Esquema de Huella Digital para Datos Relacionales con Baja Distorsión y Resistente a Ataques de Colusión

por

Daniel Cruz Paz

Tesis sometida como requisito parcial
para obtener el grado de

MAESTRÍA EN CIENCIAS EN EL ÁREA DE CIENCIAS COMPUTACIONALES

en el

Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE)

Diciembre, 2018

Santa María de Tonantzintla, Puebla, CP 72840

Asesora:

Dra. Claudia Feregrino Uribe, INAOE, México
Coordinación de Ciencias Computacionales INAOE

©INAOE 2018
Todos los derechos reservados

El autor otorga al INAOE el permiso de reproducir y distribuir copias de esta tesis en su totalidad o en partes mencionando la fuente.

Agradecimientos

Agradezco a mi asesora, la Dra. Claudia Feregrino Uribe por confiar en que yo podría realizar este trabajo y guiarme durante todo el proceso hasta alcanzar los objetivos planteados.

Agradezco a la Dra. Lil María Xibai Rodríguez Henríquez, al Dr. José Francisco Martínez Trinidad y al Dr. Luis Villaseñor Pineda por sus observaciones para que este trabajo de investigación fuera concluido.

Agradezco enormemente a mi familia por el gran apoyo que me brindaron en cada momento de esta etapa.

A mis amigos Jessica, Gabriel, Bruno y Javier por su tiempo, comprension y compañia; ha sido un honor compartir esta amistad con ustedes.

Al INAOE por haberme otorgado la oportunidad de realizar mis estudios de posgrado en esta reconocida institución.

Por último, agradezco al Consejo Nacional de Ciencia y Tecnología (CONACYT), por el apoyo otorgado durante dos años para la realización de esta investigación, por medio del beca No.450810.

Resumen

El uso de Bases de Datos Relacionales, como medio digital para el almacenamiento y análisis de datos de organizaciones, ha aumentado significativamente a lo largo de los años, dando paso al crecimiento de amenazas que pretenden vulnerar los datos con fines ilícitos.

Sin embargo, a pesar de contar hoy en día con herramientas que permiten la protección de este medio digital ante estas amenazas, la distribución ilegal de copias es un constante problema, debido a la facilidad de compartir información y el valor que tienen los datos actualmente. Por lo tanto, es esencial desarrollar mecanismos robustos que permitan la protección de propiedad de estos activos digitales.

En esta tesis se propone un esquema de huella digital resistente a ataques de colusión para bases de datos relacionales con el objetivo de proveer un mecanismo de seguridad robusto para la identificación del usuario culpable ante casos de copia ilegal y distribución de contenido no autorizado.

Este trabajo presenta un enfoque en el algoritmo de inserción de la huella digital orientado al incremento de la capacidad de inserción de bits, así como la reducción de la distorsión generada en la base de datos relacional, mejorando la recuperación de la huella digital y la identificación del usuario culpable, ante ataques a la base de datos.

A través de los resultados derivados de los experimentos realizados, se observa una mejora en cuanto a la distorsión generada por la inserción de la huella digital respecto a trabajos del estado del arte, además del incremento en la persistencia de la huella digital

ante ataques de inserción, alteración y eliminación de tuplas.

Índice general

Resumen **v**

Índice de figuras **xi**

Índice de tablas **xiii**

1. Introducción **1**

1.1. Importancia de las marcas de agua y huellas digitales 1

1.2. Seguridad en las bases de datos relacionales 3

1.3. Problemática 4

1.4. Motivación 5

1.5. Objetivos 6

1.5.1. Objetivo general 6

1.5.2. Objetivos específicos 6

1.6. Metodología 7

1.7. Estructura del documento 9

2. Marco teórico **11**

2.1. Introducción a marcas de agua digitales 11

2.1.1. Clasificación de las marcas de agua digitales 13

2.2. Huella digital 14

2.2.1. Generación de la huella digital 16

2.2.2. Rastreo del traidor 18

2.3. Bases de datos relacionales 19

2.4. Huella digital para bases de datos relacionales 20

2.4.1. Requerimientos 21

2.4.2. Ataques 22

2.4.3. Métricas de evaluación 25

2.4.4. Funciones criptográficas 28

3. Estado del arte **29**

3.1. Marca de agua digital en bases de datos relacionales 29

3.1.1. Técnicas básicas 30

3.2. Huella Digital en bases de datos relacionales 32

3.3. Discusión 37

4. Esquema de huella digital propuesto **41**

4.1. Parámetros iniciales 41

4.2. Esquema de huella digital 44

4.2.1. Construcción de la huella digital 45

4.2.2. Algoritmo de inserción 48

4.2.3. Algoritmo de detección y recuperación de huella digital 54

4.3. Rastreo del traidor 59

4.4. Discusión 61

5. Experimentos y resultados **63**

5.1. Entorno de desarrollo del esquema 63

5.2. Evaluación de usabilidad 65

5.3. Evaluación de la capacidad 71

5.4. Evaluación de robustez 74

5.4.1. Inserción de registros 75

5.4.2. Alteración de registros 77

5.4.3. Eliminación de registros 80

5.4.4. Ataque de colusión 82

5.5. Discusión de los resultados 83

6. Conclusiones y trabajo futuro **85**

6.1. Conclusiones 85

6.2. Trabajo futuro 87

A. Apéndice: Comparación de la distorsión generada en la RDB **89**

Índice de figuras

1.1. Esquema general de la metodología propuesta. 7

2.1. Esquema general de huella digital. 15

2.2. Acusación y generación de código de huella digital de G.Tardos. 17

2.3. Ejemplo de una relación de la RDB. 20

4.1. Diagrama general del esquema propuesto. 44

4.2. Diagrama del algoritmo de inserción. 48

4.3. Inserción de bits utilizando distribución proporcional. 53

5.1. Comparación de la distorsión en términos de MSE. 67

5.2. Efecto de reducción de registros a marcar. 68

5.3. Capacidad de inserción con limite de distorsión. 69

5.4. Capacidad de inserción del esquema propuesto. 72

5.5. Comportamiento con una base de datos de menor tamaño. 73

5.6. Comportamiento con una base de datos de mayor tamaño. 74

5.7. Ataque de inserción de tuplas con variación de la fracción de tupla. . . . 76

5.8. Ataque de alteración de tuplas con variación de la fracción de tupla. . . . 79

5.9. Ataque de eliminación de tuplas con variación de la fracción de tupla. . . 81

Índice de Tablas

3.1. Esquemas de huella digital del estado del arte. 39

4.1. Parámetros principales del esquema propuesto. 42

4.2. Error porcentual y modificación de los LSB. 43

5.1. Reducción de distorsión y tiempo de inserción. 68

5.2. Comparación del error relativo en la base de datos Forest Cover Type. . . 70

1

Introducción

En este capítulo se presenta una introducción preliminar a las marcas de agua digitales y los esquemas de huella digital, su importancia y características, así como el impacto de su aplicación a bases de datos relacionales. Posteriormente, se realiza una descripción de los objetivos propuestos y la metodología desarrollada a lo largo de este trabajo. Por último, se hace una breve descripción del contenido del documento.

1.1. Importancia de las marcas de agua y huellas digitales

Las marcas de agua digitales es una técnica de ocultamiento de información que permite la transmisión de información insertada dentro de un medio digital, la cual está directamente relacionada con este medio digital, siendo principalmente aplicada a la protección de propiedad de contenido digital.

Su importancia surge a partir del incremento en el uso de medios digitales tales como imágenes, audio y vídeo, debido a la penetración que ha tenido internet a través de la utilización y diversificación de dispositivos electrónicos. En los cuales la seguridad nunca es garantizada completamente, debido a que su diseño no contempla la gran diversidad de ataques maliciosos que se tienen hoy en día, además de que cualquier error

en su diseño puede ser explotado por un atacante Schneier (2011). Por lo cual, cualquier tipo de amenaza como robo, falsificación o distribución no autorizada de la información contenida en estos dispositivos, puede presentarse en cualquier momento. Por lo tanto, es indispensable minimizar los daños ocasionados ante estos tipos de amenazas.

Una estrategia para minimizar estos daños es diseñar mecanismos de seguridad que permitan proteger los activos digitales, como lo son: algoritmos de cifrado, estrategias de control de acceso y marcas de agua digitales, los cuales pueden complementar sus características para mejorar el nivel de seguridad de un sistema específico. En particular, las marcas de agua digitales han sido una de las técnicas más estudiadas principalmente para la protección de propiedad, ya que permite la identificación del propietario de un medio digital en caso de disputa de derechos de autor (Halder et al., 2010). Lo cual sucede posterior al robo o manipulación del contenido digital, es decir, si bien las marcas de agua digitales no evitan que se efectúen las amenazas antes mencionadas, proporcionan una herramienta para efectuar el reclamo de propiedad del medio digital y evitar su distribución ilegal.

Por otra parte, como extensión de las marcas de agua digitales, el uso de huellas digitales forma parte de un conjunto de técnicas aplicables en la distribución de copias de un medio digital a diferentes usuarios, con el objetivo de mantener el control sobre cada una de las copias distribuidas. Por lo tanto, en este caso, la distribución no autorizada de alguna de las copias es la principal amenaza que tratan de solucionar estas técnicas, ya que de otra forma cualquiera de los usuarios podría distribuir ilegalmente el contenido digital evitando la culpabilidad de esta acción.

Esencialmente, un esquema de huella digital se refiere al conjunto de algoritmos que permiten agregar y extraer una huella digital a un objeto, con el objetivo de distinguir a este de otros objetos similares y proveer protección de propiedad ante casos de distribución ilegal. Por lo cual, dada la importancia de estas técnicas para la protección de propiedad de contenido digital, este trabajo se enfoca en la seguridad de bases de datos

relacionales a través del desarrollo de un esquema de huella digital, lo cual ha representado nuevos retos debido a que la estructura y características de estos datos difieren de los datos multimedia inicialmente usados en el área, y por lo tanto los algoritmos de inserción y detección de huella digital necesitan de nuevos requerimientos para su funcionamiento en este tipo de medio digital Halder et al. (2010).

En este sentido, los algoritmos propuestos en esquemas de huella digital, hasta ahora buscan insertar información dentro de las relaciones de una base de datos relacional (RDB, por sus siglas en inglés), introduciendo pequeños errores en los valores de tuplas a través de la modificación de los bits que componen cada valor (Xie et al., 2016). Además de contar con algoritmos de detección y recuperación de huella digital que permiten identificar al usuario traidor de la distribución de copias no autorizadas por el propietario de la RDB.

Sin embargo, las copias autorizadas e identificadas por medio de la huella digital están expuestas a ataques con el fin de degradar la información de la huella digital. Uno de los principales ataques en esquemas de huella digital es el ataque de colusión que consiste en la combinación de los valores de la huella digital tomados de diferentes copias de la misma relación, para crear una copia que evite la extracción o identificación de la huella digital de un usuario. Debido a éstas, entre otras implicaciones del desarrollo de un esquema de huella digital, es importante proponer mejoras con lo cual se garantice la protección de las bases de datos relacionales.

1.2. Seguridad en las bases de datos relacionales

A lo largo de los años, las bases de datos han estado presente como parte esencial de la informática, en particular las Bases de Datos Relacionales han sido ampliamente usadas para el almacenamiento de datos en diversos sectores tales como la industria, organizaciones gubernamentales e instituciones académicas Elmasri and Navathe (2010). Esto debido a que su modelo relacional facilita la interpretación y el control de los datos

requeridos en diversos procesos dentro de estos organismos.

Por otra parte, los datos hoy en día son una herramienta con un gran potencial, ya que su análisis permite tomar decisiones y generar conocimiento con el objetivo de causar impacto en cualquier ámbito de la sociedad. Además, el análisis de datos ha aportado mayor significado e interés en las bases de datos, debido a que su explotación bajo fines ilícitos puede ser de gran beneficio para quien los posea, un ejemplo de ello son las bases de datos relacionadas a datos personales, los cuales pueden ser comercializados de forma ilegal para cumplir con objetivos diferentes a los que fueron recopilados originalmente.

Por lo tanto, esto ha significado un gran reto para las organizaciones que resguardan información en sus bases de datos, ya que pese a contar hoy en día con mecanismos de seguridad para la protección de estos activos digitales, es importante mejorar y proponer nuevas soluciones que incrementen los niveles de seguridad desde distintos enfoques, con el objetivo de mantener la privacidad e integridad de la información.

Una de las alternativas que se menciona en este trabajo son las huellas digitales aplicadas a bases de datos relacionales para la protección de propiedad, el cual es otro de los aspectos a considerar dentro de la seguridad de un medio digital, con el objetivo de evitar su distribución ilegal.

1.3. Problemática

Los esquemas de huellas digitales se basan en la modificación de los valores de las relaciones de la RDB para la inserción de la información que identifique la propiedad de este activo digital. Sin embargo, la usabilidad de la RDB recae en el control del grado de distorsión generado en los datos, lo cual restringe la modificación de estos valores, ya que la integridad de la información contenida en la base de datos es importante para las organizaciones. Por lo cual, es importante diseñar esquemas que reflejen una baja distorsión, pero que a su vez aporten mayor robustez a la información que se inserta,

debido a que ataques a la RDB pueden degradar esta información.

Por otro lado, en esquemas de huellas digitales, la inserción de la información en distintas copias de la RDB provoca que exista una inherente exposición a ataques de colusión, tal ataque consiste en la coalición de usuarios con diferentes copias de la RDB marcadas con su huella digital correspondiente, con el objetivo crear una copia útil resultado de las diferencias entre dichas copias, la cual no pueda ser identificada durante el proceso de detección de la huella digital(Li et al., 2005). Dado este fenómeno, la inserción de la huella digital requiere de estrategias que minimicen el impacto de este ataque, lo cual a su vez restringe la búsqueda por reducir la distorsión generada por la inserción de la huella digital.

1.4. Motivación

Los esquemas de huella digital para RDB propuestos hasta el momento, ofrecen soluciones para la distribución no autorizada y la identificación del usuario culpable, contemplando tres de los requerimientos principales de las marcas de agua digitales, los cuales son: Usabilidad, Capacidad y Robustez, los cuales están relacionados con la distorsión generada en la RDB posterior a la realización de cambios en los datos, la cantidad de datos a insertar y la capacidad de soportar ataques que puedan remover o modificar la huella digital (Halder et al., 2010).

Sin embargo, la alteración de uno de estos requerimientos afecta el comportamiento de los otros dos, es decir, a mayor robustez, se requiere de mayor capacidad, pero se compromete la usabilidad de los datos ya que la distorsión aumenta. Por otro lado, a mayor usabilidad, menor capacidad y menor robustez a ataques. Por lo cual, el encontrar un balance en el que ninguno de estos requerimientos se vea comprometido es de suma importancia.

Por lo tanto, es necesario mejorar el proceso de inserción de huella digital para

minimizar la distorsión en los datos comparado con técnicas previas, y al mismo tiempo obtener mayor robustez para resistir ataques que pretendan degradar la huella digital como son la inserción, eliminación y actualización de registros, además de contemplar el problema de ataque de colusión ya que presenta un mayor reto de persistencia de la huella digital.

1.5. Objetivos

1.5.1. Objetivo general

Diseñar e implementar un esquema de huella digital resistente a ataques de colusión para la protección de propiedad de bases de datos relacionales vulnerables a copia ilícita y distribución de contenido no autorizado, con mayor robustez y menor distorsión que trabajos del estado del arte.

1.5.2. Objetivos específicos

- Analizar los esquemas de huella digital más adecuados que hayan aportado mejores resultados en cuanto a capacidad y robustez, los cuales se tomarán como referencia para el desarrollo del esquema propuesto.
- Diseñar e implementar un algoritmo de inserción de huella digital para bases de datos relacionales resistente a ataques de colusión, con baja distorsión y mayor robustez a ataques de inserción, alteración y eliminación de tuplas.
- Diseñar e implementar un algoritmo de detección y extracción de huella digital para la identificación del usuario culpable ante la distribución de contenido no autorizado.
- Evaluar los niveles de distorsión y robustez generados por la inserción de la huella digital para valorar el rendimiento del esquema respecto a trabajos del estado del

arte.

1.6. Metodología

En esta sección se describen las etapas consideradas para el desarrollo del tema de investigación, de acuerdo a los objetivos planteados.

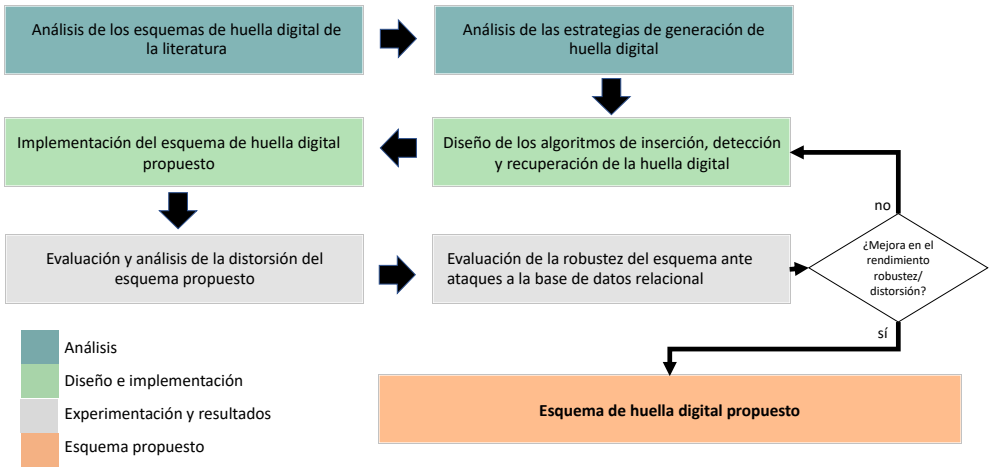


Figura 1.1: Esquema general de la metodología propuesta.

A) **Análisis de los esquemas de huella digital de la literatura**

En esta etapa se realiza un análisis de las estrategias empleadas en esquemas de huella digital de la literatura, tanto en la inserción como en la detección de la huella digital, con la finalidad de proponer una mejora al problema de identificación del usuario culpable posterior a la copia ilícita o redistribución de contenido no autorizado, además de analizar los ataques comúnmente abordados en dichos trabajos y los resultados obtenidos.

B) **Análisis de las estrategias de generación de huellas digitales**

Uno de los principales componentes de los esquemas de huella digital es la información que va a ser insertada dentro del medio digital para la identificación de

usuarios, por lo cual en esta etapa se analizan los métodos más adecuados para la generación de la huella digital, contemplando características tales como: formato, longitud y seguridad.

C) **Diseño de los algoritmos de inserción y detección de la huella digital**

En la siguiente fase, el diseño de los algoritmos de inserción y detección de huella digital son realizados basados en las mejoras encontradas en los esquemas analizados de la literatura. Además, la huella digital generada bajo el método seleccionado durante el paso anterior es incorporada al diseño propuesto.

D) **Implementación del esquema de huella digital propuesto**

Posterior al diseño, la implementación de los algoritmos de inserción y detección de la huella digital propuestos es realizada utilizando las herramientas de programación seleccionadas previamente, además del gestor de bases de datos y la base de datos seleccionada con base a lo reportado en esquemas de marcas de agua y huella digital para bases de datos relacionales del estado del arte.

E) **Evaluación y análisis de la distorsión del esquema propuesto**

Una vez realizada la implementación, se evalúa las alteraciones realizadas a la base de datos posterior a la inserción de la huella digital, debido a que las modificaciones a los valores de la base de datos afectan directamente la usabilidad de la base de datos y sus estadísticas. Por otro lado, el número de bits insertados repercute en los niveles de robustez del esquema, lo cual se evalúa en la siguiente etapa.

F) **Evaluación de la robustez del esquema propuesto ante ataques a la RDB**

Finalmente como parte de la última etapa, la copia de la base de datos que ha sido marcada con la huella digital es expuesta a modificaciones realizadas como parte de las operaciones comunes que se ejecutan en una base de datos. Estos ataques son evaluados con diferentes grados de severidad para analizar la taza de persistencia

de la huella digital y la identificación del usuario correspondiente a la copia de la RDB analizada.

1.7. Estructura del documento

Este documento tiene el siguiente orden. El Capítulo 2, describe los conceptos necesarios para comprender la propuesta de solución que se desarrolló, además de las características y restricciones que se presentan en los esquemas de huella digital. En el Capítulo 3, se describen principalmente los trabajos del estado del arte referente a esquemas de huella digital para bases de datos relacionales que se tiene hasta el día de hoy, además de discutir las ventajas y desventajas que presentan. En el Capítulo 4, se describe la estrategia de selección de tuplas y atributos para la inserción de la huella digital que se realizó, además del esquema de huella digital propuesto en su totalidad. En el Capítulo 5, se presentan los resultados obtenidos en cuanto a usabilidad, capacidad y robustez del esquema propuesto. Finalmente, en el Capítulo 6 se mencionan las conclusiones y el trabajo futuro.

2

Marco teórico

Este capítulo describe los conceptos necesarios relacionados a la solución que se propone en este trabajo, para ello se realiza una introducción a las marcas de agua digitales y sus aplicaciones. Posteriormente, se describe el objetivo de los esquemas de huella digital y sus características, además de los aspectos principales referentes a las bases de datos relacionales, el cual fue el medio digital bajo el que se realizó el esquema propuesto, y finalmente se describen las aplicaciones de huella digital para bases de datos relacionales.

2.1. Introducción a marcas de agua digitales

Las marcas de agua digitales, son un enfoque de ocultamiento de información iniciado en el año de 1988 por Komatsu y Tominaga, quienes fueron los primeros en adoptar el término marca de agua digital. Fue entonces que a partir del año de 1990 el interés por las marcas de agua digitales aplicadas a distintos medios digitales fue creciendo, principalmente en audio, imagen y video (Cox et al., 2007).

Sin embargo, pese a que los esquemas de marcas de agua digital fueron inicialmente contemplados para datos multimedia, su aplicación ha sido extendida a bases de datos

relacionales siendo investigada principalmente para proponer soluciones que mejoren su rendimiento en las siguientes aplicaciones:

- Afirmación de propiedad: La afirmación de propiedad puede ser aplicada utilizando esquemas de marcas de agua digital mediante la inserción de información dentro del contenido digital (imagen, audio, video, base de datos, entre otros). Un claro ejemplo es mostrado por Halder et al. (2010), donde plantea el siguiente escenario, suponiendo que Alicia puede insertar una marca de agua digital dentro de su base de datos relacional R, usando algunos parámetros privados como una llave secreta conocida solo por ella, Alicia entonces puede hacer pública su base de datos marcada. Posteriormente, si Alicia sospecha que una relación S publicada por Mallory ha sido copiada de su relación R sin previa autorización, puede realizar el reclamo de propiedad comprobando la presencia de su marca de agua digital en la relación S. Este tipo de esquemas es considerado como robusto, debido a que la marca de agua debe persistir a cualquier tipo de manipulación que la intente remover.

- Huella digital: Este tipo de aplicación conocida también como rastreo del traidor, es utilizada con el objetivo de identificar de manera única un medio digital a través de una marca de agua en el contexto de distribución del medio digital a múltiples usuarios. En el caso de una base de datos relacional, cada copia distribuida de la base de datos es marcada con un identificador diferente, conocido como huella digital, por lo cual, si una copia no autorizada de la base de datos es encontrada, el usuario perteneciente a dicha copia puede ser determinado recuperando la huella digital (Halder et al., 2010). En este caso, el usuario responsable del mal uso de la copia de la base de datos es conocido como el traidor Cox et al. (2007). Por otro lado, el término atacante se refiere al usuario que tiene la intención de remover o distorsionar la huella digital para evitar su identificación.

- Fraude y detección de manipulación: El objetivo de esta aplicación es asegurar la autenticidad e integridad del contenido del medio digital, es decir identificar y

localizar los cambios o manipulaciones, así como la posible recuperación de los datos modificados. De acuerdo a Halder et al. (2010), esta aplicación es utilizada cuando el contenido de una base de datos es usado para aplicaciones críticas como transacciones comerciales o aplicaciones médicas, donde la integridad de los datos es vital para las organizaciones. Esto se realiza mediante la inserción de una marca de agua digital en el medio digital y el análisis de integridad de esta marca posterior a su extracción, con el objetivo de verificar si la base de datos ha sido manipulada.

2.1.1. Clasificación de las marcas de agua digitales

En el caso de marcas de agua digitales para bases de datos relacionales su clasificación puede extenderse a los siguientes aspectos (Li, 2008):

- Tipo de dato: En este aspecto, el diseño de esquemas de marcas de agua digitales se basa en la utilización de datos numéricos y/o categóricos, siendo los datos numéricos los más utilizados, debido a que su modificación no es perceptible al usuario de forma directa, como lo es en la modificación de caracteres de un dato categórico (Li, 2008).

- Distorsión: Los esquemas de marcas de agua digitales puede ser de dos tipos, con distorsión, es decir introduciendo pequeños errores en los valores de la base de datos durante el proceso de inserción, o de libre distorsión, al no alterar los valores a través de patrones encontrados en el medio digital, que sirvan como información de la marca de agua, (Li, 2008).

- Susceptibilidad a ataques: Un esquema de marcas de agua digitales puede ser robusto o frágil a ataques a la base de datos. Se define como robusto cuando el esquema es capaz de recuperar la marca de agua contenida en la base de datos, posterior a un ataque con la intención de remover o distorsionar esta información. Sin embargo, si la marca de agua no puede ser recuperada con la mínima manipulación

el esquema se denomina frágil. Por otro lado, cabe mencionar que la usabilidad de los datos puede verse comprometida de acuerdo a la severidad del ataque, por lo cual el atacante deberá determinar el porcentaje de severidad que vaya de acuerdo a sus objetivos.

Los esquemas robustos son utilizados para aplicaciones como la protección de derecho de copia, afirmación de propiedad y huella digital, mientras que los esquemas frágiles son utilizados para la detección y localización de manipulaciones (Li, 2008).

- Información de marca de agua: La información insertada dentro de la base de datos puede ser de cuatro tipos: un bit, múltiples bits, una huella digital o múltiples marcas de agua Li (2008).

- Verificabilidad: Un esquema de marca de agua digital puede ser privado o público, se dice privado si la detección de la marca de agua digital solo puede ser obtenida por el propietario quien conoce la llave secreta. Por otro lado, se dice público si la detección de la marca puede ser verificada por cualquier usuario sin el uso de parámetros secretos, (Li, 2008).

- Estructura de datos: Esta clasificación se refiere a las diferentes estructuras de datos donde puede ser aplicado un esquema de marca de agua, como puede ser, bases de datos relacionales, cubos de datos, transmisión de datos o datos XML, (Li, 2008).

2.2. Huella digital

En la sección anterior se denomina huella digital, a la aplicación de marcas de agua digitales sobre un contexto de distribución de copias de un medio digital, con la principal diferencia que un identificador único denominado huella digital es insertado en cada copia que es distribuida.

La figura 2.1, muestra un esquema general del funcionamiento de un esquema de huella digital el cual se compone de dos componentes principales, la inserción de la huella digital, la cual consiste en la inserción de un identificador, comúnmente una cadena binaria, dentro de un medio digital(imagen, audio, video, RDB, entre otros) que será distribuido a un usuario específico. Cabe mencionar que este proceso es repetido de acuerdo al número de usuarios con los que se compartirá el medio digital. Sin embargo, como se puede ver en la parte central de la figura, el medio digital marcado puede estar sujeto a ataques con el objetivo de remover la huella digital insertada. En este caso se muestra el ataque de colusión, el cual será explicado más adelante.

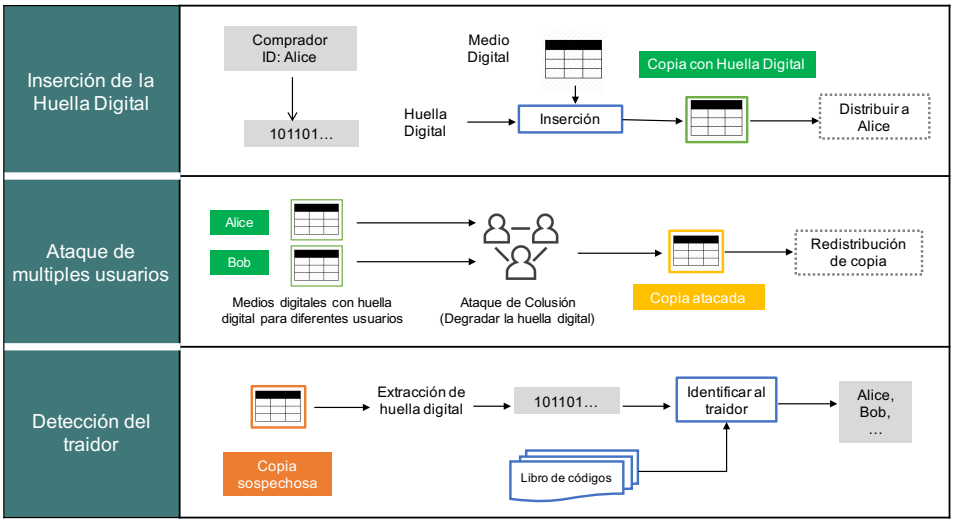


Figura 2.1: Esquema general de huella digital (Liu, 2005).

Finalmente, en la segunda etapa del esquema se muestra que dada una copia distribuida ilegalmente, el esquema deberá ser capaz de extraer la huella digital insertada en el medio digital e identificar al usuario traidor, el cual facilitó la redistribución. Por otra parte, la huella digital utilizada durante estos procesos, es de gran importancia, ya que juega un papel importante en la robustez y distorsión que ocasiona al medio digital. A continuación se describe el método utilizado para la generación de las huellas digitales utilizadas en este trabajo.

2.2.1. Generación de la huella digital

Uno de los procesos más importantes en los esquemas de huella digital, es la generación de la huella digital, ya que determinará la información utilizada como identificador del medio digital que se pretende marcar. Aunque se verá más tarde que esta información ha sido definida por distintos esquemas como el valor de una función Hash, una cadena de caracteres o una cadena de bits aleatorios, distintos autores han desarrollado esquemas de generación de huella digital específicos que aportan mayor seguridad a los esquemas de huella digital.

Principalmente, estos esquemas han sido desarrollados para contrarrestar el ataque de colusión, ya que este puede provocar la acusación de usuarios inocentes o la modificación completa de la huella digital insertada en un medio digital a través de la colaboración de un grupo de usuarios para detectar las posiciones en las que se encuentra la huella digital. A continuación se describe el esquema de generación de huellas digitales de Tardos, el cual fue seleccionado para la construcción de las huellas digitales utilizadas en este trabajo, debido a sus características y tamaños de longitud de huella digital que produce.

G.Tardos

Tardos (2008), propuso la construcción de un esquema de huella digital mejorando los códigos propuestos por Boneh and Shaw (1998), garantizando la identificación de al menos un traidor de una coalición de usuarios, derivado de un ataque de colusión.

El esquema para la construcción de la matriz de código de Tardos, se puede observar en la figura 2.2. Esta se basa en tres parámetros principales: el número de usuarios n,el tamaño de colusión c, y el límite de error ϵ , los cuales son definidos por el propietario o distribuidor del medio digital que será marcado con los códigos generados. Posteriormente, la longitud del código es definida como, $\mathrm{m} = 100\mathrm{c}^2\mathrm{k}$, siendo $\mathrm{k} = \lceil \log(1/\epsilon) \rceil$. Como paso siguiente, un vector de probabilidades es calculado y la matriz de código $X(n \times m)$

es construida, obteniendo para cada posición $1 \leqslant i \leqslant m$ y cada usuario $1 \leqslant j \leqslant n$, un valor X_{ji} asociado al valor p_i del vector de probabilidades (Laarhoven, 2011).

Es necesario mencionar que la definición del número de usuarios n no es definitiva, ya que una ventaja del esquema es la incorporación de nuevos usuarios de acuerdo a las necesidades de distribución del medio digital, esto dado que se basa en c y ϵ para encontrar la longitud m, así como los valores p_i, (Tardos, 2008).

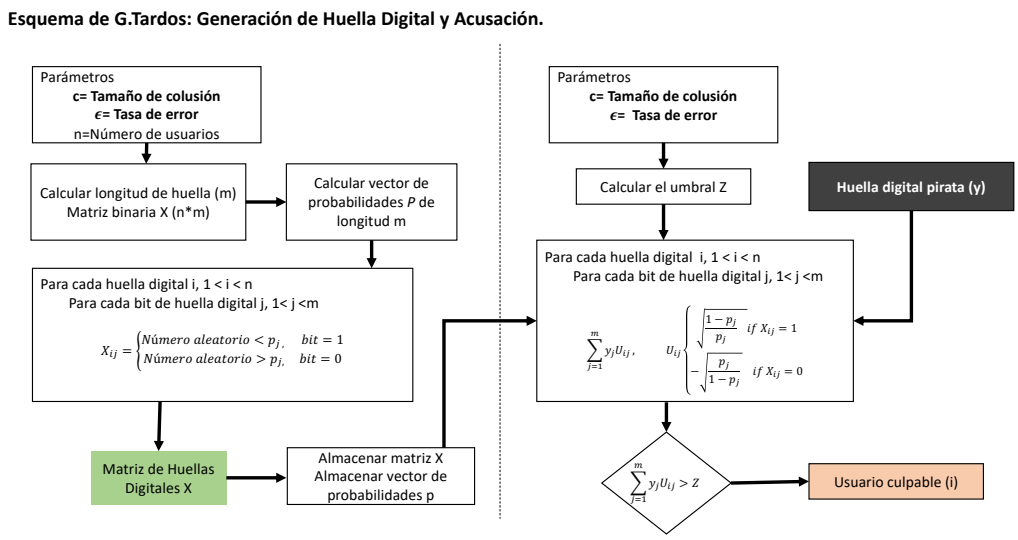


Figura 2.2: Acusación y generación de código de huella digital de G.Tardos.

La siguiente fase del esquema de Tardos, es el algoritmo de acusación, ilustrado en la parte derecha de la figura 2.2. Su funcionamiento requiere de los valores p_i y la matriz X, además del tamaño de colusión c y el límite de error ϵ definidos durante la construcción de la matriz de códigos. De acuerdo al diagrama un umbral Z es calculado como $Z = 20ck$, posteriormente, dada una huella digital pirata (y), el algoritmo determinará al usuario culpable i, si $\sum_{j=1}^{m} y_j U_{ij} > Z$, donde U es una matriz de valores basados en la matriz de códigos y el vector de probabilidades.

Como se mencionó al principio de la sección, el esquema de Tardos fue desarrollado como solución al problema de la coalición de usuarios, por lo cual el correcto funciona-

miento del esquema implica que la inserción de los códigos generados se realice en las mismas posiciones para todas las copias a distribuir, ya que solo las mezclas entre bits del mismo índice del código generan la información suficiente para la identificación de los usuarios de la coalición.

2.2.2. Rastreo del traidor

El rastreo del traidor o proceso de identificación del usuario culpable fue introducido por (Chor et al., 1994), refiriéndose a un mecanismo para proteger la distribución no autorizada de datos, a través de la información insertada en cada copia distribuida legalmente. Esto en el sentido de que los usuarios malintencionados (traidores) pueden generar copias para redistribuirlas ilegalmente. Esta característica, forma parte del proceso de detección del esquema de huella digital, y es ejecutada posterior a la recuperación de la huella digital, realizando una comparación con las huellas digitales que fueron distribuidas a todos los usuarios inicialmente, con el objetivo de determinar una coincidencia e identificar al usuario con su huella digital asignada.

Este proceso es realizado a través del esquema de Tardos, como puede observarse en la parte derecha de la figura 2.2, donde dada una huella digital y extraída de una copia distribuida ilegalmente, se realiza la comparación con la matriz de huellas digitales X y el sumatorio del valor U_{ij} de acuerdo a las coincidencias y diferencias entre los bits de la huella y y cada una de las huellas de la matriz. Por lo tanto, si el sumatorio es mayor al umbral Z, el usuario i será culpable.

2.3. Bases de datos relacionales

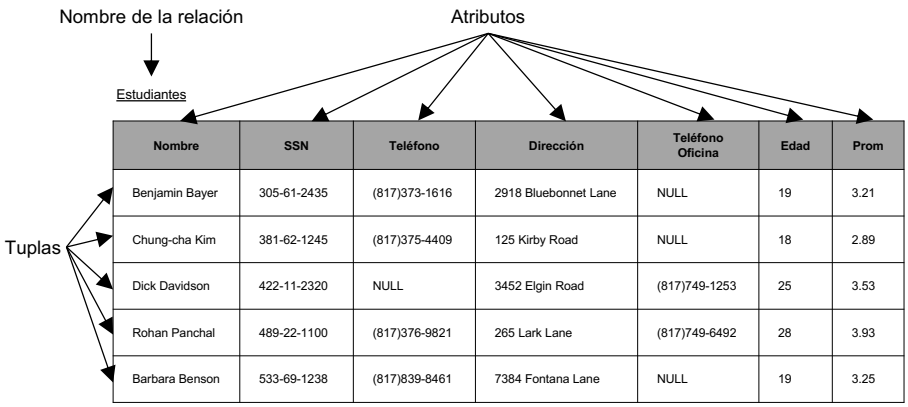
En esta sección se describen las características y conceptos referentes a bases de datos relacionales, debido a que es el medio digital bajo el que se desarrolló el esquema de huella digital propuesto.

El modelo de datos relacional, fue presentado por Codd en 1970 mediante el artículo titulado "Un modelo de datos relacional para grandes bancos de datos compartidos", siendo el inicio del cambio de paradigma hacia las bases de datos relacionales, cuyo objetivo principal fue la independencia entre las aplicaciones y los datos (Piattini, 2000).

Actualmente es el modelo de datos más utilizado en diferentes áreas de la ciencia, principalmente en la industria y compañías que requieran del almacenamiento de grandes cantidades de datos. A continuación se menciona el concepto formal y las principales características que componen una base de datos relacional, ya que será fundamental para la comprensión de este trabajo.

Una Base de Datos Relacional (RDB, por sus siglas en inglés), es un medio digital para el almacenamiento de datos, representada a través de tablas compuesta de filas y columnas, conocidas también como relaciones.

El formato de una relación de acuerdo a Maier (1983), está determinado por un conjunto finito de columnas, conocidas como atributos. Cada uno de los atributos $A_1, A_2, ..., A_n$ tiene un dominio, denotado como $dom(A_i)$, el cual definirá los posibles valores que puede tomar cada atributo. Además, las relaciones se componen de un conjunto de filas denominadas tuplas $t_1, t_2, .., t_p$, las cuales tiene como restricción que para cada $t(A_i)$, debe estar en el $dom(A_i)$. Un ejemplo de relación se puede ver en la figura 2.3, la cual contiene información referente a estudiantes.



Nombre	SSN	Teléfono	Dirección	Teléfono Oficina	Edad	Prom
Benjamin Bayer	305-61-2435	(817)373-1616	2918 Bluebonnet Lane	NULL	19	3.21
Chung-cha Kim	381-62-1245	(817)375-4409	125 Kirby Road	NULL	18	2.89
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	(817)749-1253	25	3.53
Rohan Panchal	489-22-1100	(817)376-9821	265 Lark Lane	(817)749-6492	28	3.93
Barbara Benson	533-69-1238	(817)839-8461	7384 Fontana Lane	NULL	19	3.25

Figura 2.3: Ejemplo de una relación de la RDB (Elmasri and Navathe, 2010).

Adicionalmente, uno de los aspectos de mayor importancia dentro de una relación, es la llave primaria, la cual es definida a través de uno o más atributos de la relación, y es encargada de que las tuplas de la relación se distingan entre ellas mediante un valor único determinado por el usuario o administrador de la base de datos,(Maier, 1983).

Además de estas características, la naturaleza de las bases de datos relacionales permite la realización de cambios a través del tiempo, tales como la adición, eliminación y actualización de los datos contenidos en la RDB.

2.4. Huella digital para bases de datos relacionales

La aplicación de esquemas de huella digital y/o marcas de agua digitales en bases de datos relacionales, ha significado un nuevo reto dentro del área debido a las diferencias en sus características con respecto a los datos multimedia, en los cuales la aplicación de estos esquemas se encuentra mayormente consolidado. Los aspectos principales en los que difieren estos medios digitales se mencionan a continuación (Mehta and Aswar, 2014; Halder et al., 2010).

- Redundancia de datos: Los datos multimedia se componen de una gran cantidad de bits con redundancia considerable, lo cual permite una mayor área de inserción debido a que los cambios en los valores son minimizados. A diferencia de estos, una relación consiste en un conjunto de tuplas independientes entre ellas, y por lo tanto los valores muestran mínima redundancia.

- Orden de los datos: Una relación no considera el orden entre sus tuplas, ya que no representa un cambio en el significado de la base de datos. Por otro lado, los datos multimedia están sujetos a un orden específico para dar significado a su contenido.

- Frecuencia de actualizaciones: La inserción, eliminación y actualización de tuplas, son operaciones propias de las bases de datos relacionales. Por otro lado, la eliminación o remplazo de porciones de datos en datos multimedia no esta permitido o no es una característica de estos datos.

2.4.1. Requerimientos

Además de las dificultades que implica el diseño de un esquema de huella digital o marca de agua digital mencionadas anteriormente, algunos autores como Alfagi et al. (2017), mencionan algunos de los requerimientos que caracterizan este tipo de esquemas, los cuales deben ser considerados para el desarrollo de un esquema de marca de agua digital.

Capacidad: Definida como la cantidad máxima de bits que puede ser insertada en una base de datos relacional, sin exponer la usabilidad de los datos. En los esquemas de huella digital, la capacidad determina el nivel de redundancia de la huella digital insertada.

Usabilidad: La inserción de la marca de agua o huella digital, provoca cambios en los datos de la bases de datos, los cuales no deben degradar la usabilidad de los datos, sin embargo, la distorsión causada durante este proceso difiere de una base de datos a

otra, ya que intervienen factores como el número de tuplas y la cantidad de atributos modificables.

Robustez: La marca de agua o huella digital insertada debe ser persistente ante ataques maliciosos que pretendan removerla.

Seguridad: La posición usada para la inserción de los bits de marca de agua o huella digital debe estar determinada por parámetros secretos (ej. llave secreta) asignados por el propietario de la base de datos. Es decir, la seguridad recae en el secreto de los parámetros y no en el algoritmo.

Ciego: Referente al conocimiento necesario durante el proceso de recuperación de la huella digital de una base de datos marcada. Este proceso no debe requerir del conocimiento de la base de datos original o de la huella digital.

Sistema público: Se asume que el algoritmo de inserción de la marca de agua o huella digital es público, basando la seguridad del esquema en los parámetros secretos, como se mencionó anteriormente, (Halder et al., 2010).

No ambiguo: La recuperación de la marca de agua o huella digital deberá identificar claramente al propietario de la base de datos analizada, (Xie et al., 2016). En el caso de los esquemas de huella digital, la identificación del traidor se lleva a cabo como se explicó en la sección 2.2.2.

2.4.2. Ataques

Los esquemas de huella digital, pertenecen al grupo de esquemas robustos de marca de agua digital, debido a que la huella digital insertada en la base de datos debe ser persistente a cualquier tipo de manipulación intencional o no, que busque remover dicha información. Por otro lado, es preciso mencionar que la severidad del ataque o manipulación de los datos se verá reflejada en la usabilidad de la base de datos.

En esta sección se exponen ataques que han sido investigados por esquemas del estado del arte.

Actualizaciones benignas

Como se menciono anteriormente, las bases de datos relacionales permiten la alteración de sus relaciones a través del tiempo, mediante la inserción, eliminación y alteración de tuplas. En sus artículos, Halder et al. (2010) y Alfagi et al. (2017), denominan a estas operaciones como actualizaciones benignas, debido a que una relación marcada es procesada con las mismas características definidas para una relación, siendo la actualización de los datos parte de la naturaleza de las bases de datos. Sin embargo, la modificación de estos datos puede alterar o remover los bits de la marca de agua digital previamente insertada de forma no intencional.

Por otro lado, Xie et al. (2016), menciona estas operaciones como parte de la intención de un atacante de remover la huella digital. Estas operaciones son descritas de la siguiente manera y son aplicables bajo las dos definiciones.

- **Ataque de inserción:** Consiste en la inserción de nuevas tuplas, con la finalidad de alterar la cantidad de bits de la marca de agua digital recuperados durante el proceso de detección, y finalmente obtener una huella digital errónea.

- **Ataque de eliminación:** El usuario o atacante elimina tuplas de la relación, con el objetivo de remover bits de la huella digital insertada previamente.

- **Ataque de alteración:** El usuario o atacante modifica los valores de los atributos de las tuplas en la relación, con el objetivo de alterar bits de la huella digital. Otros ataques, como lanzamiento de bits, son englobados dentro de este tipo de alteración.

Ataque de modificación del valor

- **Ataque de bit:** Este ataque consiste en distorsionar la marca de agua o huella digital insertada en la relación de la base de datos alterando uno o más bits en los valores marcados. Este tipo de ataque puede ser realizado de forma aleatoria, a lo cual se denomina ataque por aleatorización, ya que los bits que serán insertados son generados de forma aleatoria. Otra variante de este ataque, es la asignación de ceros en los bits objetivos y finalmente, el ataque de lanzamiento de bits, que consiste en invertir los valores de los bits que se quieren afectar, (Halder et al., 2010).

Ataque de sub-conjunto

Consiste en la eliminación o actualización de un subconjunto de tuplas o atributos de la relación marcada, con la intención por parte de un atacante de remover o distorsionar la marca de agua insertada, (Halder et al., 2010).

Ataque de super-conjunto

A diferencia del ataque de subconjunto, nuevas tuplas o atributos son agregados a la relación con el objetivo de afectar la detección de la marca de agua digital, Halder et al. (2010).

Falso reclamo de propiedad

- **Ataque aditivo:** En este ataque, el atacante inserta su propia marca de agua a una relación previamente marcada, con la intención de adjudicarse la propiedad de la misma y generar dudas ante el reclamo del propietario, Halder et al. (2010).
- **Ataque de invertibilidad:** Consiste en que el atacante pueda descubrir una mar-

ca de agua ficticia la cual puede ser obtenida a partir de una ocurrencia de bits aleatoria de la base de datos marcada, Halder et al. (2010).

Ataque de colusión

El ataque de colusión consiste en la coalición de un grupo de usuarios con diferentes versiones marcadas de un medio digital que trabajan juntos para generar una versión del medio digital marcado con una huella digital que no pueda ser identificada durante el proceso de detección de huella digital, (Halder et al., 2010). Para medios digitales como las Bases de Datos Relacionales existen dos tipos principales de ataque de colusión que se mencionan a continuación.

- **Mayoría:** En este ataque una nueva versión de la relación de la RDB es creada tomando el valor de cada bit calculado como la función de mayoría de los valores de bits correspondientes a cada copia de la relación que posee cada usuario de la coalición, (Alfagi et al., 2017). En este ataque se supone que el atacante no altera el medio digital en las posiciones donde todas las copias están de acuerdo, (Tardos, 2008).

- **Mezcla y Correspondencia:** Cosiste en la generación de una nueva versión de la RDB a partir de la combinación de tuplas de diferentes versiones o copias de la RDB, cada una de estas con una huella digital distinta,(Alfagi et al., 2017).

2.4.3. Métricas de evaluación

Durante la inserción de la huella digital, la modificación o introducción de pequeños errores en los valores de la base de datos son permitidos, por lo tanto, esto requiere de métricas que permitan determinar el grado de distorsión causado posterior a este proceso. Aunque distintos autores reportan como medida de distorsión la media y

varianza estadística, estos valores no aportan información objetiva del rendimiento de los distintos esquemas de huella digital en cuanto a la distorsión. Esto debido a que los cambios realizados entre dos valores de un atributo, pueden ser compensados entre ellos, manteniendo las estadísticas originales.

Observado este fenómeno,y considerando algunas de las estrategias de inserción de esquemas propuestos en el estado del arte, se determinó utilizar como medida de distorsión el error cuadrático medio, descrito a continuación.

Error Cuadrático Medio

El error cuadrático medio (MSE, por sus siglas en inglés), es una herramienta de medición de sensibilidad de error que se ha usado como medida de fidelidad entre señales, cuyo objetivo es comparar cuantitativamente dos señales y determinar el error o distorsión entre ellas (Ndajah et al., 2011). En marcas de agua digitales, ha sido utilizada para medir la diferencia por la cual un valor marcado difiere de su valor original.

Por lo cual, para dos relaciones de bases de datos d y d', donde d' es obtenida como resultado de insertar una marca de agua digital en d, el $mse(d, d')$ es definido en la ecuación 2.1, Lafaye et al. (2008).

$$mse(d, d') = \frac{1}{N} \sum_{i=1}^{N} (\vec{v}(d)[i] - \vec{v}(d')[i])^2, \qquad N = |d| \tag{2.1}$$

Esta ecuación es calculada para cada uno de los atributos de la base de datos, obteniendo el sumatorio de las diferencias entre el valor original y el valor modificado de cada una de las tuplas de la relación de la base de datos. Un MSE con valor 0, indicará que los valores no fueron distorsionados, por lo cual mientras más grande sea el valor de MSE, la distorsión sera mayor.

Error Relativo

Otra de las métricas de distorsión que fueron fundamentales para el desarrollo de este trabajo, es el error relativo. Esta medida permite conocer la distorsión generada respecto a un valor, posterior a su modificación, para esto es necesario conocer el error absoluto, el cual se calcula de la diferencia entre el valor real (V) y el valor modificado V', pudiendo ser positivo o negativo. Una vez obtenido este valor, el error relativo es calculado tomando el cociente entre el error absoluto y el valor real o el promedio de ambos valores, como se muestra en la ecuación 2.2

$$E_R = \frac{V - V'}{V} \tag{2.2}$$

De esta operación se puede obtener el error relativo porcentual, multiplicando por 100 este resultado, lo cual permite conocer el porcentaje de distorsión ocasionado en un determinado valor, después de su modificación.

De acuerdo al análisis de estas métricas de cuantificación de errores, se optó por utilizar el error relativo para calcular la distorsión ocasionada en cada atributo (A) de la base de datos, incorporando una ligera modificación, como se puede ver en la ecuación 2.3

$$A_{E_R} = \sum_{i=1}^{N} \frac{|V - V'|}{|V + V'|/2}, \qquad N = \#Tuplas \tag{2.3}$$

Esta modificación se realizó agregando el valor absoluto con el objetivo de omitir los errores negativos que pudieran producirse, además de casos donde el valor real es cero, pero existe una diferencia respecto al valor modificado. Dichos casos podrían mostrar un resultado erróneo del total de errores generados. Por otra parte, se incorpora el sumatorio de los errores de cada tupla de la RDB por cada uno de los atributos. De ésta manera, podemos observar la distorsión generada por cada valor modificado respecto a su error relativo.

2.4.4. Funciones criptográficas

Uno de los conceptos importantes de este trabajo es la función hash, ya que a través de esta función criptográfica se llevo el control de la inserción de la huella digital y su recuperación, lo cual se describe a mayor detalle en el Capítulo 4.

Una función hash es una función que toma una cadena de entrada de longitud variable y la convierte en una cadena de longitud fija generalmente menor a la de entrada. Esta función trabaja en un solo sentido, debido a que el valor hash debe ser fácil de calcular y computacionalmente difícil de calcular los datos de entrada a partir del valor hash (Schneider, 1996). La definición formal de una función hash de acuerdo a Stinson (2005) es la siguiente:

Sea h una función hash y sea x algún dato de entrada, pudiendo ser x una cadena de longitud arbitraria. El valor de la función hash, también conocido como digesto de mensaje es definido como $y = h(x)$, el cual es una cadena típicamente menor a la del dato de entrada.

Nosotros suponemos que y es almacenada en un lugar seguro, pero x no. Si x es modificado a x', esperamos que y no sea un digesto de mensaje para x'. En este caso podemos comprobar si x ha sido alterado calculando el digesto de mensaje $y' = h(x')$ y verificando que $y' \neq y$.

En este trabajo se presta principal atención al Algoritmo de Hash Seguro (SHA, por sus siglas en inglés),el cual fue diseñado por la Agencia de Seguridad Nacional (NSA, por sus siglas en inglés) y estandarizada por el Instituto Nacional de Estándares y Tecnología (NIST, por sus siglas en inglés), manteniéndola hoy en día como una de las principales funciones criptográficas más utilizadas (Ferguson et al., 2011). La función hash utilizada para el desarrollo del esquema propuesto fue SHA-256, debido al fácil acceso para ser implementada, además de los niveles de seguridad que mantiene vigentes hoy en día.

3

Estado del arte

En este capítulo, se describen los trabajos y técnicas que han sido desarrollados en el área de marcas de agua digitales para bases de datos relacionales y que son relevantes para este trabajo. Además, se describen los distintos enfoques que se han desarrollado en esquemas de huella digital y finalmente se realiza una comparativa y discusión de estos trabajos.

3.1. Marca de agua digital en bases de datos relacionales

La aplicación de marcas de agua digitales a bases de datos relacionales ha sido investigada a partir del trabajo de Agrawal and Kiernan (2002), lo cual significó nuevos retos en el área debido a las diferencias respecto a los medios digitales multimedia como se mencionó en el capítulo anterior. Desde entonces, los avances que se han realizado han sido cimentados en tres técnicas principales, las cuales se mencionan a continuación.

3.1.1. Técnicas básicas

- Bit menos significativo (LSB, por sus siglas en inglés): La modificación de los bits menos significativos ha sido una técnica ampliamente utilizada para el ocultamiento de información en marcas de agua digitales. Sin embargo, a diferencia de los medios digitales como imágenes, audio y vídeo, donde la percepción humana juega un papel importante en el ocultamiento de la marca de agua, las bases de datos relacionales no muestran este factor, y los cambios en los bits menos significativos de los valores impactan de forma directa la usabilidad del medio digital, (Xie et al., 2016).

- Partición de datos: La idea básica de esta técnica es la utilización de una llave secreta, una función hash y un número de particiones para asignar cada tupla de la relación a una partición, esto con el objetivo de aleatorizar el orden de inserción de los bits de la marca de agua (Xie et al., 2016).

- Votación por mayoría: El objetivo de esta técnica, es obtener una correcta recuperación de los bits de la marca de agua, debido a que posibles ataques o manipulaciones a los datos marcados pueden cambiar el valor del bit original con el que fue marcado (Xie et al., 2016). Esto sucede ya que durante el proceso de inserción cada bit de la marca de agua digital puede ser insertado más de una vez.

  Por ejemplo, para determinar el valor de un bit de la marca de agua digital con índice i, durante el proceso de detección podemos observar que fue encontrado con un valor de 1 en cinco ocasiones y un valor de 0 en tres ocasiones, por lo cual, utilizando la función de votación por mayoría, obtenemos que el valor original para ese bit es 1. Li et al. (2005), emplea esta función de la siguiente manera, donde $\tau \epsilon [0.5, 1]$ y $count[i][0]$ o $count[i][1]$, es la cantidad de 1s o 0s encontrados para el bit i.

$$\begin{cases} 0 & \text{si } \frac{count[i][0]}{(count[i][0]+count[i][1])} > \tau \\ 1 & \text{si } \frac{count[i][1]}{(count[i][1]+count[i][0])} > \tau \end{cases} \tag{3.1}$$

Agrawal y Kiernan

Como se mencionó anteriormente, el trabajo de Agrawal and Kiernan (2002), fue el inicio de la investigación en marcas de agua digitales para bases de datos relacionales, además de que su esquema de marcas de agua fue la base de muchos otros esquemas posteriores. Este esquema consistió en dos fases principales: la inserción y detección de la marca de agua digital. El algoritmo de inserción está basado en la modificación de valores de atributos numéricos, los cuales son seleccionados por el propietario de la base de datos. Por su parte, las modificaciones a los valores para la inserción de los bits de la marca de agua son realizados en los bits menos significativos (LSB). Cabe mencionar que el número de LSB disponibles para modificar es seleccionado por el propietario, así como el parámetro γ, que indica la cantidad de tuplas disponibles a marcar.

Dados estos parámetros, la selección de la tupla, atributo y LSB a modificar, se determina a través del cálculo de un Código de Autenticación de Mensaje (MAC, por sus siglas en inglés), donde los parámetros de entrada son la llave secreta, conocida solo por el propietario, y la llave primaria de la tupla analizada en ese momento. Posteriormente, mediante la MAC calculada se determina si el bit de marca sera 0 , cuando el valor obtenido es par y 1 si el valor es impar. De esta manera, las marcas de agua insertadas solo podrán ser recuperadas por el propietario que conoce la llave secreta.

La detección de la marca de agua es realizada mediante el cálculo de la MAC a través de la llave secreta y llave primaria, con el objetivo de encontrar la tupla, atributo y LSB donde fue insertada la marca, y posteriormente contabilizar el total de marcas insertadas y la cantidad de marcas recuperadas correctamente para determinar al propietario de la base de datos.

3.2. Huella Digital en bases de datos relacionales

A continuación, se hace una recopilación de los trabajos desarrollados en el área huellas digitales para bases de datos relacionales, sus características y estrategias relacionadas a los algoritmos de inserción y detección de huella digital, además de aspectos que complementan su aplicación a este tipo de medio digital.

En 2003, Li presentó un reporte introduciendo una extensión de las marcas de agua digitales, para posteriormente en (Li et al., 2005), presentar el primer esquema de huella digital para bases de datos relacionales basado en las características presentadas por Agrawal and Kiernan (2002), con el objetivo de aportar un mecanismo más de seguridad a las RDB, más allá de mecanismos de seguridad como la criptografía y herramientas de control de acceso. La principal característica en este esquema es la creación de cadenas binarias únicas para la identificación de usuarios o compradores, a través de su inserción en cada copia de la RDB que vaya a ser distribuida. En primer lugar Li et al. (2005) sugiere generar esta cadena binaria aplicando una función Hash, utilizando como entradas la concatenación de una llave secreta, la cual es asignada por el propietario de la RDB, y un número de serie, el cual será público y único para cada uno de los usuarios.

Por otra parte, el esquema es dividido en dos etapas, el algoritmo de inserción, el cual consiste principalmente en la generación de números pseudoaleatorios, inicializados con la llave secreta y llave primaria, para la selección de la tupla, atributo y LSB a modificar durante la inserción de los bits de la huella digital, además de la selección del bit de la huella digital correspondiente a la posición seleccionada. Cabe mencionar que el algoritmo incluye el enmascaramiento de los bits de la huella digital, como parte de un mecanismo de seguridad para ocultar la distribución de los bits durante la inserción. Por su parte, el algoritmo de detección tiene como principal función, la identificación del usuario culpable posterior a la redistribución de una copia de la RDB no autorizada, su funcionamiento consiste en la selección de la tupla, atributo y LSB, de la misma forma

que el algoritmo de inserción, esto con el objetivo de ubicar las posiciones de inserción de la huella digital para su extracción. Finalmente, la función de votación por mayoría es utilizada y la huella digital encontrada es comparada con las huellas digitales existentes, para la identificación del usuario culpable.

La desventaja de este esquema es que la pérdida de al menos un bit de la huella digital no permite la identificación del usuario, así como la utilización de estas cadenas binarias no permite la resistencia a ataques de colusión, por ello, Li et al. (2005) propone la utilización del esquema de Boneh and Shaw (1998). Por otra parte, tanto este esquema como los esquemas de marcas de agua digital en general tienen como principal desventaja la dependencia de la llave primaria para su funcionamiento, lo cual implica que cualquier manipulación o eliminación de ésta, inhabilite el algoritmo de detección de la huella digital.

En los mismos años, Liu et al. (2004), publica un esquema de huella digital basado en la inserción de bits de huella digital mediante la construcción de bloques de los bits menos significativos de los valores de la relación de la RDB. El tamaño de estos bloques es seleccionado por el propietario de la RDB y determina el nivel de distorsión ocasionado ya que si se reduce el tamaño de los bloques, la capacidad de inserción tiende a aumentar. Por otro lado, la selección de la posición del bloque a modificar, así como la huella digital a insertar es determinada mediante el valor de una función hash tomando como entrada la llave primaria y la llave secreta. Por su parte, la robustez de este esquema solo es resistente a ataques de lanzamiento de bits y ataques de subconjunto, debido a que la inserción de la huella se realiza de forma secuencial y cualquier modificación al orden de las tuplas impide la detección de la huella digital.

Otra de las técnicas de inserción mencionadas en la sección anterior es abordada en el esquema de Guo et al. (2006), y consiste en la partición de registros distribuidos en l grupos, donde l representa la longitud de la huella digital, por lo cual a cada bit de la huella le corresponde un grupo del cual se determina qué tuplas serán marcadas. Además,

una segunda inserción es realizada con el objetivo de definir un único patrón para cada relación distribuida y determinar la propiedad de las copias independientemente de la huella digital, es decir realizando una doble inserción. En este esquema también se realiza la selección del grupo, tupla y LSB, a través de la utilización de una función hash.

Más tarde, Zhou et al. (2007) propone un esquema de huella digital basado en las estrategias vistas anteriormente como la selección de tupla y bit a insertar mediante la utilización de una función hash. Sin embargo, su principal aportación es la inserción de la huella digital cada vez que un usuario accede a la RDB con el objetivo de que la respuesta a consultas muestre los valores marcados correspondientes con el usuario propietario de dicha base de datos. Por otra parte, Lafaye et al. (2008) propone el primer esquema de huella digital con un enfoque basado en la construcción de un lenguaje de restricciones de usabilidad en el que considera restricciones locales (restricciones que afectan a tuplas individuales) y globales (restricciones que afectan a un subconjunto de tuplas) para la inserción de la huella digital. Por ello, previo al proceso de inserción, el propietario de la RDB define las condiciones bajo las que se realizarán las modificaciones. Además, la selección de tuplas a marcar es realizada a través del algoritmo de inserción basado en el trabajo de Agrawal and Kiernan (2002) y Li et al. (2005), con la única diferencia de que las tuplas modificadas son almacenadas para su utilización en el proceso de detección. Cabe mencionar que cuando las restricciones no son definidas, el esquema tiene el mismo comportamiento que en (Agrawal and Kiernan, 2002).

Por otra parte, una de las características interesantes durante el proceso de inserción, es la utilización de una heurística denominada emparejamiento, que consiste en la compensación de la modificación realizada en una tupla con su correspondiente modificación en la siguiente tupla, para reducir la distorsión ocasionada. Dada esta estrategia, la distorsión es calculada utilizando el error cuadrático medio (MSE), a diferencia de los esquemas anteriores donde la utilización de estadísticas como la media y la desviación estándar son las métricas reportadas.

Con un distinto enfoque, Uzun and Stephenson (2008) de los laboratorios de Hewlett-Packard, proponen dos esquemas de marcas de agua digitales, los cuales posteriormente mencionan pueden ser extendidos a esquemas de huella digital. El primer esquema consiste en la permutación de valores de un atributo entre diferentes tuplas, seleccionadas de acuerdo a una función de distancia. La selección de tuplas y bits de la huella digital es realizada a través de un generador de números pseudoaleatorios, el cual es controlado a través de una entrada inicial, para su posterior replica durante la detección de la huella digital, como se ha mencionado a lo largo de este capítulo con la utilización de la función Hash. En el segundo esquema, se propone la creación de tuplas virtuales para la inserción de la marca de agua, esto debido a que bases de datos podrían no permitir la distorsión de los datos originales y si la inserción de datos extra. Sin embargo, el número de tuplas virtuales dependerá de la cantidad de bits de marca de agua o huella digital a insertar, lo cual significa distorsionar en gran medida la base de datos.

En este punto, se puede observar que uno de los componentes principales en todos los esquemas es la huella digital, sin embargo, como se vio en el capítulo anterior, es necesario la construcción de códigos resistentes a ataques de colusión, por lo cual Bui et al. (2013) propone la construcción de códigos a partir de una matriz d-disjunta, la cual mejora la propuesta de Tardos (2008), en cuanto al tiempo de decodificado de la huella insertada, además de tener como objetivo la anonimización de los atributos de la RDB. Sin embargo, la cantidad de usuarios que soporta depende del número de atributos de la RDB, por lo tanto, no es viable para bases de datos pequeñas. Más tarde, en su artículo Bui et al. (2015), propone una guía para la venta de bases de datos legales con evaluación de riesgo y lucro, aplicando el concepto de k-anonimidad derivado de su trabajo anterior, esta vez visto como una generalización de los valores de los atributos con el objetivo de proteger la privacidad de la información. Este concepto es abordado más adelante por Kieseberg et al. (2014), como un problema de sensibilidad de la información y leyes que regulan su manejo en la distribución de bases de datos.

Por su parte, Iftikhar et al. (2014), crea una firma secreta para la autenticación de usuarios mediante un algoritmo genético y dos parámetros secretos únicos para cada usuario, los cuales determinarán las modificaciones realizadas a los valores de la RDB en la inserción de la firma. En este proceso de inserción, los cambios realizados a los valores son almacenados y utilizados durante la detección de la huella digital.

Posteriormente, basado en el modelo de restricción de usabilidad de Kamran and Farooq (2013), Waghmode and Mohanpurkar (2014) introducen un paso de verificación de distorsión del sistema a través de algoritmos de clasificación, ya que si los datos son utilizados en una tarea de minería de datos, el conocimiento devuelto debería ser el mismo a los datos originales. A la par de este trabajo, Gursale and Arti (2014) se enfocan en la reducción de la distorsión a través de la doble selección de tuplas a marcar, tomando en cuenta la media y desviación estándar de particiones con el objetivo de limitar la modificación realizada en dichas tuplas.

Finalmente, Mohanpurkar and Joshi (2015c,b,a), retoman estos trabajos enfocados en la reducción del tiempo de ejecución del algoritmo de inserción, para lo cual proponen una selección general para todos los usuarios de las tuplas a marcar. Sin embargo, estos trabajos requieren del almacenamiento de una tabla de alteraciones para su posterior utilización en la detección de la huella digital, lo cual supone una desventaja ya que la pérdida de esta información imposibilita la recuperación de la huella digital, además de no ser resistente a ataques a la base de datos.

Un trabajo que cambia los enfoques mencionados anteriormente es el de Franco-Contreras and Coatrieux (2016), ya que se aborda el tema de rastreo de bases de datos relacionales en un almacén de datos, es decir la utilización de esquemas de huella digital para la identificación de relaciones dentro de una mezcla, lo cual supone nuevos retos. Sin embargo, la base de este esquema consiste en la partición en grupos de cada relación a marcar, con la finalidad de insertar un bit de la huella digital en cada grupo, y verificar la propiedad de cada relación en la mezcla.

Por último, Ahmad et al. (2017), propone un esquema basado en la construcción de tuplas falsas, similar a Uzun and Stephenson (2008), para la inserción de la huella digital, sin embargo, estas tuplas son generadas a partir del cifrado con RSA de la llave primaria de tuplas seleccionadas previamente, con lo cual a diferencia de Uzun and Stephenson (2008) no requiere del almacenamiento de las tuplas extras. Además, no considera la resistencia a ataques de colusión, lo cual compromete la integridad de la huella digital.

3.3. Discusión

Los esquemas de huella digital descritos hasta ahora constituyen los métodos propuestos bajo el escenario de distribución de copias de la RDB, realizados hasta el día de hoy, sobre esta aplicación de las marcas de agua digital. Derivado de esto, se puede observar que gran parte de estos trabajos han sido basados en la estrategia de selección de Agrawal and Kiernan (2002), es decir una selección determinista para el control y ubicación de la posición de los bits insertados, mediante generadores de números pseudoaleatorios o funciones Hash.

Por otro lado, la utilización de códigos seguros a ataques de colusión como Boneh and Shaw (1998) y Tardos (2008), ha resultado un reto en la búsqueda de reducir la distorsión ocasionada a la base de datos relacional, además de las estrategias para aumentar la robustez del esquema ante cualquier tipo de manipulación, ya sea a nivel de atributos o tuplas. Sin embargo, dos de los mayores problemas que se tienen hasta el momento, es la dependencia de la llave primaria de la relación para el funcionamiento de los esquemas y la longitud de las huellas digitales aplicables solo a bases de datos de gran tamaño, debido a la capacidad requerida y la distorsión que generan.

Además, otro factor importante, ha sido que el desarrollo de estos esquemas en su mayoría solo es aplicable a atributos numéricos con tolerancia a errores, lo cual deja expuestos tanto a atributos categóricos, como a aquellos que tienen mayor sensibilidad

a cambios, como pueden ser números de cuenta, números de seguridad social, etc.

Finalmente, se presenta un resumen con las principales características de los esquemas mencionados anteriormente a través de la tabla 3.1. Estas características son: la huella digital utilizada, los tipos de ataque soportados, métricas utilizadas para la evaluación de la distorsión, la capacidad de inserción y las bases de datos sobre la que fueron evaluados.

Como primer punto podemos observar las estrategias de inserción y funcionalidades desarrolladas para estos esquemas, siendo la utilización de generadores de números pseudoaleatorios y la creación de tuplas virtuales las principales técnicas implementadas para la inserción de la huella digital. Por otra parte, la implementación de códigos con tolerancia a la colusión es un factor que se observa en la mayoría de los trabajos del estado del arte, sin embargo en la siguiente columna de la tabla la mayor parte de estos trabajos no resiste un mayor número de ataques más allá del ataque de colusión.

Respecto a las métricas utilizadas para evaluar la distorsión generada por la inserción de la huella, la gran parte de los esquemas presentados utilizan medidas estadísticas como la media y varianza. Sin embargo como se mencionó en el capítulo anterior estas métricas no ofrecen valores reales de distorsión debido a la compensación entre la modificación de un valor y otro. Por otro lado, la estrategia de inserción para el control de la capacidad de inserción se basa en la selección de tuplas a través de la fracción de tupla o la creación de particiones de la base de datos, todo esto de acuerdo también a la base de datos utilizada.

En general, como podemos observar la descripción de algunas características de los esquemas presentados no son mostrados, lo cual en algunos casos impide su reproducción y análisis.

Tabla 3.1: Esquemas de huella digital del estado del arte.

Autor	Estrategia	Huella Digital	Ataques	Distorsión	Capacidad	Llave primaria	Conjunto de datos	Año
Li	Generador de secuencias pseudo-aleatorias criptográfico	Número de serie (MD5), Boneh & Shaw	Lanzamiento de bits, Invertibilidad, Colusión	Media y varianza	Basada en la fracción de tupla y secuencia pseudoaleatoria	si	Forest Cover Type	2005
Liu	Basada en el método de bloque, generador pseudoaleatorio, función Hash	Función Hash(MD5 o SHA)	Lanzamiento de bits, Subconjunto, Aditivo , Colusión	Media y varianza	De acuerdo a longitud de bloque	si		2005
Guo	Doble inserción (Huella digital y verificación de propiedad) utilizando función Hash	Código binario de 8 bits	Selección de subconjunto, Adición de subconjunto, Alteración de subconjunto.	Media y varianza	Basada en fracción de tupla	si	Forest Cover Type	2006
Zhou & J. Wang	Partición en grupos con función Hash y sub lenguaje en SQL	Código binario	Ninguno	Ninguno	No mostrado	si	Datos Médicos del Hospital Universitario de Chiba	2007
Lafaye	Pairing Algorithm para restricciones de suma independiente de peso, basado en el esquema de Agrawal and Li	Código de Tardos	Alteración de datos, Ataque de subconjuntos, Colusión (Equivalente a Agrawal cuando las restricciones de usabilidad no son establecidas	Almacena la distorsión para su utilización durante la detección. Utiliza MSE	Basado en la fracción de tupla y restricciones de usabilidad	si	Forest Cover Type	2008
Uzun & Stephenson	Permutación de atributos e inserción de tuplas virtuales	Código binario	Ninguno	Ninguno	No mostrado	si	No mostrado	2008
Thach V. Bui	Anonimización de atributos	Matriz t×N d-disjunta	Atributos, Colusión	Generalización de atributos	No mostrado	no	Forest Cover Type	2013
Saman Iftikhar	Operaciones aritmeticas en la inserción y extracción de la huella digital	Firma secreta	Eliminación de subconjunto, Alteración de subconjunto, Colusión	No mostrado	No mostrado	no	Wikileaks Cables Categories	2014
Namrata Gursale	Doble partición de datos	Código de Tardos	Colusión	Media y desviación estandar	Doble selección de registros a marcar	si	Automobile fields	2014
Varsha Waghmode	Modelo de restricciones, función Hash, tabla de alteraciones	Boneh & Shaw	Colusión, Eliminación, Adición	Algoritmos de clasificación (NaiveBayes,SMO,IBK,Bagging,Jrip,J48) y registro de los cambios generados	Basado en selección de atributos (Un atributo por tupla)	no	Breast tissue database (UCI)	2014
Peter Kieseberg	Anonimización de la base de daros por niveles	Anonimización	Colusión	No mostrado	No mostrado	no	No mostrado	2014
Mohanpurkar	Tres pasos de selección: partición, selección y selección por función Hash.	Código de Tardos	Colusión	Almacena las alteraciones generadas, Media y desviación estandar	Basado en umbral de selección de registros a marcar	si	Automobile database	2015
Mohanpurkar	Algoritmo PSO (Particle Swarm Optimization) con un modelo de restricciones de usabilidad	Código de Tardos	Colusión	Estadísticas de clasificación	Basado en umbral de selección de registros a marcar	si	Base de datos: Sonar, Mines vs Rocks	2015
Franco Contreras	Histograma circular	Código binario	Ninguno	No mostrado	No mostrado	si	Datos del Hospital Francés	2017
Mashal Ahmad	Creación de tuplas virtuales	Código binario	Selección de subconjunto, Adición de subconjunto, Colusión (Mix-Match).	Distorsión nula debido a la inserción de tuplas virtuales	No mostrado	si	No mostrado	2017

4

Esquema de huella digital propuesto

En este capítulo, se describen los componentes que conforman el esquema de huella digital propuesto, el análisis y el enfoque desarrollado para la reducción de la distorsión, así como el funcionamiento de los algoritmos de inserción y detección de la huella digital.

4.1. Parámetros iniciales

El esquema de huella digital propuesto considera una serie de parámetros, los cuales son esenciales para su funcionamiento. Estos parámetros son listados en la Tabla 4.1 y son descritos a continuación de acuerdo al funcionamiento que tiene cada uno de estos en el esquema.

El esquema de huella digital propuesto toma como base las técnicas de inserción de marcas de agua digitales del esquema propuesto por Li et al. (2005) y Agrawal and Kiernan (2002), las cuales determinan la selección de registros a marcar por medio de la fracción de tupla y la cual indica que una de cada y tuplas será seleccionada, así como la inclusión de una llave secreta K, que proporciona un mecanismo de seguridad y control

Tabla 4.1: Parámetros principales del esquema propuesto.

y	Fracción de tupla	K	Llave secreta del propietario
nB	Número de bits / registro	H	Función Hash SHA-256
d	Umbral de distorsión	F	Huella digital
l	Longitud de huella digital	R	Relación de la Base de Datos
v	Número de atributos de la relación		

al esquema, asociando esta llave en cada entrada de las funciones hash utilizadas por el algoritmo, de tal manera que la localización de los bits de la huella digital dependa del conocimiento de esta llave.

Además, se establecen dos parámetros principales para el funcionamiento del esquema: El número máximo de bits a insertar por cada tupla, denotado como nB, la cual es una variable que nos indicará el número de bits a insertar en un rango de $[1, nB]$, seleccionado a través de la obtención de un valor hash y la operación módulo con este parámetro. Además, un umbral de distorsión, denotado como d, es creado con el objetivo de limitar el número de bits a modificar en cada uno de los valores de cada atributo, de acuerdo a la longitud en su representación binaria. Limitando la modificación solo al primer LSB para aquellos valores donde su longitud binaria sea menor a d, y la diferencia entre la longitud y $(d - 1)$, para aquellos valores con longitud mayor a d. Estos dos parámetros permiten la variabilidad de la capacidad de inserción y la distorsión generada por la huella digital.

Esta estrategia de inserción permite reducir el error porcentual ante una modificación, como se puede observar en la tabla 4.2, donde se obtienen el porcentaje de error máximo después de una modificación, de acuerdo a la longitud de bits y el número de LSB alterados. En este caso si $d = 6$, los valores donde su longitud de bits sea menor a 6, podrán ser modificados solo en el primer LSB, ya que el alterar más de un LSB significaría obtener porcentajes de error mayores. Por ejemplo, la modificación de un valor con longitud de 6 bits, coloreado en gris en la Tabla 4.2 genera el 3.12 % de error en el primer

LSB, por lo cual su modificación utilizando el segundo LSB incrementa el error al 9.37 %, siguiendo con este comportamiento para los siguientes LSB.

Además, con este umbral se limita a aquellos valores con longitud de bits mayor a 6, a porcentajes de error por debajo de la diagonal en color amarillo, es decir cuando el valor a modificar tenga una representación binaria de 6 bits, podrá permitirse solo un error del 3.12 %, para 7 bits un error de 4.68 %, 8 bits un error de 5.46 %, y de la misma manera para las siguientes longitudes de bits.

Tabla 4.2: Error porcentual y modificación de los LSB.

		Error Porcentual					
#Bits del valor	**Valor máximo**	**1 LSB**	**2 LSB**	**3 LSB**	**4 LSB**	**5 LSB**	**6 LSB**
		Máximo	Máximo	Máximo	Máximo	Máximo	Máximo
1	1	100 %	N/A	N/A	N/A	N/A	N/A
2	3	50 %	100 %	N/A	N/A	N/A	N/A
3	7	25 %	75 %	100 %	N/A	N/A	N/A
4	15	12.5 %	37.5 %	87.5 %	100 %	N/A	N/A
5	31	6.25 %	18.7 %	43.7 %	93.7 %	100 %	N/A
6	63	3.12 %	9.37 %	21.8 %	46.8 %	96.8 %	100 %
7	127	1.56 %	4.68 %	10.9 %	23.4 %	48.4 %	98.4 %
8	255	0.76 %	2.34 %	5.46 %	11.7 %	24.2 %	49.2 %
9	511	0.39 %	1.17 %	2.73 %	5.85 %	12.1 %	24.6 %
10	1023	0.19 %	0.58 %	1.36 %	2.93 %	6.05 %	12.3 %
11	2047	0.09 %	0.29 %	0.68 %	1.46 %	3.02 %	6.15 %
12	4095	0.04 %	0.14 %	0.34 %	0.73 %	1.51 %	3.07 %

Por otra parte, otros parámetros utilizados son: la longitud de la huella digital l, el número de atributos a modificar v y la función hash H. La longitud de la huella digital denotada por l, indica el número de bits de la huella digital generada por G.Tardos, con la finalidad de seleccionar los bits de la huella a insertar en cada paso del algoritmo por medio de los índices $[1, l]$ del vector que almacena la huella digital.

También, el número de atributos de la relación a modificar v, es un parámetro importante ya que determinará la cantidad de bits disponibles para la inserción de la huella digital. Finalmente, la función hash SHA-256 denotada como H, es utilizada como un generador de números pseudoaleatorio, variando las entradas principalmente de acuerdo a la llave primaria de cada tupla PK y la llave secreta K, con el objetivo de tener

un control de localización de los bits de la huella digital tanto en el algoritmo de inserción y detección. Por lo tanto, el funcionamiento del esquema propuesto requiere de la declaración de la llave primaria de la relación a marcar.

4.2. Esquema de huella digital

El esquema de huella digital propuesto se compone principalmente del algoritmo de inserción de la huella digital y el algoritmo de detección y recuperación de la huella digital, como se muestra en el siguiente diagrama de bloques de la figura 4.1. Por su parte,

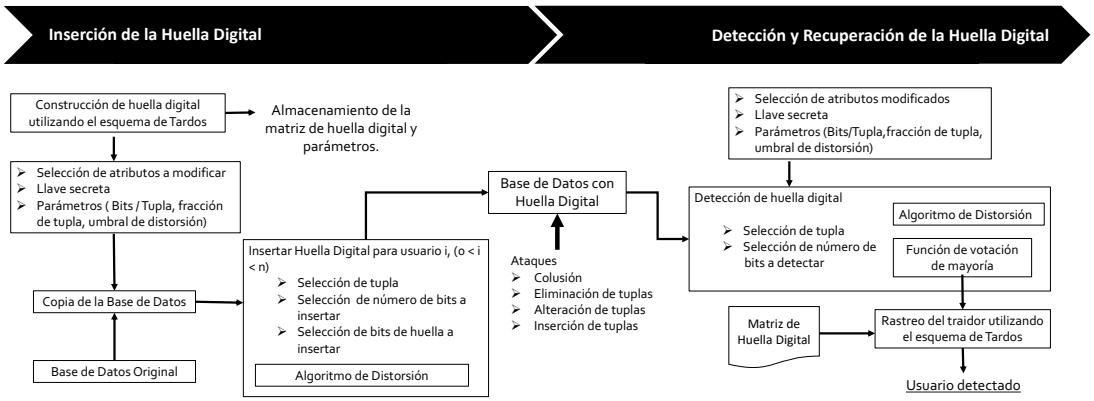


Figura 4.1: Diagrama general del esquema propuesto.

como podemos observar en la figura 4.1, el algoritmo de inserción integra la construcción y almacenamiento de las huellas digitales, ya que es el primer paso requerido para el funcionamiento del algoritmo de inserción. Posteriormente, se realiza la inicialización de parámetros requeridos y la generación de la copia de la base de datos a marcar. Finalmente, se realiza la inserción de cada una de las huellas digitales de acuerdo a cada usuario y se obtiene la copia de la base de datos marcada con una huella digital.

Una vez realizada la inserción de la huella digital, la copia de la base de datos puede sufrir alteraciones intencionales conocidas como ataques, los cuales tienen como objetivo la distorsión o eliminación de la huella digital insertada. El ataque de inserción, elimina-

ción y alteración de tuplas son abordados en este trabajo, como parte de la estrategia de inserción realizada.

En la segunda parte del diagrama, se observa el proceso de detección y recuperación de la huella digital, que consta de la inicialización de parámetros con los valores que se utilizaron durante la inserción, con el objetivo de localizar cada uno de los bits de la huella digital en el atributo y tupla donde fueron insertados. Además, la recuperación de la huella digital consta de una función de mayoría, que determinará el valor de cada bit de la huella. Finalmente, se realiza el rastreo del traidor comparando la huella digital recuperada y las huellas digitales generadas al momento de su construcción, con el objetivo de encontrar alguna coincidencia y como consecuencia determinar al usuario propietario de dicha copia de la base de datos.

En los siguientes apartados se describen a mayor detalle cada uno de los elementos que componen el esquema de huella digital propuesto.

4.2.1. Construcción de la huella digital

La generación de la huella digital es la primera etapa del esquema de huella digital, ya que determina la cadena de bits que será insertada en la copia de la base de datos a distribuir para la identificación del usuario propietario de dicha copia. Sin embargo, los esquemas de huella digital están expuestos a ataques de colusión como ya se explicó en el capítulo anterior. El esquema propuesto utiliza el esquema de Tardos (2008) como generador de huellas digitales, las cuales son creadas a partir de la inicialización de dos parámetros principales: el tamaño de colusión (c), que indica el número de usuarios soportados dentro de una colusión, y el limite de error ϵ, que indica el error permitido durante la detección de los usuarios participantes dentro de una colusión.

Estos parámetros son de mucha importancia ya que determinan la longitud de la huella digital que es insertada en la base de datos, generando huellas digitales de mayor

longitud y aportando mayor seguridad, aquellos parámetros donde el tamaño de colusión sea mayor y el limite de error menor.

En el Algoritmo 1 y Algoritmo 2, se definen los pasos para la generación de las huellas digitales, de la misma forma en que fueron descritos a través de la figura 2.2, adoptando la notación que se utilizó para la descripción del esquema propuesto.

Algoritmo 1 Generación de Huellas Digitales (Tardos)

Entrada: Número de usuarios n, Tamaño de colusión c, Límite de error ϵ

Salida: Matriz de huellas digitales M, Vector de probabilidades P

1: **procedure** GENERARCÓDIGOS(n, c, ϵ)

Inicialización de parámetros

2: $k \leftarrow ln(1/\epsilon)$

3: $l \leftarrow (100 \times c^2 k)$ ▷ Obtener longitud de huellas digitales

4: $M[n][l] \leftarrow 0$ ▷ Inicializar matriz $n \times l$ con valores cero

5: $t \leftarrow 1/(300 \times c), tp \leftarrow \sin(\sqrt{t})$

6: $rmin \leftarrow tp, rmax \leftarrow (\Pi/2.0 - tp)$

7: $p[l] \leftarrow 0$

8: $r \leftarrow 0.0$

9: **para** $i \leftarrow 0\ hasta\ i < l$ **hacer** ▷ Obtener vector de probabilidades P

10: $r \leftarrow rmin + Rand() \times (rmax - rmin)$

11: $P[i] \leftarrow (\sin r)^2$

12: **fin para**

13: *//Creación de la Matriz de Códigos*

14: **para** $i \leftarrow 0\ hasta\ i < n$ **hacer**

15: $M[i] \leftarrow ObtenerCodigo(P)$ ▷ Obtener huella digital para el usuario i

16: **fin para**

17: **devolver** M, P ▷ Matriz de huellas digitales y vector P

18: **fin procedure**

La utilización del esquema de G.Tardos, permite la creación de huellas digitales de acuerdo a las necesidades que se vayan presentando durante su aplicación, es decir no se requiere definir el número de usuarios a priori, ya que la generación de huellas digitales depende del vector de probabilidades como se puede ver en el Algoritmo 2.

Algoritmo 2 Obtener Huella Digital

Entrada: Vector de probabilidades P

Salida: Huella Digital f

1: **procedure** OBTENERCÓDIGO(P)
2: $f[l] \leftarrow 0$ ▷ Vector para almacenar la huella digital
3: **para** $i \leftarrow 0\ hasta\ i < longitud(P)$ **hacer**
4: $f[i] = Rand() < P[i]$ ▷ Generar número pseudoaleatorio
5: **fin para**
6: **devolver** f ▷ Huella digital
7: **fin procedure**

Sin embargo, a pesar de las características que brinda el esquema de G.Tardos, su utilización involucra contar con una gran capacidad de inserción, lo cual se traduce en mayor distorsión de los datos de la relación a marcar. Además, es necesario almacenar la matriz de huellas digitales y el vector de probabilidades, ya que son utilizados durante el proceso de rastreo del traidor. Por lo tanto, su almacenamiento representa una desventaja de seguridad para el esquema propuesto.

4.2.2. Algoritmo de inserción

El algoritmo de inserción consiste en la asignación de una huella digital F diferente, a cada una de las copias de la relación de la base de datos R que serán distribuidas. El diagrama de flujo se muestra en la figura 4.2, además, el Algoritmo 3 describe cada paso del proceso de inserción.

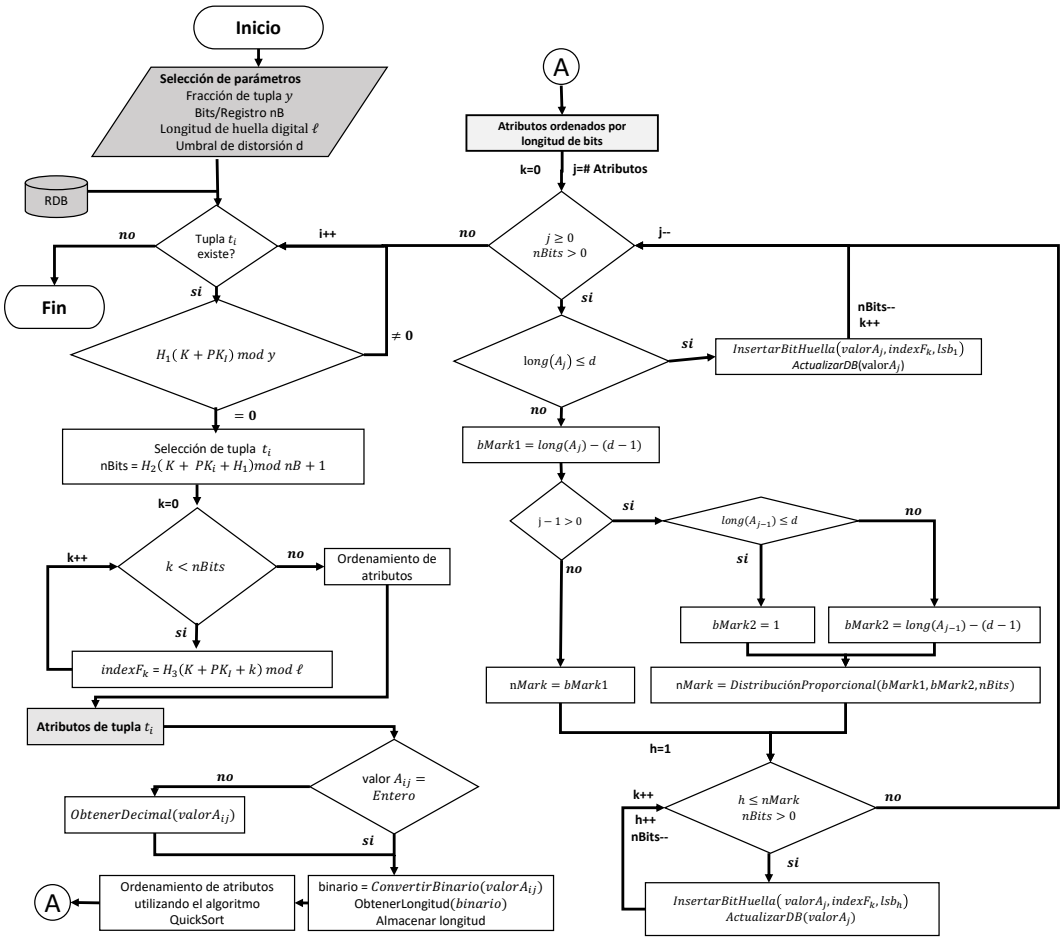


Figura 4.2: Diagrama del algoritmo de inserción.

El primer paso es la selección de los parámetros principales: fracción de tupla y, número de bits máximo a insertar por tupla nB, número de atributos a modificar v, la longitud de huella digital l, el umbral de distorsión d y la llave secreta K definida por el

propietario de la base de datos. Estos parámetros deberán ser fijos para cada una de las copias a distribuir, con el objetivo de insertar todas las huellas digitales en las mismas posiciones y de esta manera hacer efectiva la estrategia de G.Tardos respecto al ataque de colusión.

Por otro lado, la selección de los atributos a modificar es realizada por el propietario de la base de datos, considerando aquellos atributos que permitan la introducción de errores a sus valores. Los atributos considerados en el esquema propuesto son de tipo numérico, debido a que las modificaciones a los LSB no cambian la estructura de este tipo de dato.

Para cada una de las copias a distribuir el procedimiento es el siguiente. Cada tupla t_i de la relación R de la base de datos, es seleccionada para marcar si el valor hash, tomando como entrada la concatenación de la llave secreta K y la llave primaria del registro PK_i, y obteniendo el resultado de la operación módulo con la fracción de tupla y es cero, $H_1\left(K + PK_i\right) mod\, y = 0$.

Una vez seleccionada la tupla a marcar, se selecciona el número de bits de la huella digital que será insertado en la tupla $nBits$, obteniendo el valor hash de la entrada derivada de la concatenación de la llave secreta K, la llave primaria de la tupla PK_i y el valor hash H_1, con el resultado de la operación módulo con el número máximo de bits por tupla nB, asignado por el propietario, $H_2\left(K + PK_i + H_1\right) mod\, (nB + 1)$. La variación en la entrada de la función hash se realizó con el objetivo de tener mayor aleatoriedad al seleccionar esta variable. También, cabe aclarar que de esta forma el número de bits a insertar en cada tupla $nBits$ varía de una tupla a otra, diferenciándose del parámetro nB, el cual marca el límite de bits permitidos para insertar en una tupla. Esta variación se realiza con el objetivo de agregar mayor aleatoriedad al esquema.

Posteriormente, se obtienen los índices de los bits de la huella digital que serán insertados en la tupla, de acuerdo a la cantidad de bits a insertar almacenada en la variable $nBits$ previamente seleccionada. Para esto, cada uno de los índices de la huella digital a

insertar se seleccionan por medio de la operación $H_3\,(K + PK_i + k)\,mod\,l$, donde k es una variable que controla el número de índice obtenido.

Algoritmo 3 Inserción de la Huella Digital

Entrada: Relación de la DB R, Llave secreta K, Huella digital F

Salida: Relación de la DB marcada R'

1: **procedure** INSERCIÓN(R, K, F)

// Inicialización de parámetros

2: y, nB ▷ Fracción de tupla, número máximo de bits/tupla

3: l, d ▷ Longitud de la huella digital,umbral de distorsión

4: **para todo** $t_i \in R$ **hacer** ▷ Cada tupla de R

5: **si** $H_1(K + PK_i) mod\ y = 0$ **entonces** ▷ Selección de la tupla

6: $nBits \leftarrow H_2(K + PK_i + H_1) mod\ (nB + 1)$ ▷ Obtener número de bits a insertar

7: **para** $k \leftarrow 0\ hasta\ k < nBits$ **hacer**

8: $indexF_k \leftarrow H_3(K + PK_i + k) mod\ l$ ▷ Obtener índices de los bits de la huella digital a insertar

9: **fin para**

10: **para todo** $A_{ij} \epsilon\ t_i$ **hacer** ▷ Cada atributo de la tupla

11: **si** $A_{ij} \notin \mathbb{Z}$ **entonces** ▷ Verificar si A_{ij} es un número entero

12: *//Seleccionar la parte decimal del valor* A_{ij}

13: **fin si**

14: $binario \leftarrow ConvertiraBinario(A_{ij})$

15: $ObtenerLongitud(binario)$

16: *//Almacenar longitud*

17: **fin para**

18: *//Ordenar atributos de* t_i *por longitud utilizando el algoritmo QuickSort*

19: $j \leftarrow$ Número de atributos, $k \leftarrow 0$

Algoritmo 3 (continua) Inserción de la Huella Digital (Reducción de la distorsión)

```
20:             mientras nBits > 0 & j ≥ 0 hacer
21:                 si long(A_ij > d) entonces                                   ▷
22:                     //Insertar F_k en el primer LSB del valor A_ij
23:                     InsertarBitHuella(A_ij, indexF_k, LSB_1)
24:                     ActualizarDB(A_ij)                  ▷ Actualizar nuevo valor de A_ij
25:                     k++, nBits--, j--                   ▷ Actualizar variables de control
26:                 si no
27:                     bMark1 ← long(A_ij − (d − 1)   ▷ Bits disponibles para modificar en A_ij
28:                     si j − 1 > 0 entonces               ▷ Verificar si existe otro atributo
29:                         si long(A_ij−1) ≤ d entonces
30:                             bMark2 ← 1
31:                         si no
32:                             bMark2 ← long(A_ij−1) − (d − 1)
33:                         fin si
34:                         nMark ← DistribuciónProporcional(bMark1, bMark2, nBits)
35:                     si no
36:                         nMark ← bMark1
37:                     fin si
38:                     h ← 1
39:                     mientras h ≤ nMark & nBits > 0 hacer
40:                         InsertarBitHuella(A_ij, indexF_k, LSB_h)
41:                         ActualizarDB(A_ij)              ▷ Actualizar nuevo valor de A_ij
42:                         k++, h++, nBits--               ▷ Actualizar variables de control
43:                     fin mientras
44:                 fin si
45:             fin mientras
46:             i++                                         ▷ Continuar con la siguiente tupla
47:         fin si
48:     fin para
49:     devolver R'                         ▷ Relación de la Base de Datos con la huella digital
50: fin procedure
```

Una vez realizada la selección de estas variables, el algoritmo selecciona los atributos y bits menos significativos a modificar de acuerdo al enfoque de minimización de la distorsión. Los detalles sobre este procedimiento son presentados a continuación.

Reducción de la distorsión

Una parte importante del algoritmo es la reducción de la distorsión durante el proceso de inserción de la huella, mostrado en la segunda parte del Algoritmo 3. El enfoque propuesto consiste en la selección de los valores de los atributos con mayor longitud en su representación binaria, con el objetivo de reducir el error relativo al modificar estos valores, a diferencia de estrategias del estado del arte donde esta característica no es considerada. Por otro lado, una de las ventajas de esta estrategia es el incremento de la capacidad de inserción, debido a que más de 1 bit puede ser insertado dentro de un valor.

Para esto, el algoritmo almacena la longitud de la representación binaria de cada uno de los valores de los atributos de la tupla seleccionada, como puede verse en el diagrama de la figura 4.3. Para los valores no enteros, se obtiene la parte decimal del valor con el objetivo de que la modificación sea realizada en estos dígitos. Posteriormente, estas longitudes se ordenan utilizando el algoritmo de ordenamiento QuickSort, para mayor eficiencia en el proceso.

En la segunda parte del Algoritmo 3, el atributo de mayor longitud es modificado en el primer LSB, si su longitud es menor o igual al umbral de distorsión $long(A_{ij}) \leqslant d$. En este caso la inserción es realizada llevando el valor del atributo a su representación binaria y sustituyendo el valor del primer LSB con el bit de la huella digital del primer índice obtenido en el paso anterior. En caso contrario, el algoritmo calcula el número de bits permitidos para modificación mediante $bMark1 = long(A_{ij}) - (d - 1)$. Una vez realizado este paso, se verifica si existe un siguiente atributo de menor o igual longitud A_{ij-1} y en caso de existir se obtiene el número de bits disponibles para modificar, siendo $bMark2 = 1$ si $long(A_{ij-1}) \leqslant d$, y $bMark2 = long(A_{ij-1}) - (d - 1)$ en caso contrario.

Atributos	A1	A2	A3	A4	A5	A6
Valores originales	31	63	127	255	511	1023
Longitud de bits	5	6	7	8	9	10

Umbral de distorsión
d = 6

nBits = 7

	bMark2	bMark1
#Bits disponibles	4	5
#Bits a insertar		**4**

1

nBits = 3

	bMark2	bMark1
#Bits disponibles	3	4
#Bits a insertar		**2**

2

nBits = 1

	bMark2	bMark1
#Bits disponibles	2	3
#Bits a insertar		**1**

3

Inserción de bits de la huella digital y valores actualizados.

Atributos	A1	A2	A3	A4	A5	A6
Nuevos Valores	31	63	127	254	508	1008
#Bits insertados	0	0	0	1	2	4

Figura 4.3: Inserción de bits utilizando distribución proporcional.

Teniendo el número de bits disponibles del valor uno $bMark1$ y el número de bits disponibles del siguiente valor $bMark2$, se realiza la distribución proporcional de acuerdo al número de bits a insertar $(nBits)$ si $nBits \leqslant (bMark1 + bMark2)$. En caso contrario la distribución proporcional se realiza de acuerdo al total de la suma $(bMark1 + bMar2)$, con lo cual conseguimos obtener el número de bits que serán insertados en el primer valor, como se indica en la ecuación 4.1.

$$nMark = \begin{cases} \frac{bMark1 * nBits}{bMark1 + bMar2} & \text{sí } nBits \leqslant (bMark1 + bMark2) \\ \\ \frac{bMark1(bMark1 + bMark2)}{bMark1 + bMark2} & \text{sí } nBits > (bMark1 + bMark2) \end{cases} \tag{4.1}$$

Sin embargo, si no existe algún otro valor para obtener $bMark2$, $nMark = bMark1$ de forma directa. Una vez conocido el número de bits a insertar en el valor analizado en ese momento, se itera insertando el bit del índice correspondiente y el LSB correspondiente de acuerdo a cada iteración.

Finalmente, el valor es actualizado en la relación de la base de datos y se analiza el valor del siguiente atributo con menor longitud, para la inserción de los $nBits$ restantes. En caso de no haber más atributos o $nBits$ a insertar, el algoritmo continua analizando la siguiente tupla. Este proceso es ejemplificado en la figura 4.3, donde se realiza la inserción de $nBits = 7$ bits de la huella digital, con un umbral de distorsión $d = 6$. El paso 1,2 y 3 muestran la distribución proporcional de los bits restantes a insertar en cada paso. Finalmente la actualización de los valores para ese registro es mostrado en la parte inferior de la figura 4.3.

Por otro lado, uno de los aspectos a considerar en la reducción de la distorsión, es la correspondencia que puede existir entre el valor del bit de la huella a insertar y el bit del valor a modificar, es decir el caso en el que ambos valores sean iguales, lo cual no altera el valor original y por lo tanto la distorsión es nula.

4.2.3. Algoritmo de detección y recuperación de huella digital

El algoritmo de detección (Algoritmo 4), mantiene la misma estructura del algoritmo de inserción en cuanto a la selección de tupla, número de bits a insertar y selección de índices de bits de huella digital. Por lo cual, la selección de los parámetros iniciales debe ser la misma utilizada en el algoritmo de inserción, así como la llave secreta K definida por el propietario de la base de datos, ya que la modificación de estos parámetros altera los resultados de las operaciones realizadas en cada paso del algoritmo, y por lo tanto la localización de los bits de la huella digital sería errónea.

Este algoritmo considera una matriz $cont$ de dimensiones $l \times 2$, ya que debe corresponder con la longitud de las huellas digitales insertadas previamente. Esta matriz es utilizado para el almacenamiento del número de bits con valor uno y cero, los cuales son detectados durante la extracción de la huella digital.

Algoritmo 4 Detección de la Huella Digital

Entrada: Relación de la DB marcada R', Llave secreta utilizada en la inserción K

Salida: Bits de la huella digital recuperados

1: **procedure** DETECCIÓN(R', K)

// Inicialización de parámetros de acuerdo al proceso de inserción

2: y, nB ▷ Fracción de tupla, número máximo de bits/tupla

3: l, d ▷ Longitud de la huella digital,umbral de distorsión

4: $cont[l][2] \leftarrow 0$ ▷ Matriz para almacenar bits de la huella digital

5: **para todo** $t_i \in R$ **hacer** ▷ Cada tupla de R'

6: **si** $H_1(K + PK_i) mod\ y = 0$ **entonces** ▷ Selección de la tupla

7: $nBits \leftarrow H_2(K + PK_i + H_1) mod\ (nB + 1)$ ▷ Obtener número de bits insertados

8: **para** $k \leftarrow 0\ hasta\ k < nBits$ **hacer**

9: $indexF_k \leftarrow H_3(K + PK_i + k) mod\ l$ ▷ Obtener índices de los bits de la huella digital insertados

10: **fin para**

11: **para todo** $A_{ij} \in t_i$ **hacer** ▷ Cada atributo de la tupla

12: **si** $A_{ij} \notin \mathbb{Z}$ **entonces** ▷ Verificar si A_{ij} es un número entero

13: *//Seleccionar la parte decimal del valor* A_{ij}

14: **fin si**

15: $binario \leftarrow ConvertiraBinario(A_{ij})$

16: $ObtenerLongitud(binario)$

17: *//Almacenar longitud*

18: **fin para**

19: *//Ordenar atributos de* t_i *por longitud utilizando el algoritmo QuickSort*

20: $j \leftarrow$ Número de atributos, $k \leftarrow 0$

Algoritmo 4 (continua) Detección de la Huella Digital (Reducción de la distorsión)

```
21:             mientras nBits > 0 & j ≥ 0 hacer
22:                 si long(A_ij > d) entonces
23:                     //Extraer bit en el primer LSB del valor A_ij
24:                     bitHuella ← ObtenerBitHuella(A_ij, LSB_1)
25:                     cont[i][bitHuella] + +                    ▷ Contabilizar bit encontrado
26:                     k + +, nBits − −, j − −                   ▷ Actualizar variables de control
27:                 si no
28:                     bMark1 ← long(A_ij − (d − 1)              ▷ Bits disponibles en A_ij
29:                     si j − 1 > 0 entonces                     ▷ Verificar si existe otro atributo
30:                         si long(A_ij−1) ≤ d entonces
31:                             bMark2 ← 1
32:                         si no
33:                             bMark2 ← long(A_ij−1) − (d − 1)
34:                         fin si
35:                         nMark ← DistribuciónProporcional(bMark1, bMark2, nBits)
36:                     si no
37:                         nMark ← bMark1
38:                     fin si
39:                     h ← 1
40:                     mientras h ≤ nMark & nBits > 0 hacer
41:                         //Extraer bit en los h primeros LSB del valor A_ij
42:                         bitHuella ← ObtenerBitHuella(A_ij, LSB_h)
43:                         cont[i][bitHuella] + +                ▷ Contabilizar bit encontrado
44:                         k + +, h + +, nBits − −               ▷ Actualizar variables de control
45:                     fin mientras
46:                 fin si
47:             fin mientras
48:             i + +                                             ▷ Continuar con la siguiente tupla
49:         fin si
50:     fin para
51:     devolver cont                                             ▷ Contador de bits de la huella digital recuperados
52: fin procedure
```

Definidos los parámetros iniciales y la matriz de almacenamiento para la huella digital, el algoritmo realiza las operaciones descritas en el algoritmo de inserción, exceptuando la inserción de los bits de la huella digital, ya que en este caso se extraen los bits de la huella previamente insertada. Esto se realiza obteniendo los bits de los valores ubicados en los LSB que fueron modificados, y almacenándolos en los índices de la matriz, correspondientes a los índices de la huella digital que fueron seleccionados al inicio del algoritmo.

Posteriormente, habiendo analizado el total de tuplas de la relación de la base de datos R', la matriz de almacenamiento de la huella digital extraída es procesada mediante la función mayoría, lo cual es explicado en la siguiente sub sección.

Función mayoría

El algoritmo de recuperación de la huella digital es presentado en el Algoritmo 5, el cual recibe como entrada la matriz de almacenamiento de bits recuperados $cont$ y obtiene la huella digital con los valores de mayor presencia, derivados de la función de mayoría. La función mayoría consiste en la selección de los bits de la huella digital presentes con mayor frecuencia durante la detección. Esto se realiza a través de la contabilización de los valores uno $(cont[i][1])$ y cero $(cont[i][0])$ recuperados para cada índice de la huella digital, y la selección del bit mayoritario de acuerdo a la ecuación 4.2.

$$F_i = \begin{cases} 0 & \text{sí } cont[i][0]/(cont[i][0] + cont[i][1]) > 0.5 \\ 1 & \text{sí } cont[i][1]/(cont[i][0] + cont[i][1]) > 0.5 \end{cases} \tag{4.2}$$

Algoritmo 5 Recuperación de la Huella Digital

Entrada: Contador de bits de la huella digital recuperados $cont$, longitud de huella digital l

Salida: Huella digital recuperada

```
1:  procedure RECUPERACIÓN(cont, l)
2:      F' ← 0                                          ▷ Vector de longitud l
3:      T ← 0.5                                         ▷ Umbral de selección de bit
4:      para i ← 0 hasta i ≤ l hacer
5:          si cont[i][0] + cont[i][1] ≠ 0 entonces
6:              si cont[i][0]/(cont[i][0] + cont[i][1]) > T entonces
7:                  F'[i] ← 0                           ▷ Asignar valor del bit a 0
8:              fin si
9:              si cont[i][1]/(cont[i][0] + cont[i][1]) > T entonces
10:                 F'[i] ← 1                           ▷ Asignar valor del bit a 1
11:             fin si
12:         fin si
13:     fin paradevolver F'                             ▷ Huella digital recuperada
14: fin procedure
```

Una vez obtenida la huella digital con los bits definitivos, almacenados en el arreglo F', el siguiente paso es la identificación del usuario (traidor) correspondiente a la copia analizada en ese momento, lo cual es descrito en la siguiente sub sección.

4.3. Rastreo del traidor

La identificación de usuarios propietarios de cada una de las copias distribuidas de la base de datos fue realizada utilizando el esquema de Tardos (2008), Algoritmo 6. Esto se realizó a través de la comparación de cada bit de la huella digital F' obtenida en el paso anterior y su correspondiente bit de cada una de las huellas digitales generadas para todos los usuarios en la etapa de construcción de las huellas digitales, además de la ayuda del vector de probabilidades generado en esa misma etapa.

Posteriormente, realizada esta comparación, el algoritmo determina a qué usuario le corresponde la copia analizada, o en su caso los usuarios traidores si la copia es resultado de un ataque de colusión. Esto de acuerdo al número de huella digital que superé el umbral Z de Tardos durante la comparación descrita anteriormente. Este proceso es explicado detalladamente en la figura 2.2 del Capítulo 2.

Cabe mencionar que el proceso de rastreo del traidor está relacionado con la cantidad de bits de la huella digital recuperados, ya que el algoritmo depende de esta cantidad para determinar si el umbral de acusación de Tardos es alcanzado. Por lo tanto, la resistencia del esquema ante ataques de inserción, alteración y eliminación de tuplas, es muy importante, ya que permite mejorar la recuperación de los bits de la huella digital.

Algoritmo 6 Rastreo del Traidor (Tardos)

Entrada: Huella digital recuperada F', Matriz de Huellas Digitales M, Vector de probabilidades P, Tamaño de colusión c, Límite de error ϵ, Número de usuarios n, Longitud de huella digital l

Salida: Usuario(s) culpable de distribución ilegal U

1: **procedure** RASTREO(F', M, P, c, ϵ)
2: $k \leftarrow ln(1/\epsilon)$
3: $Z \leftarrow 20 \times c \times k$ ▷ Umbral definido por Tardos
4: $s \leftarrow 0$ ▷ Almacenar sumatoria
5: **para** $i \leftarrow 0\ hasta\ i \leq n$ **hacer**
6: **para** $j \leftarrow 0\ hasta\ j \leq l$ **hacer**
7: **si** $(M[i][j]\ \ \&\ \ F'[j]) = verdadero$ **entonces**
8: $s \leftarrow s + \sqrt{(1 - P_j)/P_j}$
9: **fin si**
10: **si** $(!M[i][j]\ \ \&\ \ F'[j]) = verdadero$ **entonces**
11: $s \leftarrow s - \sqrt{(P_j/(1 - P_j))}$
12: **fin si**
13: **fin para**
14: **si** $s > Z$ **entonces**
15: *Almacenar usuario culpable i en U*
16: **fin si**
17: $s \leftarrow 0$
18: **fin para**
19: **devolver** U ▷ Usuario(s) culpable de distribución ilegal
20: **fin procedure**

4.4. Discusión

En este capítulo se describió y analizó el funcionamiento de cada uno de los componentes del esquema de huella digital propuesto. Sin embargo, es necesario mencionar los aspectos más relevantes tanto positivos y negativos del desarrollo del esquema. Dentro de los aspectos positivos se encuentra el aumento en la capacidad de inserción, siendo este uno de los objetivos de esta investigación, ya que debido a esto logramos que la utilización de huellas digitales de mayor tamaño sean viables. Además, la reducción de la distorsión como observamos anteriormente, restringe el número de bits menos significativos a modificar de acuerdo a la longitud de bits de los valores evitando de esta manera la distorsión de estos valores significativamente.

Por otra parte, la inserción de la huella digital se realiza de acuerdo a los parámetros (y, nB, d) definidos por el propietario de la base de datos, por lo cual, el propietario determina el límite de distorsión permitido para la base de datos, de acuerdo a los propósitos con la que fue creada. Otro aspecto a considerar en la reducción de la distorsión, es que el desordenamiento o eliminación de algún atributo de la base de datos comprometería la recuperación de la huella digital, ya que este proceso depende del ordenamiento por longitud de bits de los atributos antes de la inserción, por lo tanto, es un aspecto con el que se deberá trabajar más adelante.

En el capítulo siguiente se mostrará el rendimiento obtenido de la evaluación realizada centrada en la capacidad de inserción, la distorsión generada y la robustez a ataques a la relación de la base de datos marcada.

5

Experimentos y resultados

En este capítulo se describen los experimentos realizados para la evaluación del rendimiento del esquema de huella digital propuesto, bajo tres requerimientos principales, usabilidad, capacidad y robustez. Además, se realiza la discusión de los resultados obtenidos y su comparación con los esquemas del estado del arte.

5.1. Entorno de desarrollo del esquema

En esta sección se describen las herramientas necesarias para el desarrollo y ejecución del esquema propuesto, así como la implementación de experimentos realizados.

Entorno de desarrollo y gestor de base de datos

Los experimentos realizados para la evaluación del esquema de huella digital propuesto, fueron desarrollados utilizando el entorno de desarrollo NetBeans 8.2, ejecutado en un equipo Mac OS Sierra 10.12 con un procesador Intel Core i5 a 1.6 GHz, además del JDK (Kit de desarrollo de Java) 12.3. En cuanto al Sistema Gestor de Bases de Da-

tos (SGBD, por sus siglas en inglés), se utilizó MySQL, debido a contar con una licencia pública GNU y a su flexibilidad al vincularlo con entornos de desarrollo.

Base de datos

Actualmente no se cuenta con un Benchmark para la evaluación de esquemas de marcas de agua digital en bases de datos relacionales, por lo tanto, para estos experimentos se empleó la base de datos Forest CoverType de UCI KDD, la cual es una de las bases de datos más utilizadas en los artículos referentes dentro del área de marcas de agua y huella digital.

Esta base de datos cuenta con 581012 registros y 61 atributos numéricos, de los cuales los primeros 10 atributos son utilizados, debido a que el dominio de estos atributos es más amplio que el resto de los atributos, donde solo dos posibles valores pueden ser asignados, lo cual podría dificultar visualizar el comportamiento del esquema propuesto. Por otra parte, la base de datos fue segmentada de acuerdo a los experimentos realizados, con el objetivo de observar el comportamiento del esquema en diferentes tamaños de la base de datos. Además, una llave primaria "id_key" fue insertada, debido a que la base de datos no cuenta con este atributo, el cual es necesario para el funcionamiento del esquema de huella digital propuesto.

Huella digital

Por su parte, las huellas digitales utilizadas durante el desarrollo de los experimentos fueron obtenidas de la ejecución del esquema de (Tardos, 2008), con un tamaño de colusión c = 2, y una tasa de error de $\epsilon = 0.01$, generando un total de 10 huellas de longitud de 1842 bits, para la ejemplificación del comportamiento del esquema.

Este tamaño de colusión supone una coalición de a lo sumo dos usuarios puede presentarse al realizarse un ataque de colusión, con la ventaja de determinar con un error

de 0.01 % a los usuarios involucrados. Por lo cual, estos parámetros representan la menor longitud de huella digital con el mayor nivel de seguridad posible en la identificación de usuarios. Debido a que el incremento en el tamaño de colusión c, significa un aumento en la longitud de la huella digital y por el contrario una mayor tasa de error reduce la longitud volviendo la huella digital vulnerable a este tipo de ataque.

Sin embargo, la selección de estos parámetros depende de la capacidad de inserción del esquema, ya que si la capacidad de inserción es menor a la cantidad de bits de la huella digital, la relación de la base de datos no podrá ser marcada, ya que la inserción de la huella digital no podrá ser completada en su totalidad, impidiendo la identificación del usuario de esta relación y volviendo vulnerable el esquema ante ataques a la base de datos.

5.2. Evaluación de usabilidad

Este experimento se realizó con el propósito de medir el grado de distorsión causado a la base de datos posterior a la inserción de la huella digital. Esto se realizó utilizando como métrica el Error Cuadrático Medio (MSE), buscando obtener la distorsión general de la base de datos y por otro lado, el Error Relativo (RE), para medir el impacto de distorsión sobre los valores modificados.

Estos experimentos se realizaron tomando en cuenta la escalabilidad de la base de datos, para lo cual se tomaron fragmentos de la base de datos Forest CoverType, divididos con el siguiente número de registros en cada uno de ellos: 5000,10000,50000 y 100000 registros.

En la primera etapa de estos experimentos, se analizó la distorsión causada en términos de MSE variando los parámetros que indican la selección de registros (y) y el número máximo de bits por tupla (nB). Para efectos de comparación frente a otros esquemas de huella digital del estado del arte, la inserción de la huella digital se realizó

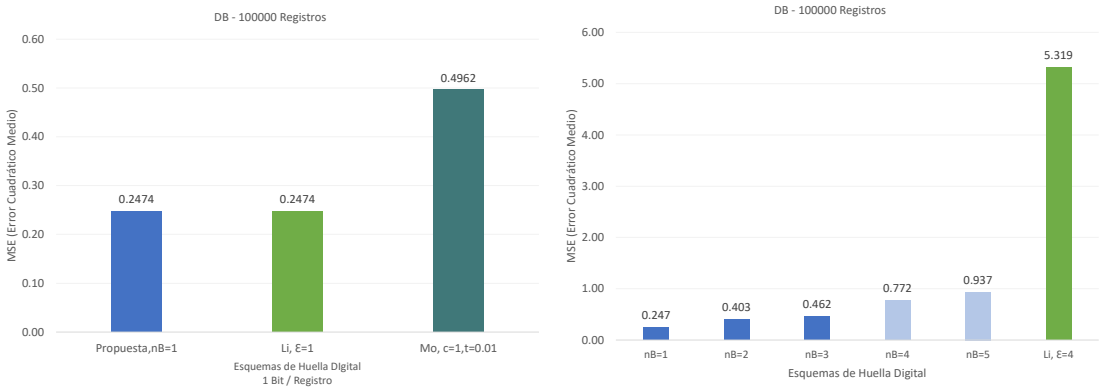
utilizando el trabajo propuesto por Mohanpurkar and Joshi (2015c), debido a su estrategia de inserción, para posteriormente tomar los registros seleccionados para marcar, y realizar la inserción tanto con el esquema de huella digital propuesto por Li et al. (2005), así como con el esquema propuesto en este trabajo.

La selección de estos trabajos es debido a que la estrategia de inserción propuesta en los dos esquemas es distinta y puede aportar mayor información del rendimiento de los resultados obtenidos con el esquema propuesto. Además, el trabajo de Li et al. (2005) ha sido fundamental para el desarrollo de algunos esquemas del estado del arte, así como del esquema propuesto, ya que la estrategia de control de inserción de bits que propone mediante la utilización de números pseudoaleatorios ha sido el principal componente dentro del área, además de su robustez a ataques a la RDB, lo cual no se presenta en la mayoría de los trabajos del estado del arte, ya que en ciertos trabajos es necesario almacenar las modificaciones realizadas para la recuperación de la huella digital, por lo cual cualquier manipulación a los valores de la base de datos marcada impide la recuperación.

Por otro lado, Mohanpurkar and Joshi (2015c) cambia la forma de selección de tuplas y la inserción del bit de la huella digital, mostrando distinta capacidad y distorsión, por lo cual su comparación resulta de interés. Además de que este esquema es tolerante al ataque de colusión, por lo que los resultados de distorsión y capacidad puede ser comparables con los resultados del trabajo propuesto. Por otra parte, otro factor que intervino para la selección de estos esquemas fue el acceso a la documentación necesaria para su reproducción.

Primeramente, se evaluó la distorsión del esquema propuesto frente a los trabajos antes mencionados, insertando 1 bit de la huella digital por cada registro, esto con el objetivo de observar la distorsión causada con los valores mínimos en cada esquema. Como se observa en la figura 5.1a, si bien la distorsión de Li con $\varepsilon = 1$ es similar a la obtenida con el esquema propuesto, el resultado es diferente cuando debido a su estrategia, al au-

mentar el número de LSB a modificar, la distorsión se incrementa de forma considerable, como puede verse en la gráfica de la figura 5.1b con $\varepsilon = 4$, donde se compara con el esquema propuesto variando el número de bits insertados por tupla $nB = [1, 5]$, bajo los registros seleccionados previamente. Sin embargo, la comparación con Mohanpurkar permite la inserción de un mayor número de bits por tupla $nB = [1 - 3]$ manteniendo la distorsión por debajo de la obtenida por este esquema, como se ve en la gráfica de la figura 5.2a, lo cual presenta ventajas mostradas a continuación.



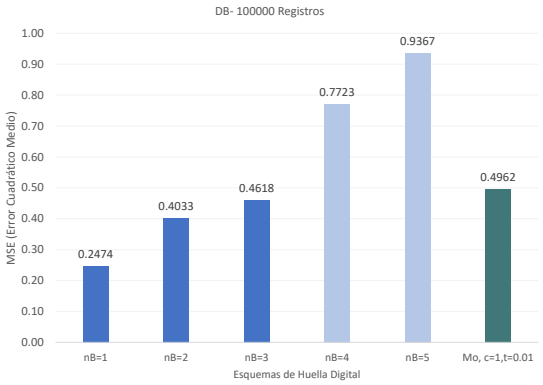
(a) 1 Bit/Registro (b) Comparación con Li cuando ε=4

Figura 5.1: Comparación de la distorsión en términos de MSE.

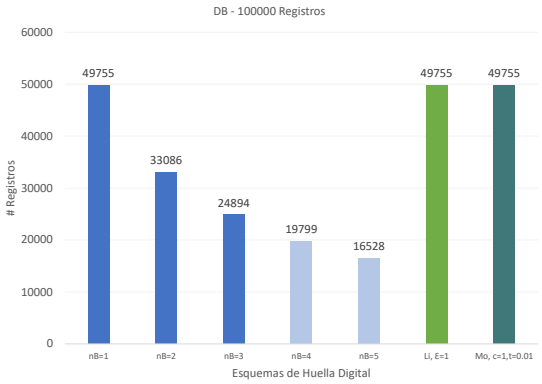
Una de estas ventajas se presenta debido a que la variación de nB, tiene el efecto de reducción de tuplas a marcar, lo cual ayuda a la conservación de un mayor número de tuplas con los valores reales de la base de datos, como se puede ver en la gráfica de la figura 5.2b en comparación con Li y Mohanpurkar.

Un resumen de los datos obtenidos de la comparación de los esquemas, se presenta en la tabla 5.1, en la cual se aprecia también la reducción del tiempo de inserción de la huella digital en un 32 % por parte del esquema propuesto, cuando $nB = 3$ generando un grado de distorsión por debajo de lo obtenido por Mohanpurkar.

Una de las ventajas del proceso de inserción que es importante mencionar y que ayuda a reducir la distorsión, es que la inserción de los bits de la huella digital no siempre



(a) MSE respecto a Mohanpurkar



(b) Reducción de registros

Figura 5.2: Efecto de reducción de registros a marcar.

produce un cambio en los valores originales, esto debido a que se pueden encontrar bits de la huella digital y el valor a modificar, con el mismo valor binario. Por ejemplo, si el bit de la huella a insertar es de 1, y el valor a modificar es 7, con representación binaria '111', la inserción en cualquiera de los bits menos significativos no produciría ninguna distorsión.

Tabla 5.1: Reducción de distorsión y tiempo de inserción.

Esquemas	MSE	#Bits	#Tuplas	Tiempo
Propuesta, nB=1	0.2474	49755	49755	2.5246'
Propuesta, nB=2	0.4033	49755	33086	1.7533'
Propuesta, nB=3	0.4617	49755	24894	1.7105'
Propuesta, nB=4	0.7723	49755	19799	1.6103'
Propuesta, nB=5	0.9367	49755	16528	1.3664'
Li, ε=1	0.2474	49755	49755	2.5195'
Li, ε=4	5.3189	49755	49755	2.5195'
Li, ε=8	673.48	49755	49755	2.5195'
Mo, c=1, t=0.01	0.4962	49755	49755	2.5683'

Además de estos experimentos, con el objetivo de reafirmar los resultados obtenidos, se realizó la inserción de la huella digital de acuerdo a la distorsión obtenida por Mohanpurkar. Por ello se obtuvo la distorsión resultante de la inserción con el esquema de huella digital de Mohanpurkar y se midió la capacidad de inserción del esquema propuesto, estableciendo como límite de inserción esta distorsión. Los resultados obtenidos son mostrados en la gráfica de la figura 5.3, en la cual se puede ver la mejora de la capacidad de inserción cuando $nB = 1, 2$ y 3, resaltados en color azul.

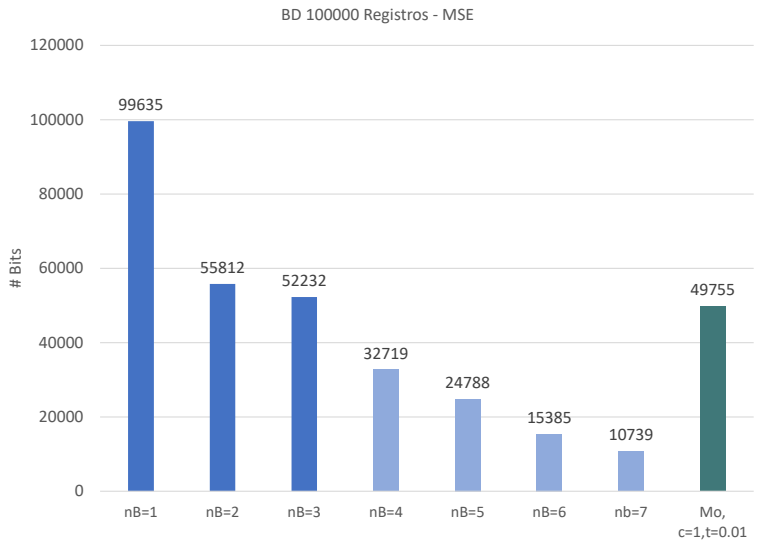


Figura 5.3: Capacidad de inserción con limite de distorsión.

Por otra parte, como se mencionó en el capítulo anterior, uno de los aspectos de mayor relevancia en el esquema propuesto fue la distorsión generada respecto al valor modificado, en términos del error relativo. Para esto se realizó el mismo experimento de comparación descrito anteriormente, midiendo el error relativo de cada atributo de la base de datos y realizando la sumatoría total.

En la tabla 5.2 se observa el error relativo en cada uno de los atributos de la base de datos, teniendo un efecto negativo en los esquemas tanto de Li y Mohanpurkar, debido a que la aleatoriedad de estos esquemas tiende a seleccionar valores de menor magnitud, por lo cual su modificación tiende a representar un mayor porcentaje de distorsión

Tabla 5.2: Comparación del error relativo en la base de datos Forest Cover Type.

	Esquema Propuesto					**Li**			**Mohanpurkar**
#Bits	**49755**	**49755**	**49755**	**49755**	**49755**	**49755**			**49755**
#Registros	**49755**	**33086**	**24894**	**19799**	**16528**	**49755**			**49755**
	1 Bit	**2 Bit**	**3 Bit**	**4 Bit**	**5 Bit**	**1 LSB**	**4 LSB**	**8 LSB**	**c=1, t=0.01**
Elevation	2.332	2.557	2.852	3.630	4.494	0.873	3.284	27.059	1.691
Aspect	0	0	0.013	0.021	0.035	95.896	235.832	902.726	199.225
Slope	0	0	0	0	0	350.779	1214.216	2313.675	586.276
HD-Hydrology	0	0	0.003	0.011	0.013	255.393	309.013	637.775	535.986
VD-Hydrology	0	0	0	0	0	604.932	1001.951	1832.466	1158.855
HD-Roadways	3.031	2.291	2.802	3.048	3.354	1.490	5.451	48.659	2.833
Hillshade-9am	0	0	0	0.014	0.023	11.302	42.622	471.577	23.392
Hillshade-noon	0	0	0	0	0	11.079	42.675	462.257	21.994
Hillshade-3pm	0	0	0	0	0	22.583	75.458	742.779	47.541
HD-Fire-Points	1.082	1.981	2.148	2.617	2.930	3.289	6.660	43.331	2.588
Total	**6.445**	**6.829**	**7.818**	**9.341**	**10.850**	**1357.616**	**2937.163**	**7482.304**	**2580.383**

respecto a estos valores, a diferencia del esquema propuesto que se centran en la modificación de aquellos valores de mayor magnitud generando un error mínimo en estos atributos, además de tomar como ventaja la reducción de registros a marcar.

Los resultados que se muestran en la tabla 5.2, fueron obtenidos insertando la misma cantidad de bits para cada uno de los esquemas, como puede observarse en la parte superior de la tabla. Posteriormente, para el esquema propuesto se varió el número de bits máximo a insertar por tupla $nB = [1 - 5]$, en cuanto a Li, la inserción se realizó utilizando el LSB: 1,4 y 8, y finalmente para Mohanpurkar se utilizaron los parámetros mínimos $c = 1, t = 0.01$.

En cada uno de estos casos se cálculo el error relativo para cada atributo y se realizó el sumatorio para obtener el total generado en la base de datos, como se puede observar en la parte inferior de la tabla. Estos resultados presentan una gran diferencia entre el trabajo propuesto y los trabajos de Li y Mohanpurkar, debido a la explicación antes mencionada.

Los experimentos presentados en esta sección fueron desarrollados utilizando 100000 registros de la base de datos, por lo cual se anexan en el Apéndice A, las gráficas correspondientes a los experimentos restantes.

5.3. Evaluación de la capacidad

Este experimento sirvió para evaluar la cantidad de bits de la huella digital insertados por el esquema propuesto en relación a los parámetros establecidos por el usuario, esto debido a que la capacidad de inserción del esquema propuesto, depende directamente de dos variables asignadas por el usuario o propietario de la base de datos, la fracción de tupla (y) y el número de bits máximo a insertar por tupla (nB). La fracción de tupla es usada para seleccionar las tuplas que serán marcados con la huella digital. Esto indica que en promedio n/y tuplas son marcadas en toda la base de datos, siendo n el total de registros de la base de datos.

Por otro lado la variable (nB), indica el número de bits máximo a insertar por cada tupla seleccionada, con base a la fracción de tupla. Sin embargo, pese a que la inserción se realiza de forma uniforme, esta depende de la operación módulo entre el valor Hash SHA-256 y la fracción de tupla o nB. Por lo tanto, a pesar de que las huellas utilizadas en estos experimentos tienen una longitud de 1842 bits, el espacio de inserción debe ser mayor a esta cantidad para asegurar que cada bit de la huella digital se haya insertado al menos una vez en la relación de la RDB.

La evaluación se realizó tomando diferentes tamaños de la base de datos Forest CoverType: 5000, 10000, 50000 y 100000 registros cada una. Además, se varió el tamaño de fracción de tupla en, $y = 1, 3, 5, 10$, y para $nB = 1, 2$ y 3.

En las gráficas de la figura 5.4 se observa el comportamiento del esquema de acuerdo a diferentes tamaños de bases de datos, resaltando los resultados mediante etiquetas que indican el total de bits insertados, en aquellos casos donde la inserción de la huella fue

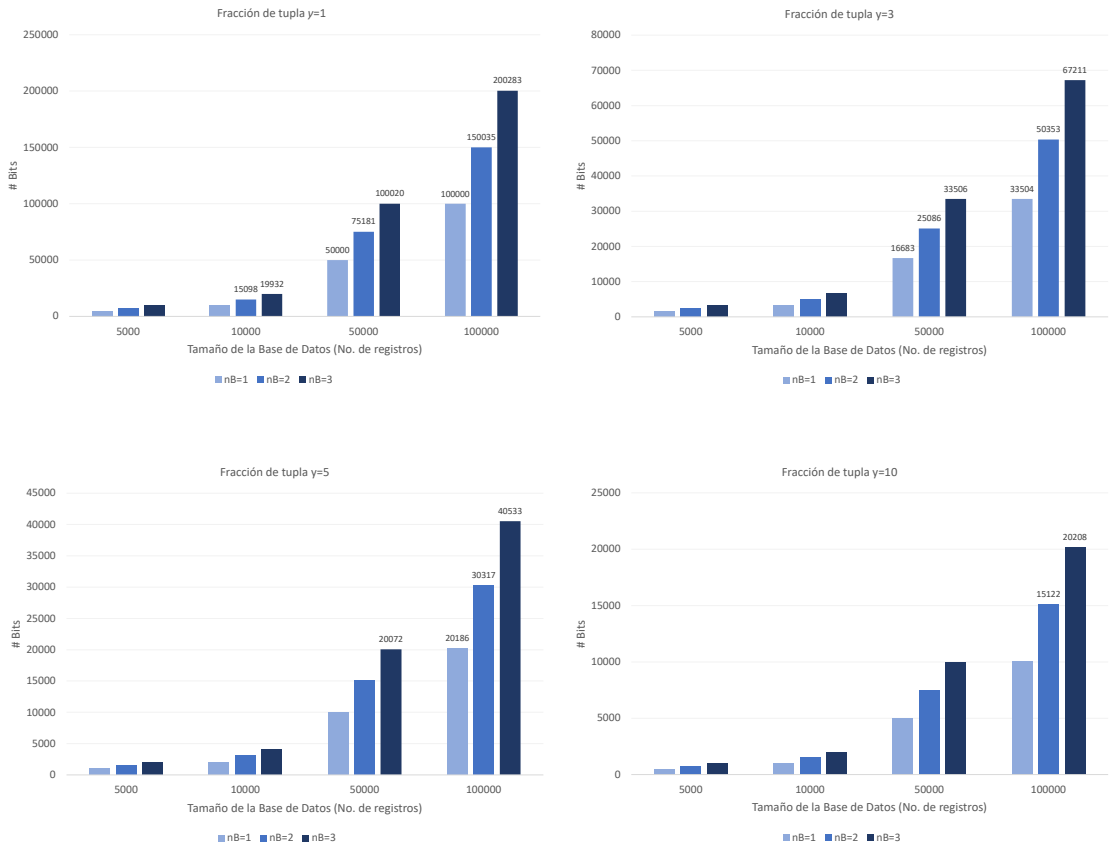


Figura 5.4: Capacidad de inserción del esquema propuesto.

completa, es decir todos los bits de huella fueron insertados. Por otro lado, la reducción de registros seleccionados por la fracción de tupla, afecta a la capacidad de inserción, por lo cual en estos casos el aumento en el número de bits por tupla aportaría una mejora en la capacidad, además de permitir la utilización de bases de datos con un menor número de registros.

En la gráfica de la figura 5.5, se observa el comportamiento de inserción del esquema en cuanto a la fracción de tupla y la cantidad de bits de huella insertados en una base de datos de 5000 registros, en la cual se observar que la capacidad de inserción tiende a duplicarse cuando $nB = 3$, sin embargo, cabe mencionar que en este caso la huella digital no es insertada en su totalidad debido a su longitud.

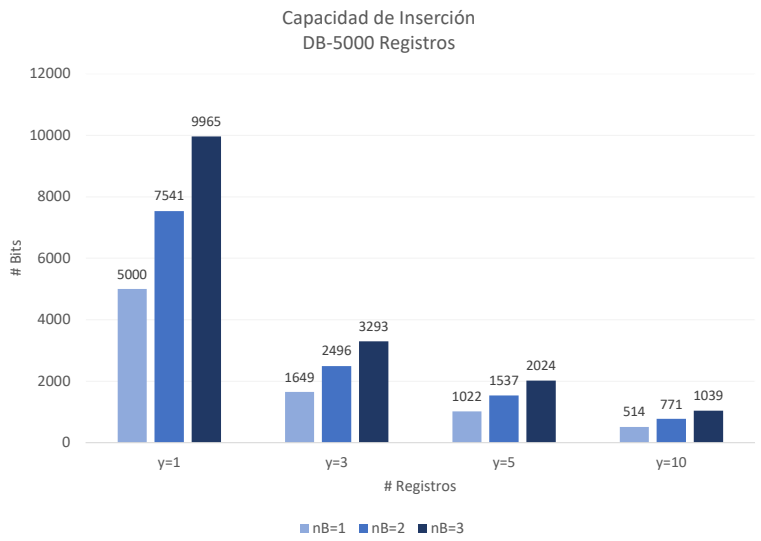


Figura 5.5: Comportamiento con una base de datos de menor tamaño.

Por otro lado, en la gráfica de la figura 5.6, el incremento de registros que podemos observar en estos resultados aportan una mayor área de inserción de la huella, sin embargo, cuando se selecciona $y = 10$, para $nB = 1$ es aún imposible insertar la huella digital completa, pese a tener una base de datos de mayor tamaño. Por lo cual, uno de nuestros objetivos fue aumentar la capacidad de inserción para hacer mas flexible el uso de huellas digitales bajo el esquema de Tardos (2008), como se puede observar cuando $nB = 2$ y $nB = 3$, donde la huella digital es insertada en su totalidad utilizando el mismo número de registros.

Derivado de estos resultados, podemos concluir que esta estrategia de incremento de la capacidad de inserción permite que huellas digitales como la de Tardos, puedan ser utilizadas en bases de datos de menor tamaño de acuerdo también a la distorsión permitida por el propietario. Por otro lado, es de suma importancia que la huella digital sea insertada en su totalidad, ya que de otra forma al no recuperar el 100 % de bits durante el proceso de detección, la identificación del usuario traidor no podría llevarse a cabo, además de que ataques a la base de datos reducirían aún más su recuperación.

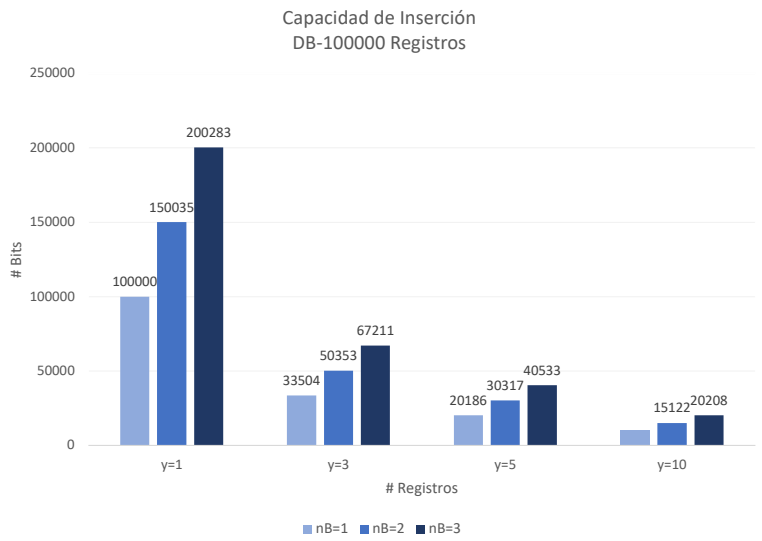


Figura 5.6: Comportamiento con una base de datos de mayor tamaño.

5.4. Evaluación de robustez

Como se mencionó al principio del documento, el objetivo principal de los esquemas de huella digital es la protección de la propiedad de la base de datos mediante la inserción de la huella digital, sin embargo, ésta puede estar expuesta a manipulaciones intencionales (ataque) o no intencionales, con el objetivo de distorsionar y evitar su detección. Por lo tanto, en estos experimentos se evaluó la capacidad del esquema de soportar tres de las operaciones más comunes en el manejo de bases de datos: Inserción, eliminación y actualización de registros.

Además, se realiza una comparación con el esquema propuesto por Li et al. (2005), debido a su estrategia de inserción y recuperación de huella digital ante este tipo de ataques. Por otro lado, la comparación con el esquema propuesto por Mohanpurkar and Joshi (2015c) no fue posible, ya que solo considera la resistencia a ataques de colusión utilizando (Tardos, 2008), y no soporta otro tipo de ataque debido a que requiere del almacenamiento del orden de las modificaciones realizadas en los registros seleccionados durante el proceso de inserción y cualquier cambio altera la sincronización entre el

algoritmo de detección y la base de datos.

Los experimentos fueron realizados utilizando la base de datos Forest CoverType con los primeros 100000 registros, lo cual permitió observar el rendimiento de la robustez para diferentes valores en los parámetros y y nB.

5.4.1. Inserción de registros

Este experimento consistió en la inserción de registros artificiales a la base de datos marcada en diferentes porcentajes (10 % - 90 %), con el objetivo de evaluar el porcentaje de detección de la huella digital posterior al ataque. En este experimento se evalúo la correcta recuperación de los bits de la huella digital, ya que la inserción de registros puede alterar el resultado al aplicar la función de mayoría durante el proceso de recuperación, debido a que bits de huella incorrectos pueden ser recuperados.

El ataque de inserción se realizó de la siguiente forma:

1. Inserción de la huella digital generada por Tardos con $c = 2$ y $\epsilon = 0.01$, variando la fracción de tupla y el número de bits por registro(nB).
   - Fracción de tupla $y = 1, 3, 5, 7, 10$ y 15.
   - Número de bits por registro $nB = 1, 2$ y 3. (Por lo analizado en la subsección de evaluación de la distorsión).
2. Creación de nueve copias de la relación de la base de datos ya marcada.
3. Inserción de registros aumentando el porcentaje de severidad del ataque (10 %-90 %).
   - Selección de valores pseudo-aleatorios para cada atributo en la creación de registros artificiales.
4. Ejecución del algoritmo de detección, reportando el porcentaje de recuperación de la huella digital.

Los resultados de la ejecución de estos experimentos son mostrados en la figura 5.7.

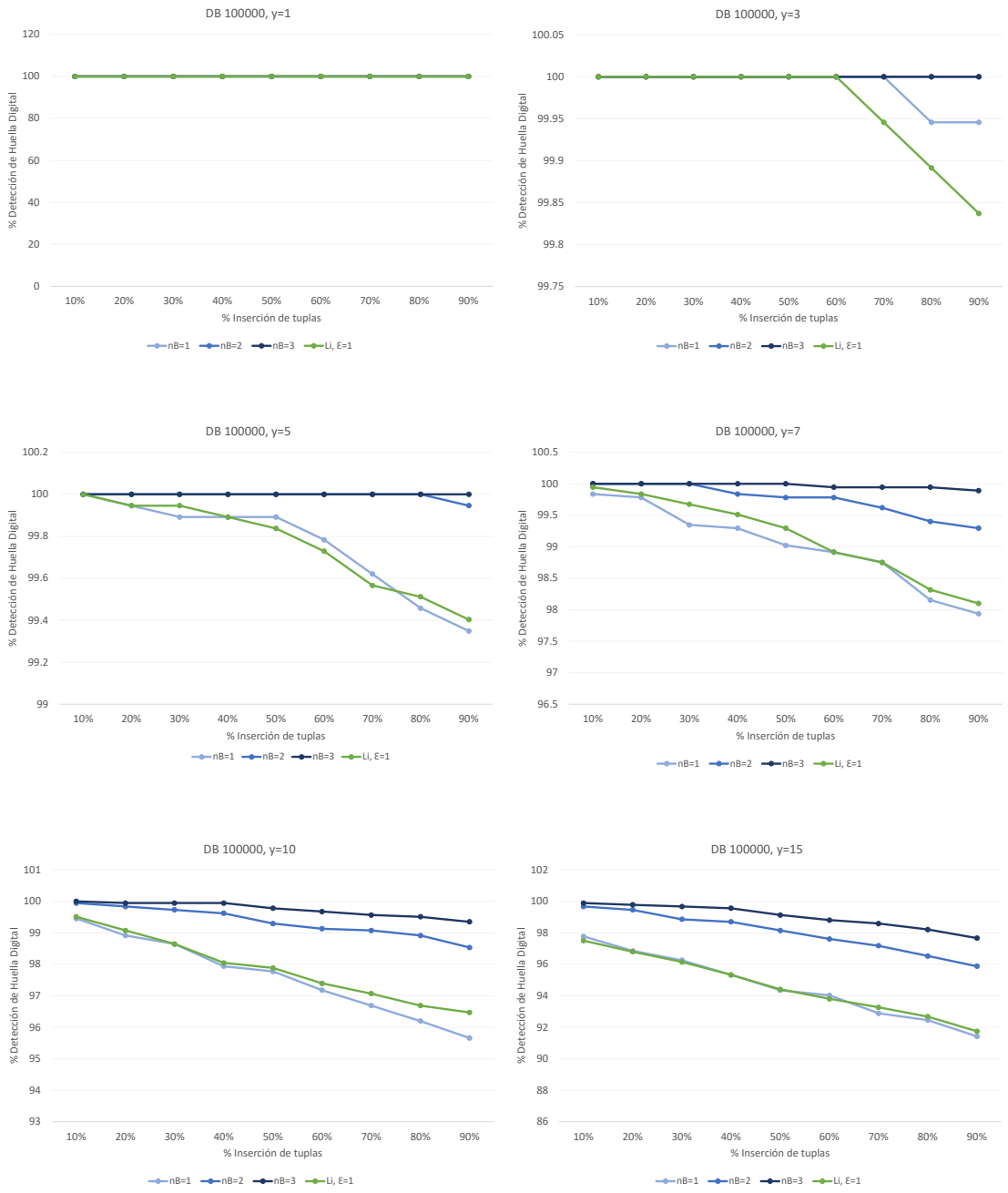


Figura 5.7: Ataque de inserción de tuplas con variación de la fracción de tupla.

En estos experimentos se aprecia la relación que existe entre la fracción de tupla con que fue insertada la huella digital y el porcentaje de recuperación de la huella posterior al ataque de inserción. En este caso, la primer gráfica de la figura 5.7, indica la inserción de la huella digital con una fracción de tupla $y = 1$, es decir todos los registros fueron utilizados para la inserción de la huella digital y por lo tanto los bits de la huella digital fueron insertados con mayor redundancia, lo cual permite que pese a la recuperación de bits erróneos por parte de los registros artificiales, la aplicación de la función de mayoría mantenga los valores originales de la huella digital.

Sin embargo, cuando la fracción de tupla aumenta, la recuperación de bits erróneos reduce el porcentaje de bits de huella digital originales y por lo tanto la detección del usuario de la huella digital se puede ver comprometida. Por lo tanto, el incremento en el número de bits por registro a insertar (nB), mejora el porcentaje de recuperación de la huella digital, como se observa cuando $y = 3, 5, 7, 10$ y 15. Además, en este caso la comparación con la propuesta de Li et al. (2005), refleja resultados muy similares a los obtenidos cuando $nB = 1$.

5.4.2. Alteración de registros

La realización de este experimento sirvió para medir el porcentaje de recuperación de la huella digital posterior a un ataque de alteración de registros, esto debido a que la modificación de los valores de las tuplas de la relación pueden distorsionar el valor del bit de la huella en caso de haber sido insertado en estos valores. Este experimento consistió en la actualización de registros de la base de datos previamente marcada, esto se realizó seleccionando pseudo aleatoriamente hasta 5 valores por registro para su actualización.

El ataque de alteración se realizó de la siguiente forma:

1. Inserción de la huella digital generada por Tardos con $c = 2$ y $\epsilon = 0.01$, variando la fracción de tupla y el número de bits por registro (nB).

- Fracción de tupla $y = 1, 3, 5, 7, 10$ y 15.
- Número de bits por registro $nB = 1, 2$ y 3. (Por lo analizado en la subsección de evaluación de la distorsión).

2. Creación de nueve copias de la relación de la base de datos ya marcada.
3. Alteración de registros aumentando el porcentaje de severidad del ataque (10 %-90 %).
   - Selección del número de atributos a actualizar por cada registro.
   - Selección de valores pseudo-aleatorios para cada atributo a actualizar.
4. Ejecución del algoritmo de detección, reportando el porcentaje de recuperación de la huella digital.

En las gráficas de la figura 5.8, se aprecia una mejora en la recuperación porcentual de la huella digital, cuando $nB = 2$ y 3, sin embargo, la propuesta de Li et al. (2005) obtiene mejores resultados en comparación a nuestra propuesta cuando $nB = 1$, esto debido a la aleatoriedad en la selección de los atributos para la inserción de los bits de huella

Por otro lado, como se mencionó en el ataque de inserción, estos resultados están relacionados con la fracción de tupla, con la que fue insertada la huella digital en todas las gráficas mostradas.

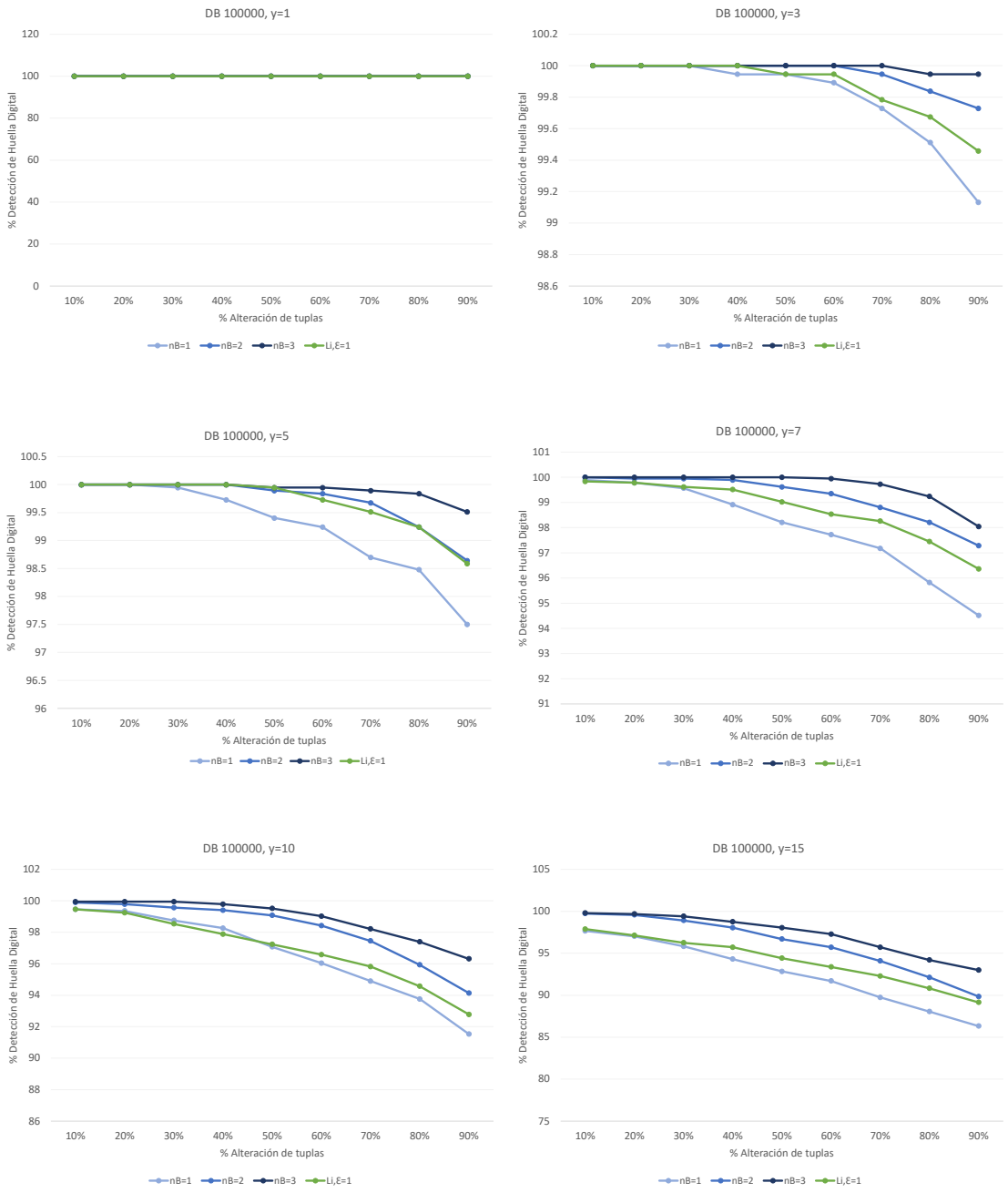


Figura 5.8: Ataque de alteración de tuplas con variación de la fracción de tupla.

5.4.3. Eliminación de registros

En este experimento, se realizó la eliminación de registros progresivamente, es decir, incrementando la severidad del ataque del 10 % al 90 %. Este experimento sirvió para observar la capacidad del esquema para resistir este tipo de ataque, ya que en este caso la eliminación de registros tiene el objetivo de eliminar bits de la huella digital que hayan sido insertados y de este modo evitar la identificación de usuarios por medio de la huella digital.

Sin embargo, uno de los objetivos del atacante es obtener la mayor cantidad de datos posibles para su beneficio, por lo cual la severidad del ataque recae en la cantidad de tuplas a obtener por parte del atacante.

El proceso de eliminación de tuplas se realizó de la siguiente forma:

1. Inserción de la Huella Digital generada por Tardos (2008) con $c = 2$ y $\epsilon = 0.01$, variando la fracción de tupla y el número de bits por registro (nB).
   - Fracción de tupla $y = 1, 3, 5, 7, 10$ y 15.
   - Número de bits por registro $nB = 1, 2$ y 3. (Por lo analizado en la subsección de evaluación de la distorsión).
2. Realización de nueve copias de la base de datos ya marcada.
3. Eliminación de registros para cada una de las copias generadas, de acuerdo al porcentaje correspondiente (10 % - 90 %).
   - Selección de registros a eliminar utilizando la función RAND del lenguaje SQL bajo la inicialización de una semilla fija para cada uno de los porcentajes reportados.
4. Ejecución del algoritmo de detección, reportando el porcentaje de recuperación de la huella digital.

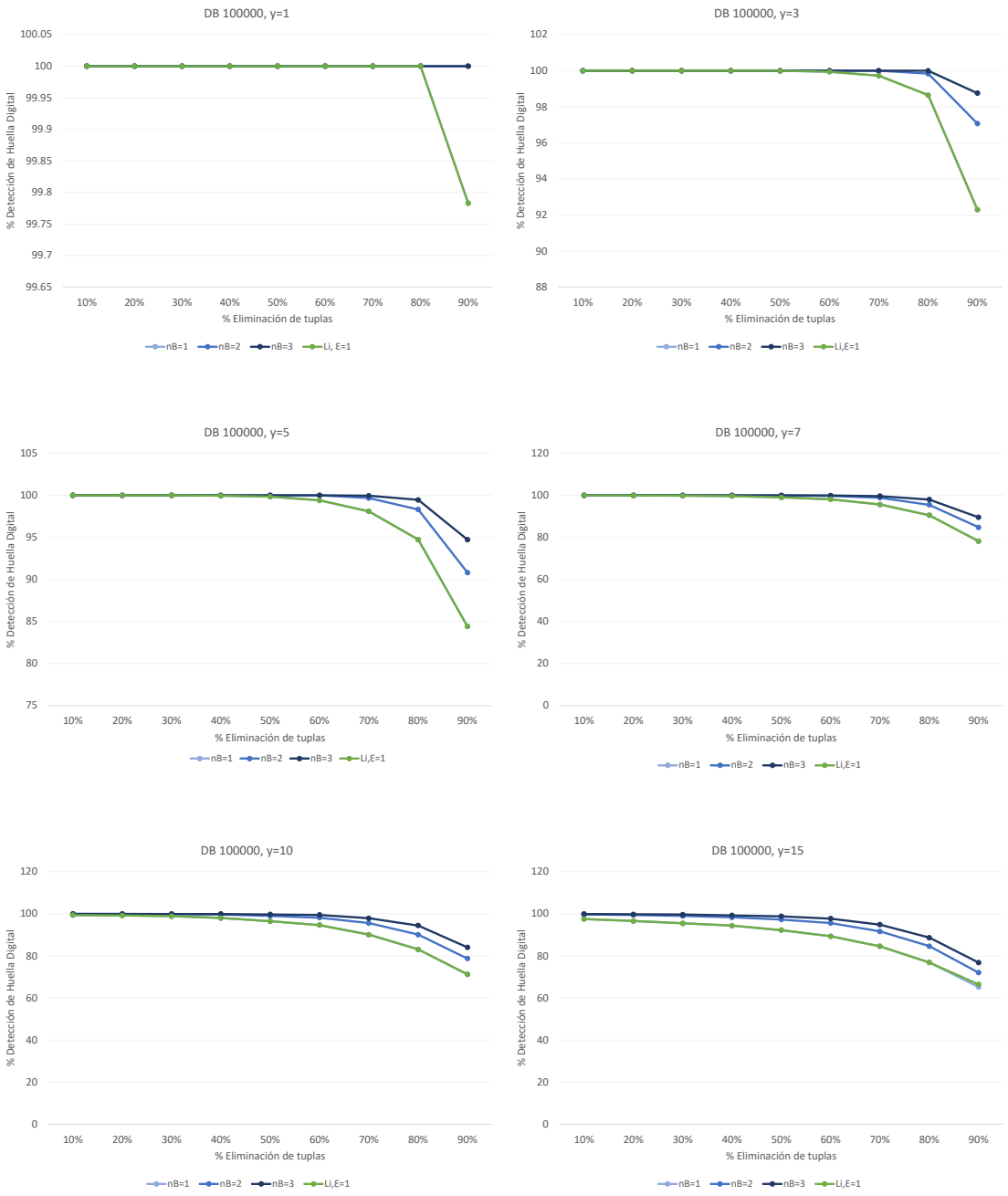


Figura 5.9: Ataque de eliminación de tuplas con variación de la fracción de tupla.

En las gráficas de la figura 5.9, se observa que el incremento en el parámetro nB, mejora el porcentaje de recuperación de la huella digital de un 1 % a un 10 % respecto a Li.

La importancia de esta mejora se encuentra en que la recuperación de un mayor número de bits de la huella digital, puede significar la identificación del usuario propietario de la base de datos analizada, ya que el proceso de rastreo del traidor mejora a medida que encuentra un mayor número de coincidencias entre la huella digital extraída y las huellas digitales distribuidas, como se explica en el capítulo anterior.

5.4.4. Ataque de colusión

El ataque de colusión, como se ha descrito a lo largo del trabajo, se deriva del funcionamiento de los esquemas de huella digital, es decir bajo el escenario de distribución de copias de la misma relación de una base de datos a múltiples usuarios. Debido a que la inserción de huellas digitales distintas, en cada una de las copias a distribuir, provoca que al realizar la comparación a nivel de bits de dos o más copias se encuentren diferencias entre ellas y por lo tanto se releve información de la localización de la huella digital insertada, utilizada posteriormente para generar una copia con una huella digital distinta que no pueda ser identificada.

Derivado de la importancia de este ataque, este trabajo previene y resiste los ataques de colusión incorporando el esquema de Tardos (2008) al esquema de huella digital propuesto. Por otra parte, si bien el esquema de Tardos ha sido implementado por algunos trabajos del estado del arte, su aplicación ha sido enfocada en bases de datos de gran tamaño, debido a las longitudes de las huellas digitales. Por lo tanto, la aportación respecto a este tipo de ataque, es la amplitud de su aplicación a bases de datos de diferentes tamaños manteniendo una baja distorsión cuando la capacidad de inserción es incrementada.

Finalmente, por lo descrito anteriormente los experimentos relacionados al ataque de colusión no fueron necesarios, desde que el esquema de huella digital propuesto, incorpora el esquema de generación de huella digital resistente a colusión de G. Tardos. La utilización de este esquema asegura la identificación de al menos un usuario de una

coalición de c usuarios, parámetro que es establecido durante la generación de las huella digitales a utilizar, anterior a la inserción. Por lo tanto, el esquema será resistente a ataques de colusión, en la medida de lo establecido por c y el límite de error ϵ.

5.5. Discusión de los resultados

Los resultados obtenidos a través de los experimentos realizados para evaluar la distorsión, capacidad y robustez del esquema fueron favorables ya que tendieron a una mejora respecto a los trabajos de Li y Mohanpurkar. Sin embargo, su implementación en una base de datos real podría cambiar el comportamiento de estos resultados bajo ciertos escenarios.

En primer lugar, la reducción de la distorsión en cuanto al error relativo de los valores de la base de datos puede cambiar, ya que si los valores analizados dentro de una tupla son de longitud similar o de longitud menor al umbral d, los resultados serían similares en todos los esquemas, ya que el trabajo propuesto se centra en la selección de valores de mayor longitud con el objetivo de generar menor distorsión, lo cual no sería posible en este caso.

Por otro lado, la capacidad de inserción y robustez del esquema siempre estará sujeta a los parámetros y, nB que defina el propietario ya que estos requerimientos siempre van de acuerdo a los niveles de distorsión aceptables para la base de datos.

Finalmente, en los experimentos de evaluación de la robustez del esquema observamos que al incrementar la capacidad de inserción la recuperación de la huella digital mejora entre 1 % y 10 %, lo cual puede significar la identificación o no del traidor o usuario de la base de datos.

6

Conclusiones y trabajo futuro

6.1. Conclusiones

Este trabajo de investigación presenta un nuevo esquema de huella digital enfocado en la mejora de tres principales requerimientos de las marcas de agua en base de datos relacionales: usabilidad, capacidad y robustez. También considera el problema de colusión, principal ataque que se presenta en esquemas de huella digital, el cual fue resuelto a través de la utilización del esquema de generación de huellas digitales de G.Tardos. Sin embargo, su utilización presenta limitantes en cuanto a la posición y área de inserción en la base de datos. Por lo tanto, el esquema mejora la capacidad de inserción minimizando la distorsión generada para que este tipo de huella digital sea aplicable a distintos tamaños de bases de datos.

Además, la estrategia de inserción propuesta provee un mecanismo para el incremento de la capacidad en relación al número de bits insertados en cada registro, garantizando a su vez la reducción de la distorsión generada en términos del MSE, siendo igual o menor que en esquemas del estado del arte. Por otra parte, la comparación respecto a estos esquemas bajo las mismas condiciones de capacidad de inserción, el esquema propuesto reduce el número de registros utilizado para la inserción de la huella digital,

lo cual se traduce en la conservación de un mayor número de registros con sus valores reales y también en la reducción del tiempo de inserción realizado por el esquema.

Por otra parte, este trabajo también se enfoca en el error relativo generado durante la modificación de los valores reales de la base de datos, lo cual marca una diferencia respecto al estado del arte. Este enfoque parte del análisis del error resultante en la modificación de los primeros n bits menos significativos en los valores de la base de datos de acuerdo a su longitud de bits, resaltando la importancia del significado de los valores en las bases de datos y el dominio de sus atributos.

En cuanto a la robustez, el esquema mejora el porcentaje de recuperación de la huella digital entre 1 % y 10 % a diferentes niveles de severidad de ataques de inserción, eliminación y alteración de tuplas.

Por otra parte, una desventaja que presenta el esquema propuesto respecto a la robustez, se da cuando el orden de los atributos es alterado, ya sea de forma intencional (ataque) o no. Esto ocurre debido a que más de un valor puede tener la misma longitud de bits y cambiar el orden durante el proceso de recuperación de la huella digital, ya que la estrategia de reducción de la distorsión la cual es utilizada tanto en el algoritmo de inserción como el de detección, depende del ordenamiento de los valores de los registros de acuerdo a su longitud de bits.

Finalmente, tanto los parámetros de inserción de la huella digital, como la llave secreta, están sujetos a la definición por parte del propietario de la base de datos, lo cual significa que la distorsión, capacidad y robustez de la huella digital recaerán en las restricciones y el tipo de aplicación que el propietario establezca.

6.2. Trabajo futuro

El esquema de huella digital propuesto, si bien mejora algunos de los requerimientos de los esquemas de huella digital en bases de datos relacionales, es posible mejorar aspectos que se presentaron en el desarrollo de este trabajo. A continuación se presentan las posibles mejoras a realizar.

1. Desarrollar un mecanismo de seguridad para la construcción de la huella digital y su almacenamiento, o en su caso un mecanismo que no requiera del almacenamiento de las huellas digitales, para su utilización en el proceso de identificación del traidor o usuario propietario de una determinada base de datos.

2. Desarrollar una estrategia para determinar los parámetros de inserción adecuados para cada una de las bases de datos a marcar, es decir aquellos que provean un equilibrio entre distorsión y robustez. Esto como sugerencia al propietario de la base de datos, ya que actualmente el propietario define estos parámetros desconociendo el impacto de la inserción de la huella digital en la base de datos.

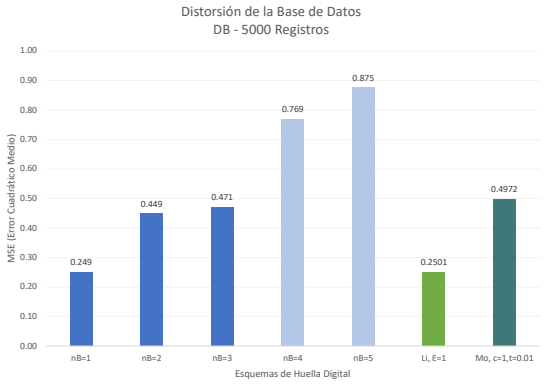
3. Uno de los problemas más importantes que se observaron en el desarrollo de este trabajo, aún con las mejoras en la capacidad de inserción y la reducción de la distorsión, fue la longitud de las huellas digitales resistente a ataques de colusión propuestas por Tardos (2008), por lo cual es necesario proponer un esquema de generación de huellas digitales que reduzca la longitud, con el objetivo de ampliar la aplicación del esquema a bases de datos con mayor sensibilidad a modificaciones.

4. Finalmente se propone mejorar el proceso de ordenamiento de los atributos para la selección de los valores con mayor longitud de bits, ya que el cambio de orden de los atributos podría afectar la recuperación de la huella digital.

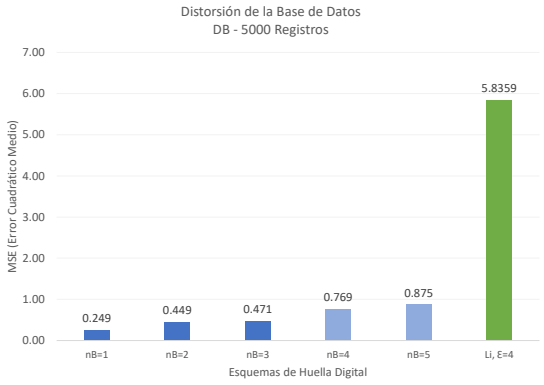
A

Apéndice: Comparación de la distorsión generada en la RDB

En este apéndice se presentan los resultados de distorsión obtenidos para las bases de datos de 5000,10000 y 50000 registros, los cuales son mencionados en el Capítulo 5. Las primeras dos gráficas hacen referencia a la comparación de distorsión en términos del MSE, variando $nB = [1, 5]$ en el esquema propuesto, los bit menos significativos en el esquema de Li y los mínimos parámetros del esquema de Mohanpurkar.



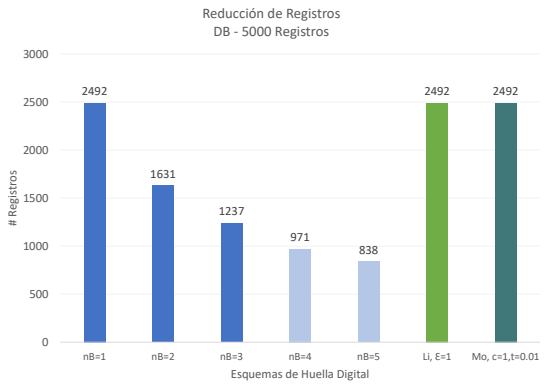
(a) 1 Bit/Registro



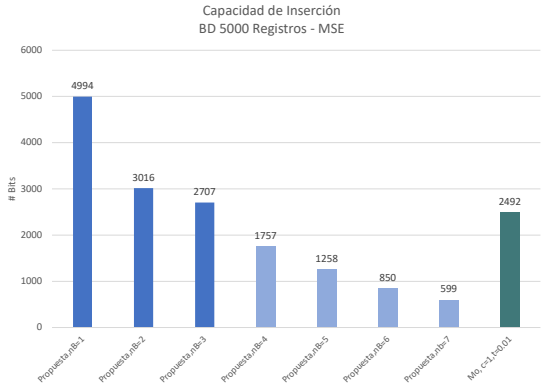
(b) Comparación con Li cuando ε=4

La tercer gráfica muestra el efecto de reducción de registros que se presenta al

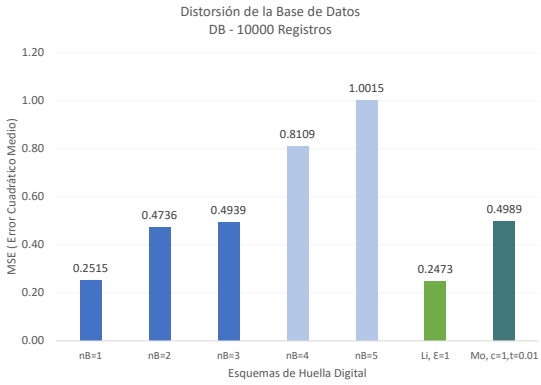
realizar esta comparación, basándose en un número de bits de inserción específico. Finalmente, en la cuarta gráfica se observa la comparación de la capacidad de inserción, respetando el limite de distorsión definido por el esquema de Mohanpurkar.



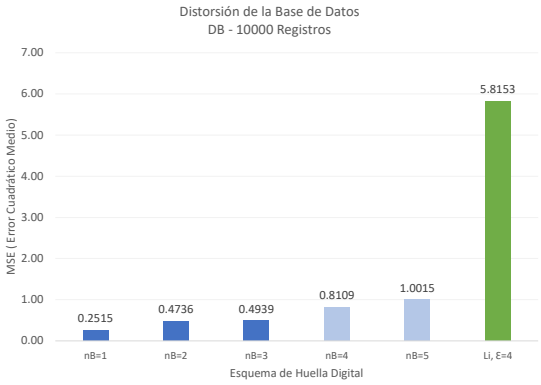
(a) Efecto de reducción de registros a marcar



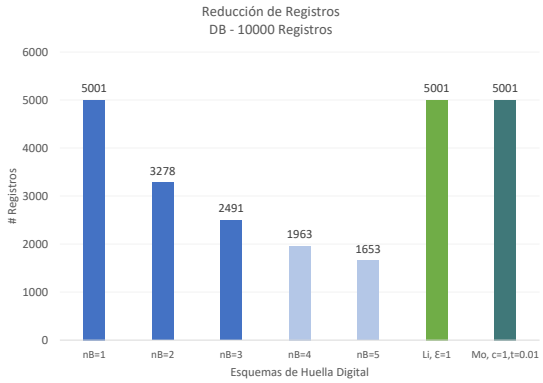
(b) Capacidad de inserción con limite de distorsión



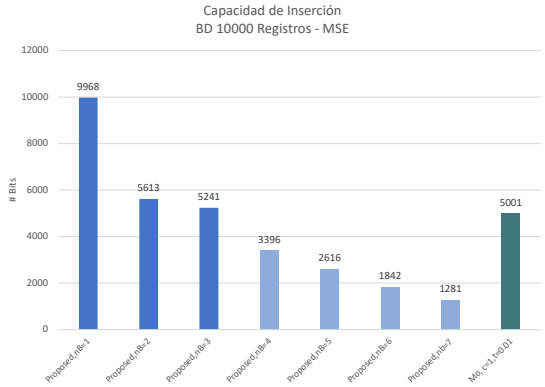
(a) 1 Bit/Registro



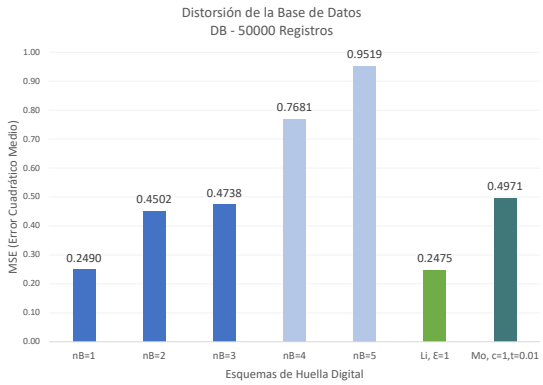
(b) Comparación con Li cuando ε=4



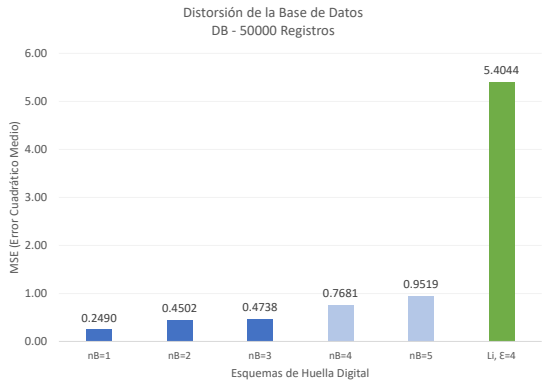
(a) Efecto de reducción de registros a marcar



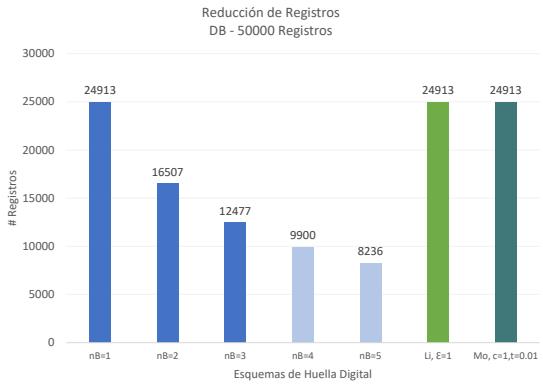
(b) Capacidad de inserción con limite de distorsión



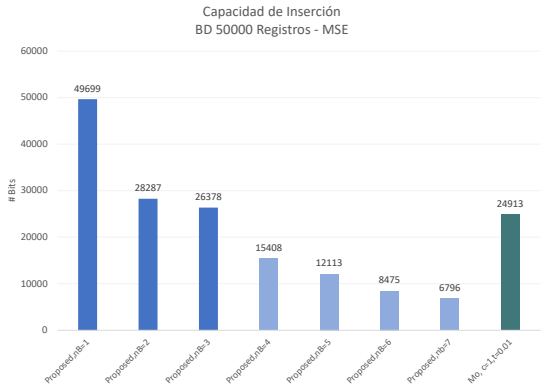
(a) 1 Bit/Registro



(b) Comparación con Li cuando ε=4



(c) Efecto de reducción de registros a marcar



(d) Capacidad de inserción con limite de distorsión

Bibliografía

Agrawal, R. and Kiernan, J. (2002). Watermarking relational databases. In *VLDB'02: Proceedings of the 28th International Conference on Very Large Databases*, pages 155–166. Elsevier.

Ahmad, M., Shahid, A., Qadri, M. Y., Hussain, K., and Qadri, N. N. (2017). Fingerprinting non-numeric datasets using row association and pattern generation. In *Communication Technologies (ComTech), 2017 International Conference on*, pages 149–155. IEEE.

Alfagi, A. S., Manaf, A. A., Hamida, B., and Hamza, M. G. (2017). A systematic literature review on necessity, challenges, applications and attacks of watermarking relational database. *In Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, 9(1-3):101–108.

Boneh, D. and Shaw, J. (1998). Collusion-secure fingerprinting for digital data. *In IEEE Transactions on Information Theory*, 44(5):1897–1905.

Bui, T. V., Nguyen, B. Q., Nguyen, T. D., Sonehara, N., and Echizen, I. (2013). Robust fingerprinting codes for database. In *International Conference on Algorithms and Architectures for Parallel Processing*, pages 167–176. Springer.

Bui, T. V., Nguyen, T. D., Sonehara, N., and Echizen, I. (2015). Tradeoff between the price of distributing a database and its collusion resistance based on concatenated codes. In *International Conference on Algorithms and Architectures for Parallel Processing*, pages 163–182. Springer.

Chor, B., Fiat, A., and Naor, M. (1994). Tracing traitors. In *Annual International Cryptology Conference*, pages 257–270. Springer.

Cox, I., Miller, M., Bloom, J., Fridrich, J., and Kalker, T. (2007). *Digital watermarking and steganography*. Morgan Kaufmann, 2 edition. ISBN:9780123725851.

Elmasri, R. and Navathe, S. (2010). *Fundamentals of database systems*. Addison-Wesley Publishing Company, 6 edition. ISBN 13: 978-0-136-08620-8.

Ferguson, N., Schneier, B., and Kohno, T. (2011). *Cryptography engineering: design principles and practical applications*. John Wiley & Sons, 1 edition. ISBN:978-0-470-47424-2.

Franco-Contreras, J. and Coatrieux, G. (2016). Databases traceability by means of watermarking with optimized detection. In *International Workshop on Digital Watermarking*, pages 343–357. Springer.

Guo, F., Wang, J., and Li, D. (2006). Fingerprinting relational databases. In *Proceedings of the 2006 ACM symposium on Applied computing*, pages 487–492. ACM.

Gursale, N. and Arti, M. (2014). A robust, distortion minimization fingerprinting technique for relational database. *In International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC)*, 2(6):1737–1741.

Halder, R., Pal, S., and Cortesi, A. (2010). Watermarking Techniques for Relational Databases : Survey , Classification and Comparison. 16(21):3164–3190.

Iftikhar, S., Anwar, Z., and Kamran, M. (2014). A novel and robust fingerprinting technique for digital data based on genetic algorithm. In *High-capacity Optical Networks and Emerging/Enabling Technologies (HONET), 2014 11th Annual*, pages 173–177. IEEE.

Kamran, M. and Farooq, M. (2013). A formal usability constraints model for watermarking of outsourced datasets. *In IEEE transactions on information forensics and security*, 8(6):1061–1072.

Kieseberg, P., Schrittwieser, S., Mulazzani, M., Echizen, I., and Weippl, E. (2014). An algorithm for collusion-resistant anonymization and fingerprinting of sensitive microdata. *In Electronic Markets*, 24(2):113–124.

Laarhoven, T. (2011). *Collusion-resistant traitor tracing schemes.* PhD thesis, M. Sc. thesis, Dept. Math. Comp. Sc., Eindhoven Univ. Techn., Eindhoven, The Netherlands.

Lafaye, J., Gross-Amblard, D., Constantin, C., and Guerrouani, M. (2008). Watermill: An optimized fingerprinting system for databases under constraints. *In IEEE Transactions on Knowledge and Data Engineering*, 20(4):532–546.

Li, Y. (2008). Database watermarking: A systematic view. In *Handbook of database security*, pages 329–355. Springer.

Li, Y., Swarup, V., and Jajodia, S. (2005). Fingerprinting relational databases: Schemes and specialties. *In IEEE Transactions on Dependable and Secure Computing*, 2(1):34–45.

Liu, K. R. (2005). *Multimedia fingerprinting forensics for traitor tracing*, volume 4. Hindawi Publishing Corporation. ISBN:977-5945-18-6.

Liu, S., Wang, S., Deng, R. H., and Shao, W. (2004). A block oriented fingerprinting scheme in relational database. In *International Conference on Information Security and Cryptology*, pages 455–466. Springer.

Maier, D. (1983). *The theory of relational databases*, volume 11. Computer science press Rockville. ISBN:9780914894421.

Mehta, B. B. and Aswar, H. D. (2014). Watermarking for security in database: A review. In *IT in Business, Industry and Government (CSIBIG), 2014 Conference on*, pages 1–6. IEEE.

Mohanpurkar, A. and Joshi, M. (2015a). The effect of the novel anti-collusion fingerprinting scheme on the knowledge from numeric databases. *In International Journal of Scientific and Engineering Research*, 6(12):334–338.

Mohanpurkar, A. and Joshi, M. (2015b). Fingerprinting numeric databases with information preservation and collusion avoidance. *In International Journal of Computer Applications*, 130(5):13–18.

Mohanpurkar, A. and Joshi, M. (2015c). A fingerprinting technique for numeric relational databases with distortion minimization. In *Computing Communication Control and Automation (ICCUBEA), 2015 International Conference on*, pages 655–660. IEEE.

Ndajah, P., Kikuchi, H., Yukawa, M., Watanabe, H., and Muramatsu, S. (2011). An investigation on the quality of denoised images. *In Int. J. Circuits, Systems and Signal Process.*, 5:423–434.

Piattini, M. (2000). *Diseño de Bases de Datos Relacionales.* Madrid,(España). ISBN:9701505263.

Schneider, B. (1996). *Applied cryptography: protocols, algorithms, and source code in C.* John Wiley & Sons, Inc. ISBN:0471128457.

Schneier, B. (2011). *Secrets and lies: digital security in a networked world.* John Wiley & Sons. ISBN:9781119183631.

Stinson, D. R. (2005). *Cryptography: theory and practice.* CRC press, 3 edition. ISBN:1-58488-508-4.

Tardos, G. (2008). Optimal probabilistic fingerprint codes. *In Journal of the ACM*, 55(2):1–24.

Uzun, E. and Stephenson, B. (2008). Security of relational databases in business outsourcing. *HP Labs Report HPL-2008-168.*

Waghmode, M. V. and Mohanpurkar, M. A. (2014). Collusion avoidance in fingerprinting outsourced relational databases with knowledge preservation. *International Journal on Recent and Innovation Trends in Computing and Communication*, 2(5):1332–1337.

Xie, M.-R., Wu, C.-C., Shen, J.-J., and Hwang, M.-S. (2016). A survey of data distortion watermarking relational databases. *In IJ Network Security*, 18(6):1022–1033.

Zhou, M., Wang, J., Wang, C., and Li, D. (2007). A novel fingerprinting architecture for relational data. In *Digital EcoSystems and Technologies Conference, 2007. DEST'07. Inaugural IEEE-IES*, pages 477–480. IEEE.