HASHING. UN CONCEPTO. UNA REALIDAD

Valdes G., Domingo
domingo.valdes1@utp.ac.pa
Tejedor-Morales, María Yahaira
Universidad Tecnológica de Panamá, C.R. de Coclé, Grupo de investigación SoftSolution Group,
maria.tejedor@utp.ac.pa

***Resumen*— Los escenarios tecnológicos actuales vistos como soluciones a problemas inherentes al intercambio y confidencialidad de la información, hacen del tema que nos ocupa en este artículo una oportunidad para comprender cómo la abstracción de un concepto puede materializarse en una realidad.**

Este artículo expone la función hash como método para convertir una clave dada en una dirección, con el fin de obtener la ruta específica del mismo dentro de un conjunto superior de datos. Dicha ruta se obtiene mediante el uso de procedimientos matemáticos traducidos a algoritmos con el propósito de ser aplicados en sistemas de seguridad, mensajería, banca, administración de datos y otros.

Paralelamente a la generación de llaves existe una amplia gama de métodos para el manejo y control de errores o colisiones que se puedan suscitar durante la ejecución de esta función, sin limitar su principal objetivo que es brindar una técnica de cálculo de direcciones con independencia en cuanto a la relación tiempo – cantidad de datos y flexible al costo de uso de memoria y del sistema.

En este artículo presentamos una revisión teórica literaria del tema propuesto y de igual manera exponemos cómo las funciones hash se encuentran presentes en los procesos con los cuales diariamente interactuamos en la sociedad.

Se plantea parte del panorama global y regional en donde esta conceptualización se cristaliza en aplicaciones que proveen un marco de trabajo seguro, confidencial, transparente, metódico y dinámico.

Además, se proponen algunas áreas en donde los estudiantes de pregrado pueden incursionar con proyectos de investigación innovadores que sean la respuesta que instituciones y organizaciones panameñas demandan actualmente.

***Índice de Términos*— Hashing, Colisiones, Token, Encriptación, e-Commerce, e-Voting, SHA, HCEs.**

I. INTRODUCCIÓN

Los métodos de búsqueda se caracterizan por estar sujetos, principalmente, al ordenamiento y a la cantidad de datos que se presenten, lo cual influye significativamente en el tiempo de búsqueda de un elemento específico.

Lo cual significa que, a mayor cantidad de datos, mayor será el tiempo de búsqueda y de igual manera, mayor será el impacto que tenga en el sistema.

Las funciones hash son un método que, a pesar de estar dentro de la categoría de los métodos de búsqueda, presenta una gran independencia en cuanto a la relación tiempo – cantidad de datos, lo cual le permite destacar de entre los demás métodos de búsqueda existentes. De un modo ampliado, hay que destacar la variada gama de métodos de los cuales se dispone para el manejo de errores que en ocasiones se suelen presentar durante la búsqueda de direcciones o rutas. Errores que comúnmente se les conoce como colisiones ya que, de modo figurativo, son el resultado del choque de dos claves en una misma ruta.

Aplicado a los distintos sectores de la sociedad; este método contribuye significativamente en la búsqueda de mayor eficiencia y eficacia de los distintos sistemas responsables del manejo, control, administración y seguridad de los datos y/o de la información de los usuarios o empresas. Como lo son los algoritmos hash de firmas digitales y de función criptográfica [1].

La intención inicial de este artículo es expandir la comprensión sobre las funciones Hash y el contexto en donde se aplica, descubriendo que son múltiples los ámbitos en donde impacta esta tecnología, sobre todo aquellos relacionados con la integridad de la información. Para ello se definen conceptos de transformación de llaves o hash y se presentan las técnicas de cálculo de direcciones y los métodos para el manejo de errores o colisiones.

Luego contextualizamos los aportes del hashing en la cotidianidad de actividades, de modo que sea el primer eslabón de estudios tendientes a abrir el panorama académico en pregrado para el aprendizaje, mejora y desarrollo de nuevas aplicaciones del hashing.

Finalmente surge la inquietud ¿cómo podemos contribuir en esta área? Es así como se analiza la temática con el fin de encontrar vacíos u oportunidades de mejoras en los métodos que se desarrollan en el artículo, presentando nuevas ideas orientadas a robustecer la seguridad informática.

Las secciones del artículo se detallan a continuación, iniciando con una breve definición del concepto de

transformación de llaves (funciones hash), luego exponemos las técnicas que se utilizan para el cálculo de direcciones y los métodos para el manejo de errores que puedan aparecer durante el proceso, posteriormente mostramos cómo estas funciones están presentes en el diario hacer del hombre y en los distintos sectores de la sociedad, y finalmente las conclusiones del trabajo.

II. Comprendiendo el concepto de Hash

Para tener acceso a los datos dentro de un arreglo el mismo debe estar ordenado y dependiendo de la cantidad de datos así mismo será proporcional el tiempo de búsqueda, lo cual significa que se hace imperante el uso de un método que permita el acceso a estos datos acortando tiempo y restándole importancia al ordenamiento de ellos.

La función hash, también conocido como hashing o transformación de llaves, es un método que permite el acceso a estos datos sin que los mismos estén ordenados, lo cual aumenta la velocidad de búsqueda reduciendo el tiempo de espera significativamente.

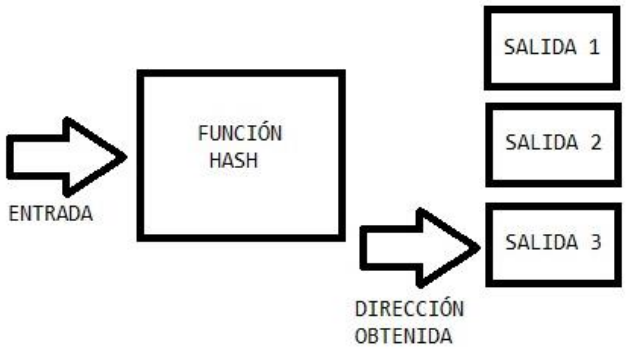

Fig. 1 Las funciones hash. Una vez que la función hash procese los datos de entrada nos dará una ruta directa a los datos de salida.

Dicha diferencia, de entre los demás métodos de búsqueda existentes, radica en la capacidad que posee de establecer una ruta o dirección a través de la previa asignación de un índice para ser ejecutado luego dentro de un arreglo permitiendo el acceso de forma directa al elemento sin la necesidad de un ordenamiento o de un tamaño de datos especifico.

Teniendo como principal objetivo el de convertir un mensaje de longitud variable en un valor de longitud fija denominado código de hash. Este valor se obtiene en función de todos los bits del mensaje y tiene la capacidad de detectar errores.

Los cuales se pueden producir o generar si durante la ejecución se obtienen varias direcciones o rutas para un mismo elemento de entrada.

III. Técnicas de cálculo de direcciones

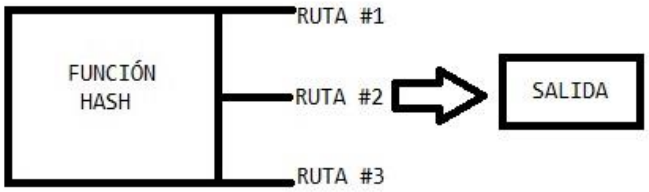

Fig. 2 Calculando direcciones. Las técnicas de cálculo de direcciones brindan la ruta apropiada a seguir optimizando el tiempo y minimizando la sobrecarga del sistema.

La ruta o dirección a utilizar se consigue a través del uso de técnicas de cálculo de direcciones como lo son:

- Hashing por residuo
- Hashing por cuadrado medio
- Hashing por pliegue [2]

3.1. Hashing por residuo: Esta función toma el residuo resultante de la división entre el total de elementos del arreglo N y la clave K a convertir, al resultado obtenido se le suma 1 para una mayor precisión de la posición buscada. Entonces la misma quedará definida de la siguiente forma:

$$H(k) = (K \bmod N) + 1 \qquad (1)$$

Para el uso de esta técnica por medio del residuo se recomienda usar números primos o divisibles entre muy pocos números de lo contrario si N no es un número primo se tomará el valor primo más cercano [3].

3.2. Hashing por cuadrado medio: El segundo método a estudiar consiste en elevar al cuadrado la clave a convertir en dirección.

De tal modo que:

$$H(k) = \text{dígitos\_centrales } (K2) + 1 \qquad (2)$$

En base a que la cantidad de elementos en el arreglo corre de 1 a 100, se obtienen los dígitos centrales y entonces se le suma 1 [3].

3.3. Hashing por pliegue: La función por pliegue consiste en separar la clave en partes iguales, aunque en algunos casos la última parte de esta separación puede variar y tener menos dígitos que las anteriores.

$$\text{O sea, si } K = 7259 \rightarrow 72 + 59$$

Una vez realizada la separación de la clave a convertir, se procede a la suma de estas. Al resultado obtenido se le suma 1 a las dos últimas cifras para finalmente obtener la dirección deseada [3]. Entonces:

$$H(k) = ((A1 + A2) + (B1 + B2)\ldots) + 1 \qquad (3)$$

IV. Métodos para el manejo de colisiones o errores

Como se mencionó con anterioridad, una colisión es el acto resultante que se genera al momento de que dos o más direcciones (claves) coincidan con un mismo elemento o ruta.

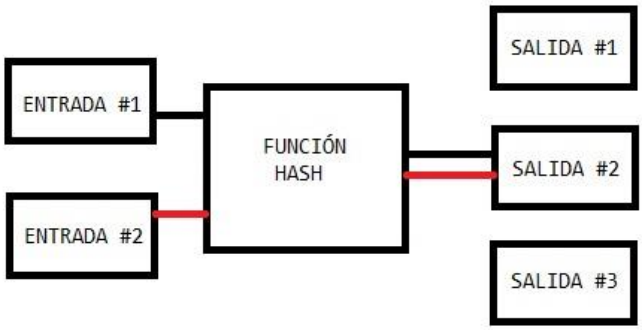

Fig. 3 Colisiones o errores. El resultado de coincidir dos direcciones en una misma salida es llamado colisión.

Al producirse este tipo de situaciones resulta complicado tener que lidiar con ellas, razón por la cual se enfatiza, previamente, en el hecho de elegir de forma apropiada el método o técnica de cálculo de direcciones correcto a utilizar con el fin de reducir en la mayor cantidad posible este tipo de errores.

De lo contrario se tiene que optar por el uso de otras alternativas o métodos a disposición, como lo son:

- Reasignación
- Arreglos anidados
- Encadenamiento [3], [4].

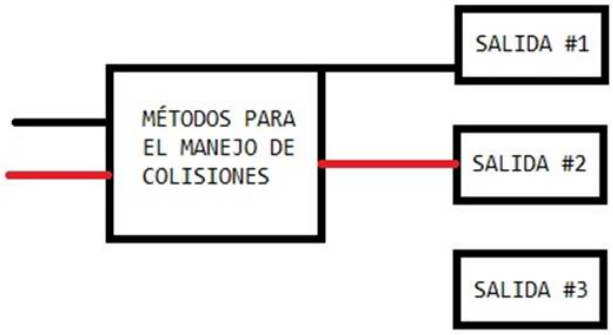

Fig. 4 Manipulando colisiones. Los métodos para el manejo de colisiones manipulan estos errores a fin de producir nuevas direcciones.

4.1. Reasignación: Dentro del apartado de reasignación, que básicamente se define como aquel método de comparación y reubicación de elementos dentro de un arreglo, se analizarán tres alternativas a disposición:

- Prueba lineal
- Prueba cuadrática
- Doble dirección lineal

4.1.1. Prueba lineal: Una vez se suscite una colisión, se da inicio a la prueba lineal que consiste en recorrer al arreglo, como estructura circular (la posición siguiente al último será la primera), de forma secuencial teniendo como punto de partida a la colisión y terminando al momento de encontrar un espacio vacío o al momento de ubicar al elemento.

Cabe señalar que esta alternativa para el manejo de colisiones se ve afectada en aquellas situaciones donde no haya un equilibrio dentro del arreglo, o sea, en las situaciones donde halla fuerte agrupamiento alrededor de las claves o, en contra parte, donde exista exceso de zonas vacías dentro del arreglo ya que la búsqueda del elemento dentro del arreglo no sería directa sino más bien secuencial.

4.1.2. Prueba cuadrática: Esta prueba es similar a la lineal, a diferencia que en esta las direcciones alternativas se generaran en intervalos diferentes hasta i2 siendo i = posición. Lo cual ayuda a una mejor distribución de las claves o elementos colisionados.

Este tipo de distribución representa, igualmente, la mayor desventaja para esta prueba ya que al distribuir de esta manera e iniciar el recorrido del arreglo pueden quedar casillas del mismo sin visitar.

4.1.3. Doble dirección lineal: Este tipo de método consiste en usar de forma repetitiva las técnicas de obtención de dirección, una vez detectada una colisión, con el fin de obtener una nueva dirección.

El proceso finalizará, al igual que en la prueba lineal, cuando el elemento haya sido ubicado a través de la(s) nuevas direcciones o cuando se encuentre una posición vacía en el arreglo.

4.2. Arreglos anidados: Esta opción para el manejo de colisiones consiste en la creación de un segundo arreglo en el cual solo se almacenan las colisiones que se produzcan en el arreglo original.

A pesar de su simplicidad, estudios previos han hallado que este método para el manejo de colisiones resulta ineficiente; la creación del segundo arreglo genera un conflicto de equilibrio entre el uso o costo de memoria y la cantidad de variables colisionadas a captar debido a ser incierto el tamaño adecuado a usar para este nuevo arreglo.

4.3. Encadenamiento: A pesar de ser el último método a estudiar es el que goza de mayor aceptación para el manejo de colisiones debido a su dinamismo.

Consiste en una lista ligada a un arreglo, donde cada elemento estará direccionado hacia la lista mediante un apuntador; de modo tal que cada vez que se genera una colisión,

esta se almacenará en la lista de forma sucesiva conforme vayan apareciendo.

Su principal desventaja radica en que ocupa espacio adicional, además exige el manejo de listas ligadas y, si la lista aumenta considerablemente, se pierde la facilidad de acceso que provee el método hash.

Cada una de las técnicas y métodos para el manejo de colisiones posee sus ventajas y, propiamente, sus desventajas; lo cual precisa un breve estudio sobre cuál de ellas escoger por sobre las demás a fin de obtener mayores resultados con el menor grado de errores posibles y con el mejor manejo apropiado de estos.

Al considerarse paralelamente ambos procesos, cálculo de direcciones y manejo de errores (colisiones), resulta imprescindible el optar por aquellos que muestren una diferencia significativa entre ventajas y desventajas. Aunque hay que destacar que ambas técnicas se complementan a fin de priorizar los tiempos de ejecución y el impacto o costo que sufrirá el sistema.

Debemos recordar que, para obtener los resultados deseados, sólo una parte le compete a la técnica, método o algoritmo hash; ya que el mismo se complementa con la parte humana y como éste lo implementa en busca de satisfacer necesidades en el ámbito social, laboral o personal.

V. Aportes del Hash al contexto real

Al pasar de los años se han presentado cambios radicales en los procesos mediante el cual la sociedad interactúa con el ambiente que lo rodea. Cambios que están dejando atrás al clásico papeleo y paulatinamente se han ido sustituyendo por procesos donde su principal característica es la unión entre software y hardware.

Algunos de índole pública, otros de índole privada y otros con características restrictivas más allá de los aspectos de seguridad conocidos. Independientemente del perfil que posea, queda claro que dentro de su composición algorítmica se debió tomar en cuenta la seguridad e integridad tanto del usuario, los datos, así como del proceso de envío y recibido de información.

En la actualidad se posee varios métodos de seguridad que garantizan la privacidad de la información, como lo son: contraseñas (password), firmas digitales, tokens, lector de huellas digitales, verificación de voz o face ID como actualmente se encuentran presente en algunos teléfonos móviles.

Entre el ingreso del password y el acceso al sistema siempre se ha llevado a cabo un proceso que sigilosamente ha velado por nuestra seguridad, tanto a la entrada del sistema como en las acciones que realicemos durante el tiempo que nos encontremos en el mismo. Este proceso cauteloso se conoce como funciones hash o hashing, que como se mencionó anteriormente, es un método que permite el acceso a un grupo de datos mediante cálculos sencillos haciendo uso de claves de entrada para luego obtener una ruta o dirección.

Estas funciones se encuentran presentes en telefonía móvil, e-Commerce, mensajería (e-mail), encriptación, firmas digitales, validación y verificación de usuarios/personal, finanzas, transferencias bancarias, e-Voting, medicina, entre otros muchos sectores en los cuales se encuentra presente.

Cabe señalar que en términos de vulnerabilidad las funciones hash no quedan exentas ya que sufren de ataques de colisiones contra, por ejemplo, el algoritmo sha-1 que es un algoritmo destinado al control de la integridad de datos financieros, como en [5]. Otro de los problemas muy comunes que se pueden presentar son los ataques a las firmas digitales (a pesar de que son muy utilizadas), ya que para verificar la veracidad de la información y del individuo las mismas deben ser certificadas por un tercero.

En la actualidad, las colisiones pueden ser aprovechadas por un individuo de esta forma: Se crean dos mensajes diferentes, pero se logra que ambos tengan el mismo resultado hash (colisión legítima). Se le envía uno de los mensajes al receptor y se consigue que él lo acepte y lo firme digitalmente. Como la persona lo que está firmando es el resultado hash entonces sería como que firmara ambos mensajes; luego se le presenta a esta persona el segundo mensaje y de esta forma se engaña legítimamente [5].

A pesar de la existencia de estos ataques en busca de quebrantar la seguridad de la información/datos de segundos o terceros las funciones hash aún son muy utilizadas. Los algoritmos hash más usados actualmente en la sociedad podemos mencionar:

- MD5 (Message-Digest Algorithm 5 o Algoritmo de Firma de Mensajes 5), realiza un procesamiento en bloques de 512 bits, generando números de 128 bits. Usado aun en comprobación de ficheros.
- SHA-1 (Secure Hash Algorithm 1 o Algoritmo de Hash Seguro 1), toma como entrada mensajes con longitud máxima de 264 bits y produce un número de 160 bits.

La familia SHA es un conjunto de funciones resumen (SHA-1, SHA-256, SHA-512) desarrolladas por la Agencia de Seguridad Nacional de los EEUU (NSA).

- DSA (Digital Signature Algorithm), es el estándar de United States Federal Government para firma digital. Es un algoritmo exclusivo de firma electrónica basado en clave pública, pero no vale para comunicaciones confidenciales.
- RJPEMD-160, es un algoritmo del resumen del mensaje de 160 bits y función criptográfica de hash. Es una versión mejorada de RIPEMD, que estaba basado sobre los principios del diseño del algoritmo MD4, y es similar en seguridad y funcionamiento al más popular SHA-1 [6].

No se encontró ningún argumento significativo que haga pensar que a pesar de los ataques de colisiones que algunos miembros de la familia SHA sufren constantemente estos dejen de ser confiables.

El sha-1 sigue siendo muy usado al igual que el MD5 por su capacidad de traducir la información de salida a una longitud de bits relativamente corta a diferencia de la longitud de la información de entrada, como en la Tabla 1.

TABLA 1.
LONGITUD DE SALIDA DE LAS FUNCIONES HASH

Función	Longitud (bits)
MD5	128
SHA-1	160
SHA-256	256
SHA-512	512

La integridad y privacidad son dos conceptos que tienen prioridad ante las funciones hash; en el campo de la medicina es muy utilizado para aumentar la seguridad de los HCE (Historial Clínico Electrónico) a fin de revelar la información sanitaria del paciente más no la información personal del mismo; información que será mostrada mediante un identificador asociado a una función hash en vez de mostrar directamente los datos o información del paciente. Una función hash asegura que sea muy difícil obtener el identificador del paciente por cualquier persona que no esté autorizada [7].

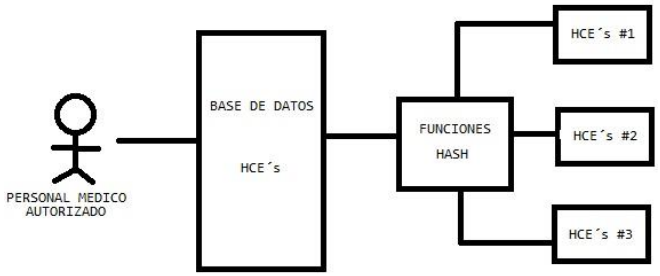

Fig. 5 Protección de la información. Las funciones hash garantizan que sólo el personal autorizado pueda tener acceso al historial médico de los pacientes.

Trascendiendo a la medicina podemos observar su uso en el **e-Commerce** al momento de realizar transferencias, pagos o compras siendo un tarjetahabiente. En el e-Commerce como se menciona en la referencia para asegurar la integridad de la información o de los mensajes no es suficiente el proceso de encriptación ya que un intermediario podría interceptar el mensaje, conseguir la clave pública del receptor y crear un nuevo mensaje con el mismo nombre que el mensaje original. Para evitar esto se aplica una función hash que se envía con el propio mensaje, de forma que al recibirlo el receptor aplica la función hash al mensaje y comprueba que sea igual al hash que se envió antes. Garantizando la seguridad y los sistemas de pago mediante banca en línea [8].

En la actualidad los bancos, por ejemplo, el **Banco General de Panamá** hace uso de los tokens que apoyados en las funciones hash garantizan un mayor nivel de seguridad a sus clientes. En dichos sistemas un usuario recibe un tokens que le permite acceder a cierto recurso o servicio; siendo el caso de realizar pagos, el uso de funciones hash permite agilizar la emisión de los mismos al sustituir operaciones criptográficas costosas por cálculos sencillos [9].

La integridad es una de las características más importantes en los procesos de **votación electrónica**, ya que lo primero que deben legitimar las máquinas es que el voto permanezca inalterado desde su ingreso hasta el último de los recuentos que se efectúan [10].

Ahora, partiendo del principio de que los sistemas de **e-Democracy** tienen una base fundamental desde los sistemas de e-Voting y que estos deben adaptarse a la sociedad en cada contexto, llámense ciudadanos, funcionarios de una organización, estudiantes, profesores o cualquier comunidad democrática; el despliegue exitoso de estos sistemas debe superar una problemática relacionada con la simplicidad y facilidad del proceso para el usuario final, así como las garantías que debe brindar en lo referente a la autenticación de los votantes, emisión de votos, y publicación de resultados; garantizando el anonimato, evitando la votación por parte de votantes no autorizados o que ya lo hayan hecho y en el recuento correcto de los votos cuando se requiera.

Conseguir el cumplimiento de las garantías que ofrece el método electoral tradicional, desde la implementación tecnológica se hizo necesario adoptar una serie de mecanismos tecnológicos de seguridad, dentro de los cuales se utilizaron las firmas y los certificados digitales, que apoyados en la criptografía y funciones de hashing, son capaces de proveer los servicios de seguridad necesarios para proteger los sistemas de **e-Voting**.

Fig. 6 Las funciones hash presentes en los procesos electorales. Las funciones hash garantizan la seguridad del debido proceso en el e-Voting.

En los sistemas de **votos telemáticos** (democracia digital) en instituciones educativas; cuando el votante se ubica en la cabina de votación, debe suministrar su carnet (a través de alguno de los lectores habilitados) y posteriormente la contraseña (digitación manual en teclado en pantalla), seguido de la lectura de huella dactilar. Éste último paso hace que se active el envío de Id capturado por el lector biométrico y el hash del password

digitado (generado con SHA-512), mediante una petición al servidor para verificar que el votante está registrado y puede participar en la elección.

Fig. 7 Verificación de datos de identidad. El servidor electoral verifica si los datos ingresados por el votante son correctos mediante el uso del SHA-512.

El servidor compara el hash de pass recibido con el hash de pass retornado por la base de datos; tras haber consultado si existe el Id recibido; y responde al cliente enviando la huella dactilar almacenada en la base de datos, la cual se encuentra serializada. El cliente procede a hacer la comparación de la huella dactilar contra la huella dactilar almacenada en el servidor o base de datos y envía una nueva petición para obtener los candidatos de la elección.

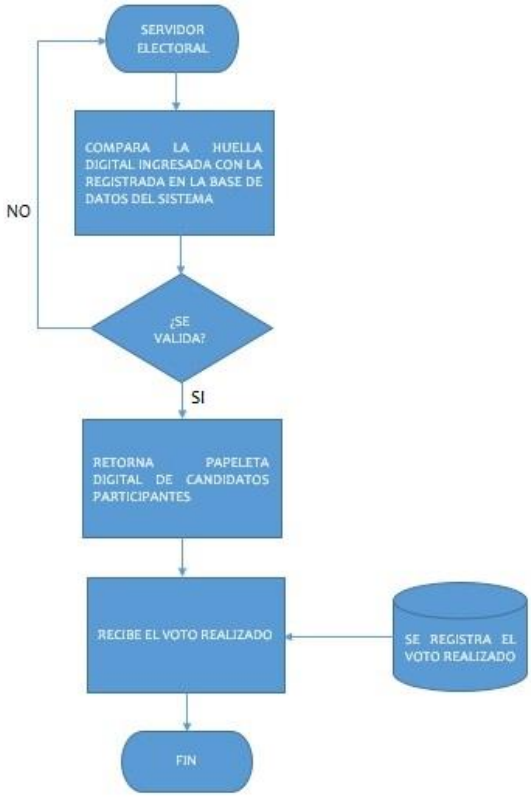

Fig. 8 Diagrama de flujo del proceso de votación. El sistema se encarga de validar la información registrada en el servidor con la captada para de ese modo llevar a cabo la votación.

Una vez efectuado el proceso de autenticación, se realiza el proceso de votación, brindando al usuario los tarjetones asociados a su perfil de votante. Por lo cual se hace una carga aleatoria de los candidatos y se almacenan en la base de datos, equivalente a la urna, la cual cifra los datos de los resultados, hasta que la autoridad electoral, proceda a ingresar la clave para el descifrado, previa validación automática de la marca de tiempo, que indique el fin de la jornada [11].

A. ¿Y en Panamá cómo se aplican los algoritmos hash?

Es un hecho que instituciones y empresas ya están trabajando con esta tecnología. Lo vemos en el uso de firmas digitales en las universidades. Transacciones a través de Bitcoin es una realidad pudiendo adquirir algunos productos y servicios en Panamá, como por ejemplo de Zaza, El Apartamento, Irepair y algunos locales en Balboa Boutiques.
Corporaciones como COOPEDUC, por mencionar una, ya ofrecen servicios a través de tarjetas de trasferencia segura, las cuales usan mecanismos de encriptación que integran las tecnologías de hash más contemporáneas.

El impacto y aporte de los métodos planteados en este documento trasciende sectores y niveles de la sociedad, ya que las funciones hash se encuentran presentes en cajeros automáticos, dispositivos de verificación policial, sitios web o consolas de video juegos donde su principal objetivo es evitar colisiones, es decir, evitar que dos cadenas distintas tengan asociadas el mismo código o clave.

B. ¿Puede un estudiante de pregrado aportar en este campo?

La respuesta es afirmativa, puede hacerse más a nivel de investigación.

- Análisis Comparativo en base a criterios de vulnerabilidad, consumo de recursos informáticos, velocidad, eficiencia de los distintos métodos de generación de llaves y manejo de colisiones.
- Proponer modelos algorítmicos tendientes a optimizar el tratamiento de colisiones, a través de diferentes estructuras de datos.
- Desarrollo de plataforma Bitcoin como solución segura a las transacciones electrónicas, mediante la encriptación del pago.
- Incursionar en el desarrollo de aplicaciones que fortalezcan el voto electrónico en nuestro país.

VI. CONCLUSIONES

La transformación de llaves, función hash o hashing es un proceso de búsqueda que brinda la posibilidad de obtener una ruta hacia un elemento específico dentro de un arreglo de “n” cantidades de elementos, aportando grandes beneficios en cuanto al costo de uso de memoria y al tiempo de búsqueda.

La disponibilidad de técnicas para dar con esta ruta es amplia al igual que las alternativas para el manejo de colisiones. Por lo tanto, es recomendable saber elegirlas apropiadamente.

Queda claro a partir del presente estudio que los aportes que cada una de estas técnicas ofrece son exclusivas de ellas mismas, pero no debemos descartar que en su contraposición poseen desventajas que dependiendo de la situación puede dar cabida a una acción secundaria: ya sea en el manejo de colisiones, en la obtención de direcciones o en la facilidad de acceso que provea cualquiera de las tres técnicas.

Las funciones hash revisten particular importancia en entornos donde la seguridad, integridad y privacidad de la información es prioritaria, y la implementación de mecanismos de control de acceso es inminente.

Panamá cumple con las condiciones propicias para explotar estas técnicas y métodos, pues siendo el Host de las Américas, con uno de los centros bancarios más importante de la región, con una plataforma abierta al marketing, se hace imprescindible el desarrollo de aplicaciones tecnológicas robustas que impliquen el uso de técnicas y métodos de hashing.

Es necesario avocarse a la búsqueda de nuevas estrategias de investigación en esta área, descubrir qué elementos de hardware y software están surgiendo para desarrollo de proyectos novedosos que demanden seguridad en el intercambio de información, integridad de datos a través de encriptación, transacciones sobre la base de Bitcoin, cifrado electrónico y otros que pueden ser propuestas interesantes de solventar.

Estando presente, activo y siendo constantemente aplicado dentro de los procesos sistematizados de la sociedad causa gran estruendo dentro de un mundo digital silencioso al cual no podemos ver pero que está involucrado en todas las actividades del hombre proveyéndolo de facilidades en su diario vivir.

REFERENCIAS

[1] Luis J. Aguilar, Ignacio Z. Martinez, Estructura de Datos en Java, 2nd ed., McGraw-Hill, España, 2008.

[2] Luis J. Aguilar, Fundamentos de Programación: Algoritmos y Estructura de Datos, 2nd ed., McGraw-Hill, España, 1996.

[3] Osvaldo Cairó, Silvia Guardati, Estructura de Datos, 2nd ed, McGraw-Hill, México, 2002.

[4] Mary E. S. Loomis, Estructura de Datos y Organización de Archivos, 2nd ed., Prentice-Hall, México, 1991.

[5] "Colisiones en el algoritmo de ciframiento SHA-1" [Online]. Available: http://181.49.226.34:8090/revistas/index.php/gd/article/view/55 [Accessed: 19-En-2018].

[6] "Algoritmos HASH y vulnerabilidad a ataques" [Online]. Available: http://www.revistasbolivianas.org.bo/scielo.php?pid=S1997-40442009000200026&script=sci_arttext&tlng=es [Accessed: 29-May-2017].

[7] I. Carrión S., A. Toval A., J. L. Fernández A. and P. A. Oliver L. "Seguridad y Privacidad en Historiales Clínicos Electrónicos: una Revisión Sistemática de la Literatura", Revista Salud, Vol.7, N°26, 2011.

[8] "Sistemas de pago seguro. Seguridad en el comercio electrónico" [Online]. Available: http://revistaselectronicas.ujaen.es/index.php/REE/article/view/359 [Accessed: 19-En--2018].

[9] "CADAT: Control de acceso basado en tokens y cadenas hash delegables" [Online]. Available: http://www.revistasbolivianas.org.bo/scielo.php?pid=S1997-40442013000100025&script=sci_arttext&tlng=es [Accessed: 10-En-2018].

[10] "Método de búsqueda por funciones de Hash" [Online]. Available: https://estructuradedatositp.wikispaces.com/6.3.+M%C3%A9todo+de+b%C3%BAsqueda+POR+FUNCIONES+DE+HASH [Accessed: 16-May-2017].

[11] "SISTEMA DE VOTO TELEMÁTICO CON URNA ELECTRONICA PARA IMPLEMENTAR DEMOCRACIA DIGITAL EN INSTITUCIONES EDUCATIVAS" [Online]. Available: http://fcbi.unillanos.edu.co/cici/Articulos/CICI_2016_paper_167.pdf [Accessed: 27-May-2017].

[12] "Tablas de Dispersión" [Online]. Available: http://btocastro.blogspot.com/2011/07/tablas-de-dispersion.html [Accessed: 17-May-2017].

[13] "Funciones Hash ó Hashing" [Online]. Available: http://www.itnuevolaredo.edu.mx/takeyas/apuntes/Administracion_Archivos/Apuntes/Hashing.PDF [Accessed: 16-May-2017].

[14] "Sistema de eVote: Verificabilidad del voto electrónico" [Online]. Available: http://materias.fi.uba.ar/7500/obremski-tesisdegradoingenieriainformatica.pdf [Accessed: 19-Jun-2017].

[15] "Diseño e implementación de un sistema de voto electrónico" [Online]. Available: https://riunet.upv.es/bitstream/handle/10251/69228/MORENO%20-%20Dise%c3%b1o%20e%20implementaci%c3%b3n%20de%20un%20sistema%20de%20voto%20electr%c3%b3nico.pdf?sequence=1&isAllowed=y [Accessed: 19-Jun-2017].

[16] "Tablas HASH" [Online]. Available: http://decsai.ugr.es/~jfv/ed1/tedi/cdrom/docs/tablash.html [Accessed: 16-May-2017].

[17] "Algoritmos de búsqueda y ordenamiento" [Online]. Available: https://es.slideshare.net/Juaniito1/informe-analisis-de-algoritmos-33620953 [Accessed: 17-May-2017].