresión de los símbolos numéricos en un disco de vidrio de 24 cm. de diámetro por medio de un rayo láser. El disco de vidrio está recubierto de una capa fotorresistente. Esta operación dura alrededor de 90 minutos.

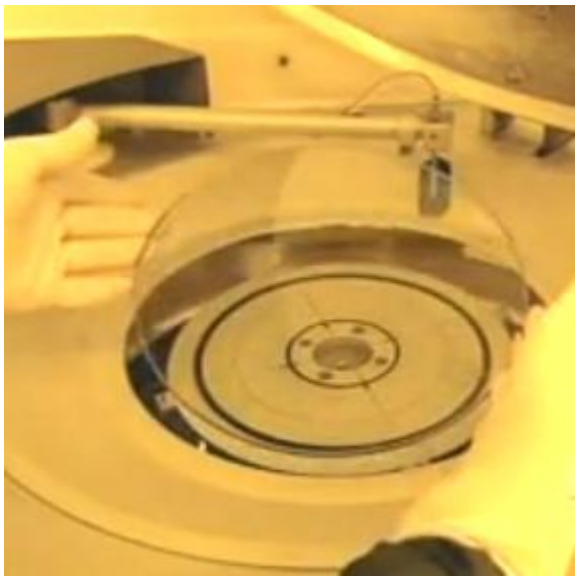

Figura 6.3. Grabado.

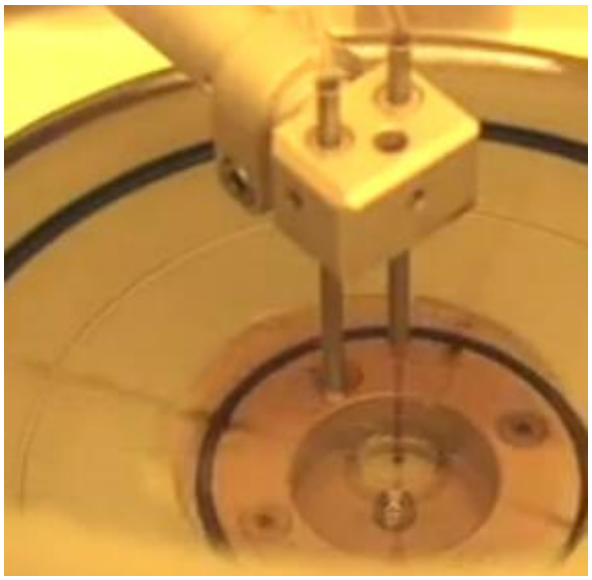

Figura 6.4. Cubierta fotorresistente.

Cuando el disco ya se encuentra grabado se procede a su metalización con una capa de níquel, de la que, a través de un tratamiento electrolito, se obtienen un total de tres planchas. La última de ellas es llamada matriz, que pasa por un control de comprobación de errores. Si en algún lado se encuentra alguna falla el proceso deberá iniciarse de nuevo.

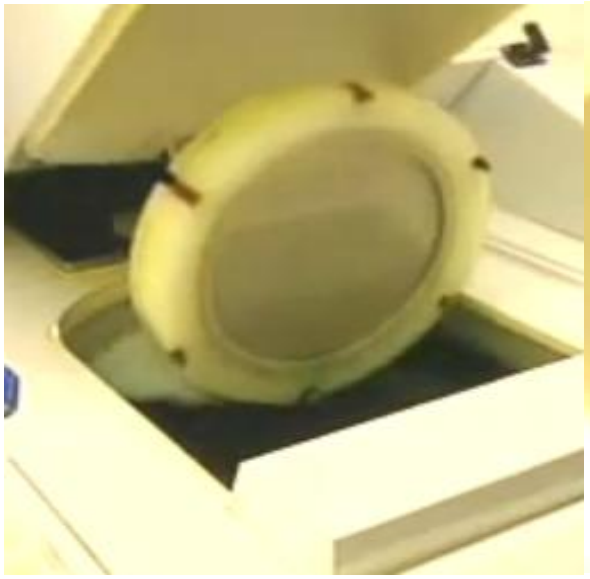

Figura 6.5. Metalización.

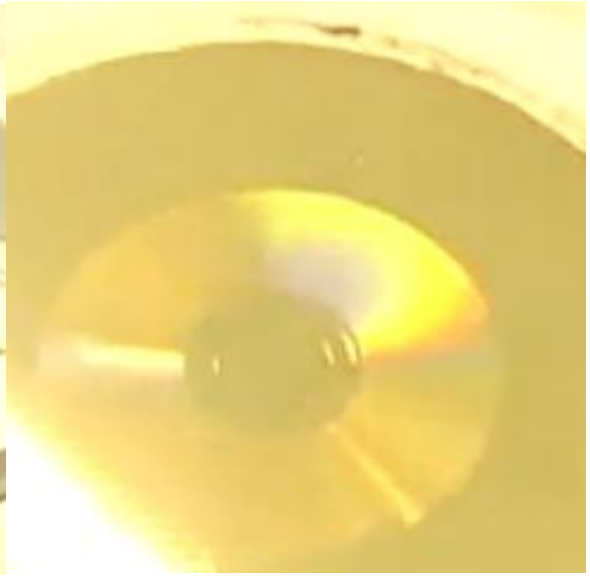

Figura 6.6. Fin de la metalización.

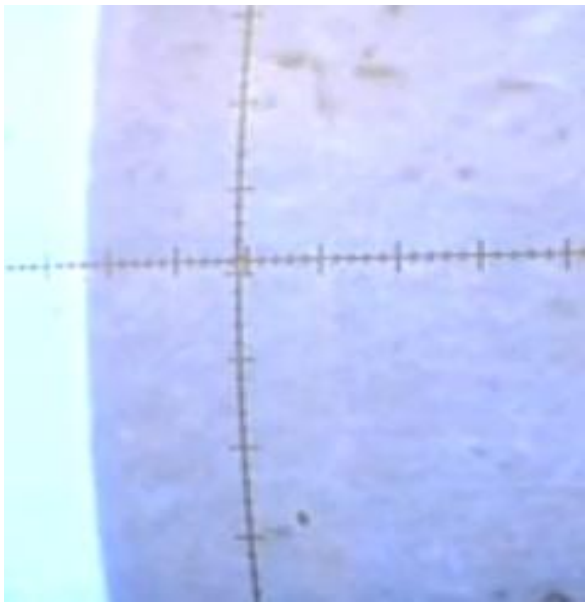

Figura 6.7. Comprobación de errores.

Figura 6.8. Se remueve capa protectora.

3. Prensado del CD

El policarbonato es la materia prima base para fabricar un CD. El policarbonato es un material plástico transparente y con cualidades ópticas muy definidas. Su forma original es granulada, pero se licúa a 310 °C; cuando ha sido licuado, se presiona sobre el molde que contiene la matriz. El plástico se enfría rápidamente, con lo que se forma una copia de la matriz. Esta copia no puede ser leída por una unidad de CD-ROM, ya que es totalmente transparente, y el rayo láser no se reflejaría en ella.

Figura 6.9. Prensado del policarbonato.

Figura 6.10. Acabado del prensado.

Para que la luz del láser pueda ser reflejada se procede a la metalización, que consiste en depositar una fina capa de aluminio sobre el disco. Este disco es muy sensible y es vulnerable a arañazos y podría oxidarse por lo que se le aplica una capa de laca, esta laca protege y permite imprimir sobre el CD las tintas serigráficas especiales (hasta cuatro colores), formando lo que se conoce como etiqueta.

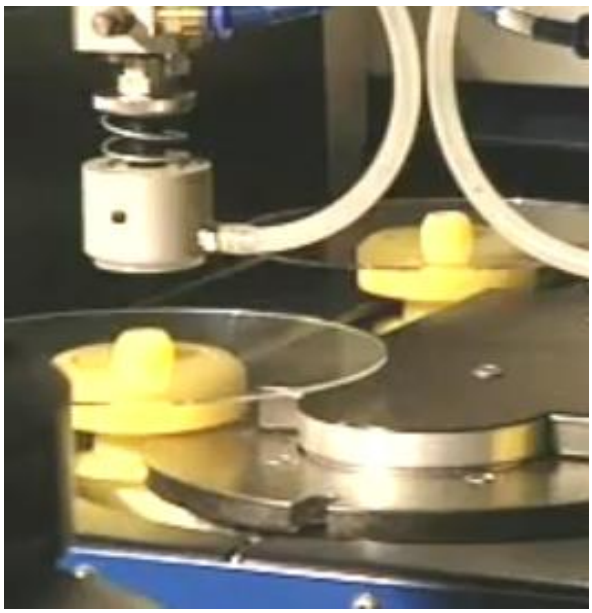

Figura 6.11. Metalización.

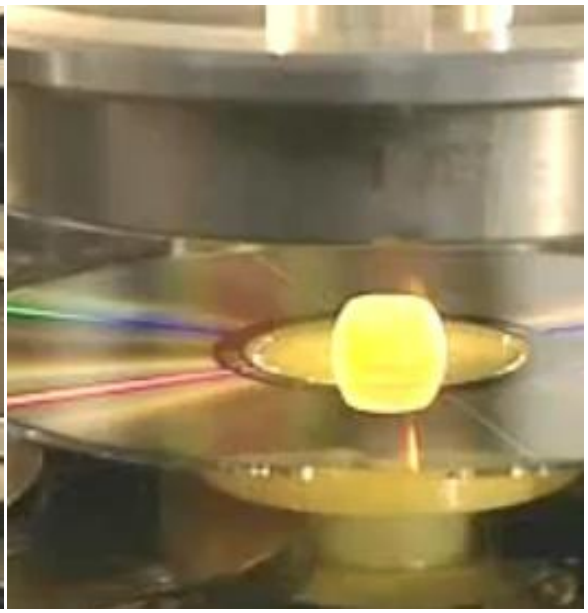

Figura 6.12. Acabado de la metalización.

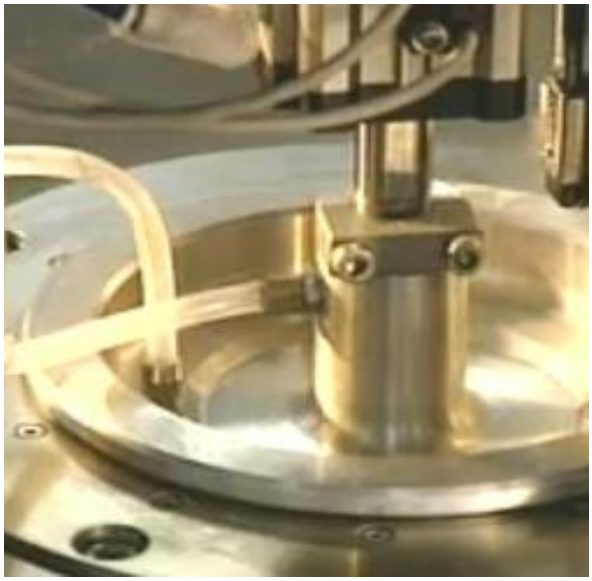

Figura 6.13. Preparación de laca protectora.

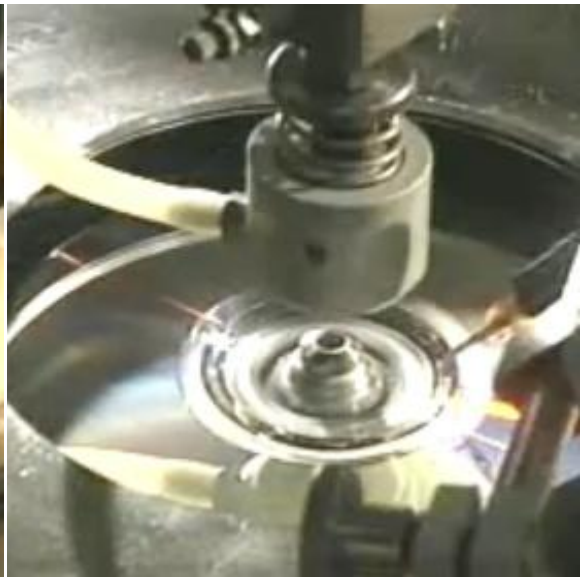

Figura 6.14. Aplicación de laca protectora.

Una vez salidos de la línea, los CD's son probados por un escáner que detecta las posibles impurezas del plástico o defectos de la capa de aluminio.

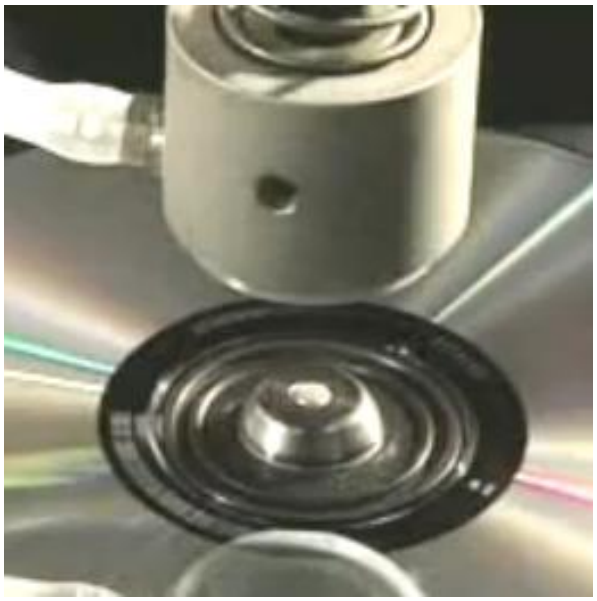

Figura 6.15. Escaneo de impurezas.

4. Acabado

El CD-ROM ya puede ser leído, pero le hace falta serigrafía para que tome una presentación atractiva para comercializarlo.

Figura 6.16. Aplicación de serigrafía.

Figura 6.17. Disco terminado

Figura 6.18. Producto final a la venta.

Teniendo en cuenta este proceso de creación de un CD de música, propongo lo siguientes:

1. Al terminar la grabación de un artista, se crea el master de grabación, a este master se le inserta una Marca de Agua, la cual contendrá la siguiente información:
    a. Información el artista
    b. Fecha de grabación
    c. El lugar donde se realizó la grabación
    d. La compañía disquera
    e. El lugar donde se realizará el proceso de copiado
    f. El nombre a quien va dirigido el master
    g. Id de identificación

    Con esta información nos aseguramos de tener un master de grabación protegido, ya que si se encuentran copias ilegales en el mercado, y al

extraerle la Marca de Agua tiene esta información, se aplicarían políticas dentro de la empresa para saber donde fue la fuga del master.

2. Al llegar el master de grabación a la compañía donde se realizará el copiado, se tendrá que extraerle la marca de agua y los datos originales de audio. A los datos de audio se le insertará nuevamente una Marca de Agua antes de ser impresos en el disco de vidrio, con la siguiente información:
    a. Información del artista
    b. Fecha de grabación
    c. Fecha de copiado
    d. Lugar donde se realizó la grabación
    e. Compañía disquera
    f. Lugar donde se realiza el proceso de copiado
    g. Lugar donde se distribuirán las copias
    h. Los números de serie de los dispositivos que intervinieron en el proceso de copiado
    i. Id de identificación

    Al insertar esta información nos aseguramos de tener un control de creación y procedencia de la copia. Un ejemplo tendría lugar cuando la compañía disquera distribuyera las copias a los centros de venta. Se haría una muestra de las copias ilegales en venta y obtendríamos porcentajes de las zonas donde provienen estas.

    Cómo un ejemplo más práctico muestro el Distrito Federal. La compañía que produce los discos esta en Coyoacán y de ahí distribuye a los centros de ventas, cada tiraje de discos a un centro de venta estará identificado dentro de la Marca de Agua.

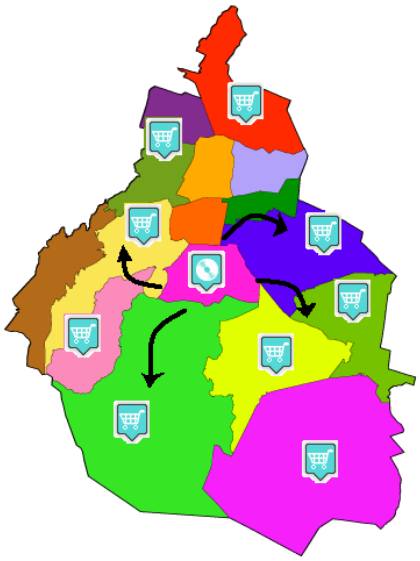

Figura 6.19. Distribución de la compañía discográfica.

Al identificar copias ilegales, se hace un muestreo de las copias vendidas en el Distrito Federal y se obtiene de qué punto de venta son procedentes y se obtiene valores porcentuales.

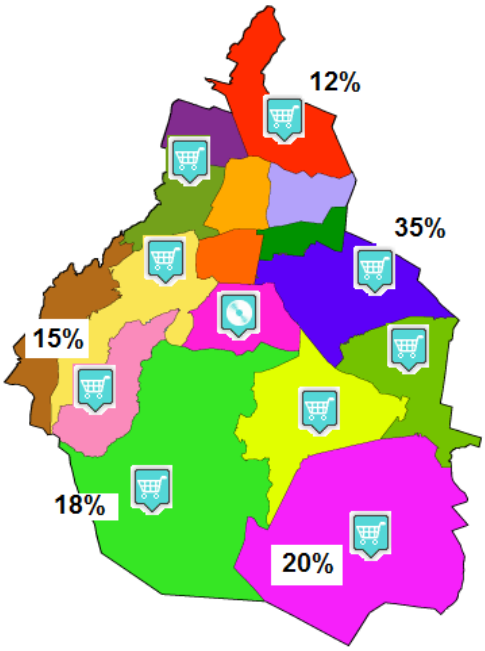

Figura 6.20. Porcentaje de copias ilegales por delegación.

Ya obteniendo el porcentaje de las muestras dependiendo el punto de venta, ahora hay que hacer andar las políticas de las compañías disqueras conjuntamente con las autoridades del estado, para atacar el problema de las copias ilegales. No propongo acabar con las copias ilegales, ya que es un problema que abarca varios aspectos en este rubro, pero si las políticas, aunada a la Marca de Agua y a las autoridades, serían un buen conjunto para combatir este mal que aqueja a las compañías disqueras. Buenas políticas, disminuyen el riesgo o vulnerabilidad de lo que se quiere proteger, pero sin la tecnología se ven atenuadas o tal vez inexistentes.

CONCLUSIONES

Las raíces históricas de la marca de agua digital se derivan principalmente de la esteganografía, el arte de la ocultación de datos. A pesar que la Marca de Agua y la esteganografía son en cierto sentido similar, la diferencia principal radica en la idea de la solidez de las Marcas de Agua digitales.

El rápido crecimiento de tecnología multimedia facilita la producción y transmisión de datos digitales. Esto ofrece no sólo oportunidades, sino también retos en la protección de derechos de autor, prueba de propiedad, autenticación y detección de manipulación, huellas dactilares (fingerprinting), control de emisión, control de copia, control de acceso y portador de información. Y esto conlleva a la investigación de métodos de seguridad para ser implementados en nuestra vida cotidiana.

La aplicación que se muestra en esta tesis muestra el comportamiento de la inserción de una Marca de Agua, el resultado de este aplicación mostró que el sonido no se altero con los bits que se cambiaron con el método LSB, pero si mostró la debilidad que tiene a la manipulación indebida, ya que, se tiene que diseñar muy bien para un problema específico, en este caso sirvió bien para una demostración.

El futuro de las Marcas de Agua sigue en aumento y en constante investigación, las grandes compañías de software las han tomado para su beneficio y esto hace que la investigación no sólo provenga de las compañías en sí, sino que las universidades tengan una participación importante y que hagan de estas marcas una posible inversión.

Las Marcas de Agua no sólo protegen a la música, sino muchísimos formatos digitales, mientras que los datos estén en binario y haya un problema de resguardarlos, éstas estarán presentes para dar un gran apoyo y, en varios casos, solucionarlos.

BIBLIOGRAFÍA

- M. Barni, F. Bartolini, R. Caldelli, A. Piva, Geometric-Invariant Robust Watermarking through Constellation Matching in the Frequency Domain, Proceedings of 7th IEEE International Conference on Image Processing ICIP 2000, Vancouver, Canada, September 10-13, 2000, Vol. II, pp. 65 -68.

- S. Voloshynovskiy, S Pereira, T. Pun, JJ. Eggers and J. K. Su. Attacks on Digital Watermarks: Classification, Estimation-based attacks and Benchmarks submitted to IEEE Communication Magazin, 2001.

- S. Voloshynovskiy, S Pereira, V. Iquise, T. Pun. Towards a second generation watermarking benchmark Signal Processing, Special Issue on Information Theoretic Issues in Digital Watermarking, 2001.

- Yu H, Kundur D & Lin C (2001) Spies, thieves, and lies: The battle for multimedia in the digital era. IEEE Multimedia 8(3): p 8–12.

- Wu M & Liu B (2003) Multimedia Data Hiding. Springer Verlag, New York, NY. Kirovski D, Malvar H & Yacobi Y (2002) Multimedia content screening using a dual watermarking and fingerprinting system. In: Proc. ACM Multimedia, Juan Les Pins, France, p 372–381.

- Cox I, Miller M & McKellips A (1999) Watermarking as communications with side information. Proceedings of the IEEE 87(7): p 1127 –1141.