La seguridad de la firma electrónica con el estándar criptográfico algoritmo de hash seguro 3 (SHA-3)

Luis Miguel Alamilla Hernández, Alejandro Hernández Cadena, José Ney Garrido Vázquez, José Ángel Jesús Magaña, José Manuel Gómez Zea

[1]Tecnológico Nacional de México/Instituto Tecnológico de Villahermosa, Computación y Sistemas, Villahermosa, México

Resumen

La firma electrónica en México está implementada principalmente por el Servicio de Administración Tributaria (SAT), así como también otras dependencias del gobierno federal. En 2019 alrededor de 9.6 millones de personas contaban con una firma electrónica. En otras palabras, el 7.7% de la población cuenta con firma electrónica donde actualmente está soportado por la infraestructura de clave pública (PKI por sus siglas en ingles), que en conjunto con el estándar criptográfico SHA (Secure Hash Algorithm por sus siglas en ingles), forman la firma electrónica avanzada. Dicha firma tiene el propósito de identificar al emisor de un mensaje como el autor legítimo de este, tal como si se tratara de la firma autógrafa del emisor, en términos de medios electrónicos con seguridad técnica y jurídica. En lo que concierne a SHA como estándar para el resumen de datos de longitud fija, tenemos que en sus dos primeras versiones SHA-0 y SHA-1 son algoritmos vulnerables a colisiones y pre-imagen el último vulnerado a principios del 2016.

En la actualidad el estándar criptográfico de resumen hash vigente es SHA-2 y sus diversas versiones, ya se han hecho diversas pruebas para comprobar su seguridad y sigue mostrándose fuerte. No obstante, el instituto Nacional de Estándares y Tecnología (NIST por sus siglas en ingles) el 23 de enero de 2007 lanza una convocatoria para decidir el nuevo algoritmo que llevaría por nombre SHA-3, dicho lo anterior en octubre de 2012 algoritmo Keccak gana el concurso y es declarado estándar a nivel mundial en agosto de 2015.

SHA-3 es el último de la familia del estándar SHA, pero esto no significa que SHA-2 quedo en el pasado es todo lo contrario porque oficial y científicamente sigue siendo bueno, en el momento que sea vulnerado se puede estar seguro de que se tiene a SHA-3 como respaldo y estándar criptográfico. SHA-3 se diferencia mucho a sus versiones anteriores ya que no sigue los mismos principios, su estructura es totalmente diferente y se basa en la estrategia de esponjas herméticas, que en cuanto a seguridad es muy robusto, pero en software demuestra procesar el algoritmo con el doble de tiempo que SHA-2.

Abstract

The electronic signature in Mexico is mainly implemented by the Tax Administration Service (SAT), as well as other federal government agencies. In 2019, around 9.6 million people had an electronic signature. In other words, 7.7% of the population has an electronic signature which is currently supported by public key infrastructure (PKI), which together with the cryptographic standard SHA (Secure Hash Algorithm), form the advanced electronic signature. The purpose of this signature is to identify the sender of a message as the legitimate author of that message, just as if it were the sender's own signature, in terms of electronic means with technical and legal security. Regarding SHA as a standard for the summary of fixed-length data, we have that in its first two versions SHA-0 and SHA-1 are algorithms vulnerable to collisions and pre-image the last one violated at the beginning of 2016.

At present the current hash summary cryptographic standard is SHA-2 and its various versions have already been tested for safety and it is still strong. However, the National Institute of Standards and Technology (NIST) on January 23, 2007, launched a call to decide the new algorithm that would be named SHA-3, said the above in October 2012 Keccak algorithm wins the competition and is declared standard worldwide in August 2015.

SHA-3 is the last one of the SHA standard family, but this does not mean that SHA-2 is in the past, on the contrary, it is officially and scientifically still good, at the moment that it is violated you can be sure that you have SHA-3 as cryptographic support and standard. SHA-3 is very different from its previous versions because it does not follow the same principles, its structure is totally different and it is based on the strategy of hermetic sponges, which in terms of security is very robust, but in software, it demonstrates to process the algorithm with twice the time than SHA-2

Palabras clave: SHA, SHA-3, firma electrónica, criptográfico, estándar, vulnerable, algoritmo.
Keywords: SHA, SHA-3, electronic signature, cryptographic, standard, vulnerable, algorithm.

1. INTRODUCCIÓN

La firma digital es una firma electrónica que se usa para autenticar la identidad de quien envía un mensaje o quien firma un documento electrónico.

La utilización de hojas papel para apoyo de información en trámites y procesos exige disponer de espacio físico para el documento, a la vez que vuelve inútil su transformación. Hoy en día, las tecnologías de la información y comunicaciones (TICS) posibilita cambiar la información de papel a diversos medios digitales.

Digitalizar un documento en papel resulta en gran medida un ahorro considerable de costos para una empresa, ya que, enviando a través de medios electrónicos, como, por ejemplo: el correo electrónico, agiliza considerablemente el envío y recepción del documento.

En el caso de archivos o documentos firmados con firma autógrafa al momento de ser digitalizados, pierden exponencial mente su valor legal ya que, durante el proceso de la firma autógrafa, originalmente efectuada de puño y letra, pudo ser editada, alterada, reemplazada o borrada por agentes externos.

El estándar criptográfico RSA permite hacer uso de su método para la generación de firmas electrónicas avanzada, ya que con RSA se puede dar soporte para la generación del par de claves que se necesitan para la generación de la firma electrónica avanzada.

Asumiendo que generamos el par de llaves las cuales serán los medios para cifrar y descifrar mensajes, no está del todo correcta ya que necesita de la infraestructura PKI (Public Key Infrastructure) para tener un soporte lo sufrientemente valido para tener una firma electrónica avanzada.

2. ESTRUCTURA RSA Y PKI

En febrero de 1978, 3 investigadores americanos del instituto tecnológico de Massachusetts patentan el algoritmo RSA por las iniciales del apellido de cada investigador.

RSA es un algoritmo asimétrico y sus siglas provienen de sus tres inventores: Rivest, Shamir y Adleman. Se basa en la dificultad para factorizar números grandes, así se calcula las claves a partir de un número que se obtiene como producto de dos números primos grandes.

El algoritmo RSA basa su fortaleza en la dificultad de factorizar un numero compuesto muy grande producto de dos números primos grandes, lo cual es un problema muy grande para la capacidad computacional.

No fueron fáciles los años de RSA ya que nadie creía su utilidad sin embargo el tiempo fue dándole la razón a sus inventores y finalmente se convirtió en un estándar.

2.1 ALGORITMO PARA LA GENERACIÓN DE LLAVES RSA

Se determinan dos usuarios A y B, cada uno elige un módulo de cifra

$$n = p * q$$

ISSN: 2007-4786

Siendo p y q primos de un tamaño igual o superior de 1024 bits.

Los valores de los primos p y q serán un secreto, solo conocido por el propietario de esa clave
En el caso de A ese modulo será:

$$nA = pA * qA$$

Para el caso de B el modulo será

$$NB = pB * qB$$

Para cada usuario se calcula el indicador de Euler Φ del módulo n que dando de la siguiente manera:

$$\Phi n = (p-1) * (q-1)$$

Entonces para el usuario A su indicador de Euler es:

$$\Phi nA = (pA-1) * (qA-1)$$

y para el usuario B:

$$\Phi nB = (pB-1) * (qB-1)$$

Ambos serán números secretos muy grandes conocidos también como trampa.

Cada usuario utiliza un valor de llave publica

$$1 < e < \Phi(n)$$

Para asegurarse que exista el inverso multiplicativo y por tanto la existencia de la clave privada inversa de la clave pública debe cumplirse:

$$inv(e, \Phi(n)) \longrightarrow mcd\,[e, \Phi(n)] = 1$$

Usando el algoritmo extendido de Euclides, se procede a calcular de cada usuario la clave privada d, quedando para el usuario A:

$$dA = inv(eA, \Phi(nA))$$

y para B:

$$dB = inv(eB, \Phi(nB))$$

Los usuarios A y B hacen publico el cuerpo o módulo de cifra n y su clave pública e y se guarda en secreto la llave privada d, además se guarda en secreto los primos p y q los cuales sirven para acelerar el descifrado mediante el teorema chino de los restos.

ISSN: 2007-4786
Innovación y Desarrollo Tecnológico
Volumen 12 – Número 2
Abril – Junio 2020

2.2 CIFRANDO Y FIRMANDO CON RSA UN MENSAJE

Si las claves públicas y privadas son las que se indican se puede realizar las siguientes operaciones criptográficas: primero: Enviar el numero N calculando

$$Neb\ mod\ nB\ =\ C \text{ (cifrado)}$$
$$CdB\ mod\ nB\ =\ N \text{ (descifrado)}$$

2.3 PUBLIC KEY INFRASTRUCTURE (PKI)

Public Key Infrastructure (PKI) por sus siglas en inglés, define una jerarquía o una estructura de confianza entre una entidad que requiera ser conocida y tenga la necesidad de tener su firma en internet y un usuario, entonces básicamente el núcleo de todas las comunicaciones que utilizan PKI es establecer una identidad ya sea de la persona que está ingresando a un sitio al que solo debe acceder o un sitio que se está mostrando al mundo, pero la idea es siempre establecer una identidad.

Que podemos hacer con PKI:

- Certificar usuarios, computadoras y servidores.
- Solucionar problemas generando certificados para equipos específico: verificando al cliente que sea quien dice ser y el equipo valide al servidor sea quien dice ser.

Funciones de PKI: Mantener, distribuir, validar y revocar certificados SSL/TLS.

PKI encierra ciertos conceptos por su esencia que se considera prudente definirlos:

Protocolo SSL/TLS
El principal objetivo del protocolo SSL es proveer privacidad y confiabilidad entre dos aplicaciones que se comunican. El protocolo está compuesto por dos capas: El protocolo de registro SSL y el protocolo de Handshake SSL. El protocolo de registro SSL se encarga de encapsular otros protocolos a más alto nivel. Por otro lado, el protocolo SSL Handshake permite tanto al cliente como al servidor autenticarse e intercambiar un algoritmo de encriptación y llaves criptográficas antes de que el protocolo de aplicación reciba o transmita cualquier bit de datos [1].

Certificado
Los certificados digitales son el equivalente digital del DNI, en lo que a la autentificación de individuos se refiere, ya que permiten que un individuo demuestre que es quien dice ser, es decir, que está en posesión de la clave secreta asociada a su certificado [2].

La conexión entre la clave pública de una autoridad certificadora y uno o varios de sus atributos referenciados a su identidad son características importantes de un certificado de clave pública. Dicho certificado avala que una clave pública corresponde a la autoridad certificadora reconocida y que la autoridad conoce la clave privada.

Para que los certificados digitales sean útiles debe existe una autoridad o entidad certificadora que los garantice, asumiendo esto, uno mismo también puede auto certificar los certificados, pero no habría ninguna

ISSN: 2007-4786

garantía de que la identidad sea la correcta, y por tanto difícilmente un tercero lo aceptara por no reconocer la procedencia del certificado.

Certificados X.509

El formato de certificados X.509 es un estándar del ITU-T (International Telecommunication Union-Telecommunication Standardization Sector) y el ISO/IEC (International Standards Organization / International Electrotechnical Commission) que se publicó por primera vez en 1988. El formato de la versión 1 fue extendido en 1993 para incluir dos nuevos campos que permiten soportar el control de acceso a directorios [2].

Los elementos del formato de un certificado X.509 v3 son [2]:

- Versión. El campo de versión contiene el número de versión del certificado codificado. Los valores aceptables son 1, 2 y 3.
- Número de serie del certificado. Este campo es un entero asignado por la autoridad certificadora. Cada certificado emitido por una CA debe tener un número de serie único.
- Identificador del algoritmo de firmado. Este campo identifica el algoritmo empleado para firmar el certificado (como por ejemplo el RSA o el DSA).
- Nombre del emisor. Este campo identifica la CA que ha firmado y emitido el certificado.
- Periodo de validez. Este campo indica el periodo de tiempo durante el cual el certificado es válido y la CA está obligada a mantener información sobre el estado de este. El campo consiste en una fecha inicial, la fecha en la que el certificado empieza a ser válido y la fecha después de la cual el certificado deja de serlo.
- Nombre del sujeto. Este campo identifica la identidad cuya clave pública está certificada en el campo siguiente. El nombre debe ser único para cada entidad certificada por una CA dada, aunque puede emitir más de un certificada con el mismo nombre si es para la misma entidad.
- Información de clave pública del sujeto. Este campo contiene la clave pública, sus parámetros y el identificador del algoritmo con el que se emplea la clave.
- identificador único del emisor. Este es un campo opcional que permite reutilizar nombres de emisor.
- Identificador único del sujeto. Este es un campo opcional que permite reutilizar nombres de sujeto [2].

La difusión de las técnicas de clave pública requiere una infraestructura que defina un conjunto de estándares, autoridades de certificación, estructuras entre múltiples CAs, métodos para descubrir y validar rutas de certificación, protocolos operacionales, protocolos de gestión, herramientas que pueden operar entre sí y un marco legislativo [2].

3. USO DE LA FIRMA DIGITAL VS FIRMA TRADICIONAL

La firma digital es una firma electrónica que se usa para autenticar la identidad de quien envía un mensaje o quien firma un documento electrónico.

La utilización de hojas papel para apoyo de información en trámites y procesos exige disponer de espacio físico para el documento, a la vez que vuelve inútil su transformación. Hoy en día, las tecnologías de la información y comunicaciones (TICS) posibilita cambiar la información de papel a diversos medios digitales.

Digitalizar un documento en papel resulta en gran medida un ahorro considerable de costos para una empresa, ya que, enviando a través de medios electrónicos, como, por ejemplo: el correo electrónico, agiliza considerablemente el envío y recepción del documento.

En el caso de archivos o documentos firmados con firma autógrafa al momento de ser digitalizados, pierden exponencial mente su valor legar ya que, durante el proceso de la firma autógrafa, originalmente efectuada de puño y letra, pudo ser editada, alterada, reemplazada o borrada por agentes externos.

Por lo tanto, a continuación, se desglosa los puntos clave que representa el uso de la Firma Electrónica Avanzada a diferencia de la Firma autógrafa:

Eficiencia en los procesos: agiliza el envío y recepción de archivos de una forma sobresaliente. La gran cantidad de papeles utilizados para un solo expediente y el tiempo que demora el expediente en recorrer cada proceso, todo esto dependiendo del flujo que tenga que seguir, es minimizado y organizado.

Reducción de errores: logra una amplia y extensa reducción de errores administrativos producidos durante la manipulación del papel.

Reducción de costos: eliminación del uso de papel relacionado en los procesos y el amplio espacio físico necesario para archivar los documentos.

En términos prácticos, desde un punto de vista legal, una firma digital provee una solución viable para contar con archivos o documentos electrónicos con una validez jurídica. Parecido al método de firma basada en papel y tinta, la firma digital agrega al documento digital la identidad del firmante. Sin embargo, a diferencia de la firma autógrafa, es considerado imposible falsificar una firma digital en la forma en que si se pudiera falsificar una firma autógrafa. Además, la firma digital asegura que cualquier cambio realizado a los datos firmados no pueden ser detectados.

3.1 IMPLEMENTACIÓN DE LA FIRMA ELECTRÓNICA AVANZADA CON RSA

El estándar criptográfico RSA permite hacer uso de su método para la generación de firmas electrónicas avanzada, ya que con RSA se puede dar soporte para la generación del par de claves que se necesitan para la generación de la firma electrónica avanzada.

Asumiendo que generamos el par de llaves las cuales serán los medios para cifrar y descifrar mensajes, no está del todo correcta ya que necesita de la infraestructura PKI (Public Key Infrastructure) para tener un soporte lo sufrientemente valido para tener una firma electrónica avanzada.

4. ESTÁNDAR CRIPTOGRÁFICO EN MÉXICO

Estados unidos se convirtió en el primer país en desarrollar una Ley en materia de Firma Digital. En mayo, de 1995 fue publicada la denominada “Utah Digital Signature Act”, en el estado de Utah [3].

Hoy en día, muchos países ya cuentan con legislación en términos de Firma Electrónica Avanzada, así como una instancia que se encarga de poner en la práctica dicha ley. En caso de países como Alemania, Bélgica, Dinamarca, Francia, Irlanda, Italia, Portugal entre otros [3].

ISSN: 2007-4786

En México se consideró necesario la aprobación de la Iniciativa de Decreto por que se expide la Ley de Firma electrónica Avanzada, por lo tanto, fue remitida a el senado de la republica el 9 de diciembre de 2010 para que dicha ley fuera aprobada el primero de marzo de 2011.

El diseño de la Firma Electrónica Avanzada se basa en estándares internacionales de infraestructura de claves públicas PKI.

Actualmente en México se utiliza el estándar de algoritmo criptográfico SHA2, donde se encuentra una familia de versiones de este.

Asumiendo esto, para ser más exacto el SHA256 dentro de la familia de SHA2, con RSA para una longitud de 2048 bits para la creación de firma electrónicas.

4.1 REQUERIMIENTOS DE SEGURIDAD PARA LAS HUELLAS DIGITALES

Hasta este punto se ha tocado todo lo referente a funcionamiento y estructura para realizar una firma digital, pero no sobre las huellas digitales y como lograr una que sea única para cada usuario.

4.1.1. HUELLA DIGITAL

Una huella digital es un identificador único para una persona en internet.

Hoy en día existen diferentes formas obtener una huella digital, en este artículo se enfoca especialmente en las funciones de resumen hash o de picadillo, ya que son las más utilizadas y probadas hasta ciertas versiones ser las más seguras.

Con las funciones de resumen Hash se puede obtener una huella digital, pero para realizar esto es necesario que cumplan criptográficamente hablando las siguientes 5 propiedades:

1. Determinista: Las funciones de resumen has deberán ser determinísticas. Lo cual significa que para un conjunto de datos dado se ha de obtener siempre la misma huella hash.
2. Computacionalmente eficiente: Significa que el algoritmo ha de ser computacionalmente rápido para obtener la huella hash.
3. No reversible: Las funciones hash son de un solo camino lo que significa que es computacionalmente difícil obtener el conjunto de datos original a partir de la huella hash.
4. Pequeños cambios en la entrada cambian completamente la huella hash: Sí se cambia en la entrada a lo mucho un bit, la huella hash debe cambiar drásticamente. Así se asegura que los algoritmos sean sensibles a cualquier cambio por el mínimo que sea.
5. Resistencia a colisiones: La resistencia a colisiones menciona que la probabilidad de que dos datos de entrada diferentes generen la misma huella hash deber ser altamente improbable.

4.2 FUNCIÓN DE RESUMEN SHA

La familia SHA (Secure Hash Algorithm, Algoritmo de Hash Seguro), definido como un sistema de funciones criptográficas hash aprobadas por la Agencia Nacional de Seguridad de estados unidos y publicada por el National Institute of Standards and Technology (NIST).

ISSN: 2007-4786

Dentro de la familia SHA el primer miembro fue publicado en 1993 y es el oficialmente llamado SHA. Sin embargo, actualmente y no oficialmente se le conoce como SHA-0 esto para evitar confundir a sus sucesores. Dos años después SHA-1 fue publicado. En la actualidad existen cuatro versiones que se han publicado desde entonces y las diferencias se basan en un diseño modificado y bits de salida incrementados:

- SHA-224
- SHA-256
- SHA-384
- SHA-512

Todos ellos son versiones de SHA-2.

5. RESISTENCIA A COLISIONES EN LAS FUNCIONES HASH

Una colisión en un algoritmo de resumen hash es conseguir que dos entradas de datos diferentes tengan el mismo resumen o hash.

La resistencia a colisiones en este caso nos indica que será computacionalmente imposible encontrar un paralelo aleatorio de mensajes M y M prima sea igual a:

$$H(M) = h(M')$$

En 1998 se encontró una vulnerabilidad a SHA-0, aunque no fue reconocido para SHA-1, se desconoce si la Agencia de Seguridad Nacional (NSA) fue quien lo descubrió, pero ante esto se aumentó la seguridad de SHA-1.

En 2004 se encontró una debilidad matemática en SHA-1.

La resistencia de SHA-1, se ha visto comprometida durante el año 2005.

Después de que MD5, entre otros, quedará fuertemente comprometido en el 2004 por parque de un equipo de investigadores chinos [4],

El mismo equipo de investigadores chinos, compuesto por Xiaoyun Wang, Yiqun Lisa Yin y Hongbo Yu (principalmente de la Shandong University en China), ha demostrado que son capaces de romper el SHA-1 en al menos 2^{69} operaciones, unas 2000 veces más rápido que un ataque de fuerza bruta (que requeriría 2^{80} operaciones). Los últimos ataques contra SHA-1 han logrado debilitarlo hasta 2^{63} [5].

Según el NIST: "Este ataque es de particular importancia para las aplicaciones que usan firmas digitales tales como marcas de tiempo y notarías. Sin embargo, muchas aplicaciones que usan firmas digitales incluyen información sobre el contexto que hacen este ataque difícil de llevar a cabo en la práctica."

A pesar de que 2^{63} suponen aún un número alto de operaciones, se encuentra dentro de los límites de las capacidades actuales de cálculos, y es previsible que con el paso del tiempo romper esta función sea trivial, al aumentar las capacidades de cálculo y al ser más serios los ataques contra SHA-1.

ISSN: 2007-4786

El 23 de febrero de 2017, un equipo formado por Google y CWI Amsterdam, han anunciado la primera colisión de SHA-1, la cual ha sido nombrada como SHAttered.

En donde google consigue dos PDF diferentes con el mismo resumen o hash, para esto utiliza atraques de prefijo idéntico, que en teoría estaban en 2^{61}, pero google lo consigue en 2^{63} tras unos cuantos meses de computación y un costo de más de 1,000.00 dólares en Unidad de procesamiento Grafico (GPU).

6. PANORAMA DE SHA-2

Conforme a las pruebas mostradas por google, SHA-1 es vulnerable a ataques de colisiones y hoy en día no se requiere tanto poder computacional para realizar dicho ataque.

Ya que SHA-1 demostró ser vulnerable entra en materia SHA-2 y sus diferentes versiones, y aquí lo preocupante es los ataques a pre-imagen, que consiste en asegurarse de que, conociendo un hash y, el atacante no pueda encontrar un valor x tal que $h(x) = y$, 80 bit de complejidad todavía se considera generalmente "Bueno" y posiblemente vulnerable a un atacante bien financiado como una nación o estado.

Lo que significa que requiere un hash donde el mejor atacante de pre-imagen tiene una complejidad de más de 80 bits.

A continuación, se muestra la tabla 1 con los diferentes ataques al que se ha sometido SHA-2 y se ha mostrado fuerte:

Tabla 1. Ataques a SHA-2

Publicado	Año	Método de ataque	Ataque	Variante
Nuevos ataques a colisión contra un máximo de 24 pasos SHA-2	2008	determinista	Colisión	SHA-256 SHA-512
Pre-imagen para reducido paso a SHA-2	2009	Encontrarse en el medio	Pre-imagen	SHA-256 SHA-512
Avanzada satisfacer-en-el-medio ataques imagen inversa	2010	Encontrarse en el medio	Pre-imagen	SHA-256 SHA-512
De orden superior Ataque diferencial en Reducida SHA-256	2011	Diferencial	Pseudo-colisión	SHA-256
Bicliques para pre-imagen: Ataques a la madeja-512 y la familia SHA-2	2011	biclique	Pre-imagen Pseudo-imagen inversa	SHA-256 SHA-512
La mejora de los abordajes locales: Nuevos ataques contra Reducida SHA-256	2013	Diferencial	Colisión Pseudo-colisión	SHA-256 SHA-256
La heurística de ramificación en colisión diferencial Buscar con aplicaciones a SHA-512	2014	diferencial heurística	Pseudo-colisión	SHA-512
Análisis de SHA-512/224 y SHA-512/256	2016	Diferencial	Colisión Pseudo-colisión	SHA-256 SHA-512 SHA-512

ISSN: 2007-4786

7. IMPLEMENTACIÓN DE SHA-3

Como se había mencionado anteriormente, investigadores de china demostraron que se puede romper la seguridad de SHA-1 en menos de 269 rondas. Estas situaciones hicieron que el NIST convocará a una competencia para escoger la siguiente generación de funciones de resumen hash SHA-3.

En octubre de 2012 la función KeccaK fue elegida ganadora y en agosto de 2015 se publicó el nuevo estándar de funciones criptográficas hash SHA-3, bajo el Federal Information Processing Standards Publication, FIPS 202 [6].

No obstante, aparentemente el algoritmo es parte de la familia de la serie de estándar SHA, SHA-3 internamente es bastante diferente en filosofía a los algoritmos anteriores como SHA-1, SHA-2 y MD5.

La familia de las funciones hash SHA-3 está compuesta por seis funciones. Cuatros de ellas son funciones criptográficas hash llamadas SHA3-224, SHA3256, SHA3-348 y SHA3-512 y las otras dos funciones son de salida variable, conocidas como XOFs por sus siglas en inglés, llamadas SHAKE128 y SHAKE256, respectivamente. Las funciones de salida variable son diferentes a las funciones hash, pero es posible utilizarlas de forma similar, con la posibilidad de adaptarse a las necesidades criptográficas de manera individual [7].

Cualquier implementación de una familia de funciones esponja Keccak usa una de las siete permutaciones Keccak-f (Bertoni et al., 2001), denotada Keccak-f[b], donde el ancho de la permutación b ∈ {25, 50, 100, 200, 400, 800, 1600}. Las permutaciones Keccak-f son estructuras que constan de una secuencia de rondas casi idénticas. El número de rondas nr depende de b, y está dada por nr = 12 + 2l, donde 2l = b/25. Por ejemplo, Keccak-f [1600] usa 24 rondas. Un resumen de la función Hash Keccak-f[b] se presenta en el Algoritmo 12 (Bertoni, G., et. al., 2011) [8].

Algoritmo $Keccak\ f[b][(A)$

$for\ i\ in\ 0 \ldots . nr - 1$
$A\ = Round[b](A, RC[i])$
$Return\ A$

Algoritmo 1. Algoritmo $Keccak - f[b](A)$

7.1 DESCRIPCIÓN DEL ALGORITMO DE LA FUNCIÓN HASH

Keccak La ronda Keccak-f consistió en una secuencia de transformaciones invertibles, donde cada una realizó un proceso sobre la matriz de estado A de 5x5 líneas (lanes), y cada línea tiene una longitud w ∈ {1, 2, 4, 8, 16, 32, 64} (b = 25w). El Algoritmo 2 describe las transformaciones de la ronda de Keccak-f [8].

ISSN: 2007-4786

Transformaciones de ronda	
Entrada: b, A, RC	
Salida: A	$\forall\ x\ in\ 0\ \ldots\ 4$
1. Transformación θ	$\forall\ x\ in\ 0\ \ldots\ 4$
$C(x) = A[x,0] + A[x,2] + A[x,3] + A[x,4],$ $D[x] = C[x-1] + ROT[C[x+1]1],$ $A[x,y] = A[x,y] + D[x],$	$\forall\ (x,y) in$ $(0\ \ldots\ 4, 0\ldots 4)$
2. Transformaciones ρ y π	$\forall\ (x,y) in$ $(0\ \ldots\ 4, 0\ldots 4)$
3. Transformaciones Z	
$A[x,y] = B[x,y] + (NOT\ B[x+1,y])\ AND\ b[x+2,y]),$	$\forall\ (x,y) in$ $(0\ \ldots\ 4, 0\ldots 4)$
4. Transformaciones t	
$A[0,0] = A[0,0] + RC[i],$ Return A	

Algoritmo 2. Trasformaciones de Ronda Keccak-f.

8. CONCLUSIONES

En este artículo se ha analizado los principales hashes criptográficos dentro de la familia de estándar SHA y al día de hoy para desarrollar una firma electrónica en México se realiza con el estándar criptográfico de hash SHA-2, para ser más exacto se utiliza SHA-256 con una longitud de 2048 bits, seguridad que demuestra ser muy fuerte al día de hoy, no obstante y mientras el nivel computacional siga creciendo el algoritmo se podrá romper con colisiones o pre-imagen a como le paso a su sucesor SHA-1 independientemente la NIST ha estandarizado el nuevo sucesor del algoritmo de hash SHA-2 el cual aumenta su seguridad.

La filosofía de SHA-3 es muy diferente a sus versiones anteriores así que la forma de para romper el algoritmo será muy difícil de hacer ya que con la capacidad computacional de hoy en día no permite hacer tantas operaciones para realizar dicha tarea.

Asumiendo lo anterior, se puede estar seguro de que al momento de encontrar debilidades en el algoritmo de hash SHA2, el nuevo algoritmo SHA está a disposición de ser utilizado y demostrar que llegar a romperlo será una tarea que solo naciones podrán financiar.

Y se estará preguntando porque no utilizar SHA-3 desde ahora para estar más seguros, y la repuesta lo tiene su misma complejidad el algoritmo, por su naturaleza las operaciones y procesos que realiza para crear un hash o resumen requieren igual una capacidad computacional eficiente, lo cual hace que a nivel computacional todavía sea muy lento sus operaciones y en este sentido su versión anterior ofrece mayor eficiencia en la realización de operaciones criptográficas haciendo que las aplicaciones que lo usan sean lo suficientemente rápidas.

Por último y no menos importante que está tomando mucha importancia es la computación cuántica la cual se encuentra en teoría, pero siendo una realidad podría en tiempos y capacidad computacional romper los algoritmos que hoy en día son los estándares y sobre todo cualquier algoritmo que esté hecho para el sistema computacional clásico, no descartemos la idea de que pronto tendremos computadores cuánticos que harán estás operaciones en cuestión de segundos y el internet y la seguridad se verán comprometidos, pero esto ya es tema para otro artículo así que cuidado con sus algoritmos y siempre verifiquen que sean los más seguros estén en los sistemas.

REFERENCIAS

[1] **Angulo Castro, D. C., & Henao Leiva, J. F. (2018)**. Análisis de herramientas de interceptación para el control de ataques reales de suplantación con certificados SSL.

[2] **Talens-Oliag, S. (2008).** Introducción a los certificados digitales. Universidad de Valencia, España. https://www. uv. es/sto/articulos/BEI-2003-11/certificados_digitales. html. [Accessed: 28-Enero-2018].

[3] **Gaceta: LXI/2SPO-230/28987 (2011, marzo 17).** Recuperado de https://www.senado.gob.mx/64/gaceta_del_senado/documento/28987

[4] **Xiaoyun Wang, Dengguo Feng, Xuejia Lai, Hongbo Yu (17 de agosto de 2004).** «Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD» (pdf). International Association for Cryptologic Research (en inglés). Archivado desde el original el 20 de diciembre de 2004. Consultado el 08 de febrero de 2020.

[5] **Schneier, Bruce (18 de febrero de 2005).** «Cryptanalysis of SHA-1» (html). Bruce Schneier's Blog (en inglés). Archivado desde el original el 21 de febrero de 2005.

[6] **Nieto Ramirez, N., & Nieto Londoño, R. D. (2019).** Implementación hardware de la función Hash SHA3-256 usando una arquitectura Pipeline. Ingeniare. Revista chilena de ingeniería, 27(1), 43-51.

[7] **LEglisse, A. F. D. A., & García, G. G. (2018).** IMPLEMENTACIÓN DE LA FUNCIÓN SHA3-3 EN UNA ARQUITECTURA ARM. Pistas Educativas, 39(127).

[8] **Ramírez, M., Pino, C. A., Olaya, V. T., & Medina, J. V. (2013).** Implementación hardware del algoritmo Keccak para Hash-3 y comparación con Blake, Grosti, JH y Skein. Informador técnico, 77(2), 167-182.

Correo electrónico autor: *alamilla-96@hotmail.com, alejandrohc1984@hotmail.com*