C&T Riqchary
Revista de Investigación en Ciencia y Tecnología
ISSN: 2810-8124 (en línea) / ISSN: 2706-543x

Vol. 4 Núm. 1 (2022)
Publicado:05/11/22
DOI: https://doi.org/10.57166/riqchary/v4.n1.2022.81

Esquema de cifrado indistinguible bajo el ataque de texto sin formato elegido (IND-CPA)

Indistinguishable encryption scheme under chosen plaintext attack (IND-CPA)

Raul Huillca_Huallparimachi[A], Ecler Mamani_Vilca[B], Karla C. Rojas_ Pedraza [C] y Johel Cárdenas_Solano[D]

https://orcid.org/ 0000-0002-5205-3660[B] , https://orcid.org/0000-0002-7324-7616[C]
https://orcid.org/0000-0002-8607-3854 [D]

(Recepción: 15/06/2022 y aceptación 10/11/2022)

Resumen— El cifrado homomórfico es un tipo de cifrado con capacidad de procesamiento y evaluación sobre datos cifrados. En el presente trabajo se examina la vulnerabilidad con respecto al tipo de ataque indistinguible bajo ataque de texto sin formato elegido (IND-CPA), existente con respecto a la seguridad en el cifrado homomórfico. El cifrado homomórfico realiza diferentes tipos de cálculos sobre datos cifrados, algunos de ellos son los parcialmente homomorficos, bastante homomorficos y cifrado completamente homomórfico. Los cifrados parcialmente homomorficos procesan un solo tipo de operación, suma o multiplicación; los bastante homomorficos que pueden procesar las dos operaciones, pero para un conjunto limitado y los totalmente homomorficos que permiten procesar y evaluar una cantidad arbitraria de veces. Un ataque con texto claro elegido se hace presente en este tipo de cifrado donde el adversario conoce el sistema, es decir, que el sistema tiene que ser seguro frente a un adversario que conoce todo sobre él con excepción de la clave privada. Los niveles de seguridad presentes en la mayoría de los cifrados completamente homomorficos son suficientes para aplicaciones específicas, la carencia de la propiedad de Indistinguibilidad de un criptosistema hace, aun, poco viable la aplicación en producción de estos tipos de cifrado.

Palabra clave: cifrado homomórfico, RSA, criptografía.

Abstract— Homomorphic encryption is a type of encryption with processing and evaluation capabilities on encrypted data. This paper examines the vulnerability with respect to the indistinguishable attack type under chosen plaintext attack (IND-CPA), existing with respect to security in homomorphic encryption. Homomorphic encryption performs different types of computations on encrypted data, some of them are partially homomorphic, fairly homomorphic, and fully homomorphic encryption. Partially homomorphic ciphers process only one type of operation, addition or multiplication; the fairly homomorphic ones that can process both operations, but for a limited set, and the fully homomorphic ones that allow processing and evaluation an arbitrary number of times. A chosen cleartext attack occurs in this type of encryption where the adversary knows the system, that is, the system has to be secure against an adversary who knows everything about it except the private key. The security levels present in the majority of completely homomorphic ciphers are sufficient for specific applications; the lack of the Indistinguishability property of a cryptosystem makes the production application of these types of encryption still impractical.

Keyword: Homomorphic encryption, RSA, cryptography

A. *Raul Huillca_Huallparimachi, Departamento Académico de Ingeniería Informática de la UNSAAC, raul.huillca@unsaac.edu.pe.*
B. *Ecler Mamani_Vilca, Universidad Nacional Micaela Bastidas de Apurimac, eclervirtual@unamba.edu.pe*
C. *Karla C. Rojas_Pedraza, Universidad Nacional San Antonio Abad del Cusco, karla.rojas@unsaac.edu.pe*
D. *Johel Cárdenas_Solano, Universidad Nacional José María Arguedas, jcardenas@unajma.edu.pe*

1 INTRODUCCIÓN

En la mayoría de las empresas, organizaciones o instituciones (por no mencionar todos), en donde se utilicen equipos informáticos como herramienta de apoyo para el trabajo, las bases de datos tienen un papel muy importante y casi crítico, esto porque nos permite almacenar volúmenes mayúsculos de información acerca de las mismas. Por otra parte, la organización de esta información nos debe de permitir planear, gestionar, controlar y tomar decisiones.

Tradicionalmente, los datos se organizaban en archivos, los mismos que presentaban algunos problemas como la redundancia, el cual da lugar a inconsistencias cuando un dato no se actualiza en todos los lugares donde aparece; rigidez de búsqueda por ficheros; dependencia de los programas y por último los problemas de confidencialidad y de seguridad el cual nos permite evitar la consulta de ciertos datos de determinados usuarios. De las observaciones anteriores, se abordó una solución a estos problemas, y estas vienen a ser las bases de datos, los mismos que entre otras características nos permiten evitar la redundancia, flexibilidad de búsqueda, independencia de programas y la seguridad y confidencialidad integral.

Dentro del conjunto de actividades, de carácter crítico y cotidiano, correspondiente al rubro de la administración de la información en una institución, se tiene el de procesamiento de información y su posterior resguardo en una base de datos.

2 ESTADO DEL ARTE

Para la revisión de la literatura se utilizaron los siguientes criterios de inclusión: Año de publicación: 5 años atrás; Idiomas: inglés y español; Base de datos: Scopus y Google Académico; Área: Computación; Tipo de documento: Artículos, Tesis; Cantidad de artículos: 32, Cantidad de Tesis: 7; Palabras claves: cifrado homomórfico, RSA, criptografía homomórfica, privacidad, confidencialidad; Prioridad de revisión: título, resumen, conclusiones.

Dentro de las tendencias actuales, la era digital y la computación en nube, la seguridad de nuestros datos es una cuestión sin absolución de confiabilidad alta aun, es así entonces que aceptamos un riesgo para poder beneficiarnos de estas tecnologías de presentes. Una solución para garantizar esta confidencialidad son los cifrados homomorficos quienes aparte de ayudarnos con la seguridad nos permiten procesar los mismos sin necesidad de descifrarlos.

En este sentido, "La criptografía homomórfica contribuye a aumentar el uso de las tecnologías de nube dado que la información podría ser almacenada y manipulada mientras permanece cifrada" [1, p. 6]. Así mismo, "La criptografía simétrica define que el proceso de cifrado y descifrado hace uso de una única clave" [2, p. 10] . Por otra parte, "Este tipo de criptografía usa dos claves una de uso público y otra de uso privado que es la que permite descifrar el texto" [2, p. 11]. Para concluir, este tipo de cifrado "Permite realizar operaciones sobre el texto cifrado, sin necesidad de descifrar el contenido" [3, p. 59]. En efecto, "El cifrado homomórfico evita que los datos nunca estén desencriptados a la vez que permite su manipulación por las personas autorizadas" [4, p. 25].

2.1 CIFRADO PARCIALMENTE HOMOMÓRFICO

Las operaciones admitidas dentro de este tipo de cifrado con las de adición o las de multiplicación. Así mismo, "dentro de estos esquemas de cifrado parcialmente homomórfico (PHE) citamos como ejemplos (Rivest et al. 1978b; Goldwasser and Micali 1982; ElGamal 1985; Benaloh 1994; Naccache and Stern 1998; Okamoto and Uchiyama 1998; Paillier 1999; Damgård and Jurik 2001; Kawachi et al. 2007)" [5, p. 6]. Para resumir, estos esquemas descritos sirven como base para los otros esquemas de cifrado.

El cifrado RSA (Rivest, Shamir y Adleman) "es el primer ejemplo de PHE y fue introducido por Rivest et al. (1978b)" [5, p. 6]. Así mismo, "es el primer cifrado de tipo clave pública que era reconocido por tener propiedades homomórficas" [6, p. 10]. Este tipo de cifrado permite operaciones homomórficas una limitada cantidad de veces y solo con algunas operaciones; la operación homomórfica que permite es la multiplicación. Para comprender el funcionamiento de las operaciones se expone los pasos para cifrar en RSA:

Generación de claves: Primero, es necesario seleccionar números primos grandes, p y q. Después calculamos otro número, n = pq. También, usando la phi de Euler, calculamos la cantidad de números coprimos con p y q, y los multiplicamos. $\boldsymbol{\alpha = (p-1)(q-1)}$. Finalmente, buscamos un número, e, que tenga $\boldsymbol{mcd(e,\alpha)}$ y d usando el inverso multiplicativo de $\boldsymbol{ed \equiv 1\ mod\ \alpha}$. Con esto, tenemos los dos pares que necesitamos: $\boldsymbol{(e,n)}$ seria la clave publica y $\boldsymbol{(d,n)}$ la clave privada [5, p. 6].

Cifrando un mensaje: en RSA, cuando estamos cifrando mensajes, realmente lo que estamos haciendo es elevar el número del mensaje al número que habíamos declarado como e, y tomando el resultado módulo n. Es decir: $\boldsymbol{c = E(m) = m^e (mod\ n), \forall\ m\ \in M}$. RSA nos sirve como cifrado porque el proceso indicado anteriormente para cifrar los mensajes es relativamente fácil de computar. Por otro lado, el proceso de intentar descifrar el mensaje sin saber el inverso multiplicativo que habíamos declarado en e sería más complicado. Rivest, Shamir y Adleman marcan eso como requerimiento de un sistema

de criptografía público, un problema que sea fácil solucionar con todos los números del problema, pero muy difícil solucionar con solo un subconjunto de los números, por ejemplo, los datos de la clave pública [7].

Descifrar: los mensajes cifrados con RSA, entonces, pueden ser recuperados elevando el mensaje cifrado al inverso multiplicativo, modulo n. Recordemos que habíamos identificado previamente el multiplicativo inverso, y lo teníamos guardado como d, $\boldsymbol{m = D = C^d (mod\ n)}$.

Con referencia a lo anterior, el cifrado propuesto por Shafi Goldwasser y Silvio Micali en el año 1982 es un "es un cifrado probabilístico, esto quiere decir que el cifrado incluye un elemento de aleatoriedad, implicando que el mismo mensaje, cifrado dos veces, no tendrá el mismo texto cifrado" [6, p. 12]. Este cifrado permite operaciones homomórficas de adición.

Este algoritmo fue propuesto en el año 1985 "fue el primer cifrado de tipo clave pública que ofrecía seguridad basado en los problemas de los logaritmos discretos" [6, p. 14]. Dentro del mismo precisa que "Las curvas elípticas presentan posibilidades muy interesantes para el futuro de criptografía porque se cree que son problemas computacionalmente imposibles de romper" [6, p. 14]. Este tipo de cifrado está compuesto por las llaves públicas $< \boldsymbol{h, G, q, g} >$ donde G es un grupo cíclico de orden q con el generador g, mientras que la llave privada corresponde al valor $< \boldsymbol{x} >$ en donde x es un valor aleatorio elegido entre 1 y q. Además de que se debe cumplir que $\boldsymbol{h = g^x}$. Para realizar la encriptación de un mensaje m, con las llaves públicas $< \boldsymbol{h, G, q, g} >$, se genera un valor r que es un número aleatorio entre 1 y q. $\boldsymbol{E(m) = (c_1, c_2)}$; $\boldsymbol{c_1 = m * h^r}$; $\boldsymbol{c_2 = m * g^x}$ Mientras que para realizar la desencriptación del mensaje m es: $\boldsymbol{m = c_1 (c_2^x)^{-1}}$, considerando que $\boldsymbol{h = g^x}$, esto implica que $\boldsymbol{h^r = g^{xr}}$. Entonces tenemos que: $\boldsymbol{c_1 (c_2^x)^{-1} = m * g^{rx} * g^{-rx} = m}$ [8, p. 28]. Este cifrado posee una propiedad homomórfica, respecto a las operaciones multiplicativas.

Igualmente "El cifrado Benaloh permite cifrar los mensajes en bloques, en vez de cada bit individualmente" [6, p. 15], el cual "está basado en el problema de residuosidad superior (higher residuosity problem), que está relacionado al problema de la residuosidad cuadrática que se usa en los cifrados" [9]. Este cifrado posee una propiedad homomórfica en donde cualquier operación multiplicativa en el texto cifrado corresponde a una operación aditiva en el texto descifrado.

Después de las consideraciones anteriores citamos el "cifrado de Paillier que está basado en el problema de decisión de la residuosidad compuesta" [6, p. 19]. Este problema tiene que ver con "la búsqueda de un entero, x, que satisfaga la ecuación $\boldsymbol{x^n \equiv a (mod\ n^2)}$" (Acar, 2018, p. 6). Paillier es uno de los varios cifrados contra operaciones de adición. Para lograr la adición contra los mensajes descifrados, tenemos que multiplicar los mensajes cifrados. Donde : $\boldsymbol{E(m_1) * E(m_2) = \big((1+n)^{m1} r_1^n (mod\ n^2)\big) * \big((1+n)^{m2} r_2^n (mod\ n^2)\big) = E(m_1 + m_2)}$ **(Pardo, 2012)**

2.2 Cifrado bastante Homomórfico

Este tipo de cifrado "permite realizar algunas operaciones homomórficas (no todas) y una cantidad limitada de veces" [6, p. 8]. Dentro de estos tipos de cifrado podemos citar "(Yao 1982; Sander et al. 1999; Boneh et al. 2005; Ishai and Paskin 2007) en la literatura antes de 2009" [5, p. 9].

A este conjunto de cifrados también se les denomino cifrado Polly Cracker el cual puso énfasis en el uso de las Bases de Gröbner, que tuvo su comienzo en los años 90, cuando las vieron como una alternativa a la algebra multivariante. Los autores, como herramienta para este tipo de cifrado, muestran un procedimiento, que es similar a la reducción de matrices usando el método de Gauss. Así mismo, se "recomiendan que, si un investigador quería seguir investigando los cifrados que usen las ecuaciones multivariantes, que investiguen algoritmos más dispersos (sparse)" [10]. "La clave para estos tipos de cifrado es un par de ideales en un anillo polinomial multivariante" [11].

El cifrado Polly-Cracker empieza cuando Alice genera un punto en un vector, y, y un conjunto de polinomiales $\{\boldsymbol{q_i}\}$ que llegan a 0 en el punto y. Para mandar un bit, entonces, Bob tiene que generar una suma $\boldsymbol{p = \sum g_i q_i}$ y le manda a Alice el polinomial $\boldsymbol{p + m}$. Para descifrar, Alice evalúa el texto cifrado en secreto, $\boldsymbol{c\ mod\ q = m}$ para recibir el mensaje, m. Analizando lo descrito acá, vemos que el cifrado Polly Cracker tiene la distinción de tener una propiedad homomórfica tanto de adición como de multiplicación [6, p. 24]. En cuanto a los riesgos presenta vulnerabilidades los cuales son abordados agregando ruido al cifrado donde el mensaje original se mantiene escondido incluso frente a los ataques anteriormente mencionado.

2.3 Cifrado totalmente homomórfico

Un esquema que admite operaciones de suma y multiplicación en textos cifrados fue propuesto por Craig Gentry usando criptografía basada en celosía, a partir de estas operaciones es posible construir circuitos para realizar cálculos arbitrarios. El esquema propuesto por Gentry brinda no solo un esquema completamente homomórfico sino también un marco general para obtener un esquema FHE. En efecto, muchos investigadores han intentado diseñar un esquema FHE seguro y práctico después del trabajo de Gentry [5, p. 12]. El modelo que siguen estos criptosistemas parte de la construcción de un criptosistema algo homomórfico para posteriormente convertirlo en un criptosistema completamente homomórfico usando bootstrapping.

3 Discuciones

Las bases del cifrado homomórfico se sentaron a partir del año 1978, dentro de los mismos fue indispensable la

evolución que sufrió hasta el presente año. En primer lugar, la concepción del cifrado Parcialmente Homomórfico sirvió de base para el Bastante Homomórfico y esta para el cifrado Completamente Homomórfico. Visto esto, los conceptos matemáticos desde los más simples hasta los más complejos de entendimiento son la fortaleza para este tipo de cifrado.

Los trabajos posteriores a las diferentes etapas de la evolución del cifrado homomórfico llámese Parcialmente, Bastante y Completo tienen sus bases en estas o son aplicaciones de estos cifrados a ciertos tipos de resolución de problemas. Así mismo, el cifrado homomórfico si bien tiene un futuro prometedor, existen dificultades presentes como son los del "rendimiento computacional y los problemas de seguridad "(Schemes, 2015). Pero, por otra parte, si podemos evidenciar que varios cifrados completamente homomórficas están en producción en la actualidad.

Para el análisis del rendimiento computacional se evidencia que los cifrados parcialmente homomórficos son más rápidos que los cifrados completamente homomórficos. En cuanto a la seguridad podemos citar que "En 2019, investigadores marcaron que todos los cifrados homomórficos conocidos carecían de seguridad IND-CCA" [12, p. 5], donde el atacante tiene la habilidad de cifrar o descifrar mensajes arbitrarios con el cifrado elegido. Así mismo, "el tiempo de cómputo asociado es muy alto; Gentry menciona que, si Google usara un algoritmo completamente homomórfico en su buscador para proteger la privacidad de las búsquedas, el tiempo de procesamiento de todas las operaciones podría aumentarse en trillones" [4, p. 18]. En definiva, las propiedad de seguridad de los cifrados homomorficos son mas debiles que los esquemas no homomorficos.

"Varios articulo ponen énfasis en la seguridad de este tipo de cifrado y que para el mismo proponen esquemas de cifrados cuánticos" [13], así como esquemas con circuito de tamaño polinomial arbitrario. Por otra parte, en cuanto al performance muchos de los autores citados coinciden que para la viabilidad de este tipo de cifrado se tiene que superar el rendimiento computacional requerido por el mismo, es así, que algunos proponen el uso de programación paralela haciendo uso de unidades de procesamiento de gráficos(GPU).

Sobre el cifrado de la información:

Para dar inicio a este proceso se tiene ya definido el conjunto de los datos y las dos operaciones binarias y sus correspondientes propiedades, los cuales forman una estructura algebraica de cuerpo. Así mismo, se tiene ya definido la función homomórfica de anillos que se hará uso durante todo el proceso de cifrado, a continuación, se exponen el cuadro de las dos operaciones binarias de la estructura algebraica y el cuadro de la tabla de disyunción con su equivalente booleano.

Tabla 1. Tabla de verdad

P	Q	PVQ
0	0	1
0	1	1
0	0	1
0	1	1

Tabla 2. Tabla de equivalente booleano

P	Q	PVQ
F	F	F
F	V	V
V	F	V
V	V	V

Así mismo, un homomorfismo de anillos cumple las siguientes propiedades:

$$\emptyset(\boldsymbol{a}+\boldsymbol{b}) = \emptyset(\boldsymbol{a}) + \emptyset(\boldsymbol{b})$$
$$\emptyset(\boldsymbol{ab}) = \emptyset(\boldsymbol{a})\emptyset(\boldsymbol{b})$$

Entonces, retomando el ejemplo anterior donde se cita el conjunto $\boldsymbol{K} = \{\boldsymbol{0}, \boldsymbol{1}\}$ con cuyos elementos podemos definir un homomorfismo de anillos como $\boldsymbol{a} \rightarrow \boldsymbol{a}\ (\boldsymbol{mod}\ \boldsymbol{n})$ y dentro de ellos se definen dos operaciones binarias como son la suma y la multiplicación los cuales cumplen con las propiedades pertinentes a dichas operaciones. En este sentido, se toma como parámetros de entrada para las propiedades de un homomorfismo de anillos a todas las combinaciones posibles de los elementos del conjunto, este conjunto de combinaciones es equivalente al conjunto de combinaciones booleanas de los parámetros de entrada de una consulta de disyunción.

Operando para los valores a= 1, b= 1 y n= 2 de la tabla de verdad:

$\emptyset(1+1) = (1+1)(mod\ 2)$
$\emptyset(1+1) = (1\ mod\ 2) + (1\ mod\ 2)$
$\emptyset(1+1) = \emptyset(1) + \emptyset(1)$
$\emptyset(1+1) = 1+1$
$\emptyset(1+1) = 0$

Y

$\emptyset(1*1) = (1*1)(mod\ 2)$
$\emptyset(1*1) = (1\ mod\ 2) * (1\ mod\ 2)$
$\emptyset(1*1) = \emptyset(1) * \emptyset(1)$
$\emptyset(1*1) = 1*1$
$\emptyset(1*1) = 1$

Tabla 3. Tabla de transformación homomórfica

+	0	1
0	0	1
1	1	0

Tabla 4. Tabla de transformación homomórfica

*	0	1
0	0	0
1	0	1

La estructura de homomorfismos de anillos aplicada y cuyo resultado es el cuadro mostrado, nos da a entender que estas dos tablas son equivalente o iguales. Quiere decir que el resultado de la operación que realizamos en la parte derecha de la tabla es igual al resultado de la operación que realizamos en la parte izquierda de la tabla, es así entonces que se logra el objetivo planteado en el presente trabajo de investigación.

TABLA 4
Tabla resultada (equivalente homomórfico)

A	B	A V B
F	F	F
F	V	V
V	F	V
V	V	V

A(+)	B(*)	A V B
0	0	**0**
1	0	**1**
1	0	**1**
0	1	**1**

Tabla 5. Tabla resultada (equivalente homomórfico)

A	B	AVB
F	F	F
F	V	V
V	F	V
V	V	V

Tabla 6. Tabla resultada (equivalente homomórfico)

A(+)	B(*)	AVB
0	0	0
1	0	1
1	0	1
0	1	1

5 TRABAJOS FUTUROS

La literatura presente a la actualidad propone que en trabajos futuros al respecto del cifrado homomórfico un aspecto muy importante es el que corresponde a la seguridad. Es cierto que los niveles de seguridad presentes en la mayoría de los cifrados completamente homomorficos son suficientes para aplicaciones específicas, la carencia de la propiedad de Indistinguibilidad de un criptosistema hace, aun, poco viable la aplicación en producción de estos tipos de cifrado.

Él porque es, aun, inadecuado el cifrado homomórfico para aplicaciones de propósito generar se responde al hecho de que un criptosistema se considera seguro si un adversario modelado por una máquina de Turing de tiempo polinómico probabilístico tiene una sola ventaja insignificante sobre las conjeturas aleatorias. Se dice que un adversario tiene una ventaja insignificante si es ganador del juego con probabilidad $(\mathbf{1/2}) + \boldsymbol{e(k)}$, donde $\boldsymbol{e(k)}$ es una función insignificante en el parámetro de seguridad k, que es para cada función polinómica distinta de cero $\boldsymbol{poly}()$ existe $\boldsymbol{k_0}$ tal que $|\boldsymbol{e(k)}| < |\mathbf{1/poly(k)}|$ para todos $\boldsymbol{k > k_0}$.

En esta revisión bibliografía se abordó los conceptos básicos de los diferentes esquemas FHE, los mismos que desde sus inicios fueron un avance significativo como una herramienta para ayudar a la seguridad de la información. Esperamos que esta revisión ayude a la comprensión y entendimiento de nociones relacionadas al Cifrado Homomorfico.

4 CONCLUSIONES

Las prestaciones confidenciales de un gestor de base de datos actual nos permiten garantizar la confidencialidad, integridad, disponibilidad y autenticación de la información contenida en los mismos, así mismo, para respaldar esta seguridad también podemos hacer uso de políticas para la gestión de claves como la tokenizacion, riguroso control de acceso, identificación de datos sensibles y críticos, cifrar parte de la información, y la anonimización (el cual también puede ser alcanzado haciendo uso del cifrado homomórfico de anillos). Sobre la base de estas consideraciones, al ser la disyunción información contenida en una tabla de la base de datos todas estas prestaciones también se aplican a ellas.

Con referencia a lo anterior, los diferentes algoritmos de cifrado existentes tienen características que se tienen que tener muy en consideración, con respecto al cifrado empleado en el presente trabajo de investigación como por ejemplo que un cifrado más seguro consume más recursos que uno no seguro, que las claves largar producen un cifrado más fuerte que las cortas, un cifrado asimétrico es más lento que uno simétrico, etc. Sobre la base de estas consideraciones, nos planteamos el cómo realizar una consulta cotidiana en estas bases de datos cifradas y si este fuera el caso tenemos que hacer una consulta simple como cualquier otra y es aquí donde el cifrado homomórfico entra a tallar, puesto que para alcanzar el objetivo (resultado de una consulta), no se requiere hacer un proceso previo de transformación o de descifrado.

6 RECOMENDACIONES

De la experiencia recogida en este trabajo, el cual trata sobre el cifrado de datos homomórficos, se pone de manifiesto la dificultad de este tipo investigación puesto que se requiere del estudio y comprensión de la matemática moderna, teoría de conjuntos, estructuras algebraicas entre otros, los cuales en conjunto nos ayudan a mejorar la seguridad de los datos. En este sentido, se recomienda las siguientes tendencias de cifrado de información para la década venidera, los cuales comprenden los años, 2020 -2030.

Como primera recomendación para mejorar la seguridad de la información podemos citar la criptografía cuántica, del cual citamos que hasta el momento actual se ha desarrollado un algoritmo cuántico de distribución de claves que garantiza únicamente la confidencialidad e integridad de estas; sin embargo, no se ha logrado el desarrollo de un algoritmo cuántico que garantice su autenticidad. En este sentido, si en el futuro algún trabajo de investigación llegase a encontrar algoritmos cuánticos de autenticación, entonces si podríamos decir que habría nacido la criptografía cuántica.

Como segunda recomendación para mejorar la seguridad de la información podemos citar el protocolo para el intercambio seguro de mensajes propuesto por Diffie-Hellman haciendo uso de estructuras algebraicas no conmutativas [12], de cual podemos citar que los trabajos de investigación abarcan a estructuras de anillos, pero podríamos utilizar otras estructuras no conmutativas como son los quasigrupos, grupos de trenzas, grupos nilpotentes.

Referencias

[1] G. A. Cubillos Franco, «Protección de datos compartidos en entornos de nube,» p. 76, 2020.

[2] B. D. Rugel Campoverde, «Seguridad en el sistema de gestiónde datos medidos de energía eléctrica aplicando cifrado homomórfico,» p. 27, 2019.

[3] K. V. Villacres Pacheco, «Estudio de factibilidad de la seguridad para el desarrollo de un prototipo Web de voto electrónico en la Universidad de Guayaquil,» Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas, p. 215, 2021.

[4] L. A. Quemba Martínez, «Cifrado de la información y su incidencia actual en la seguridad de la información para pequeñas empresas pymes en Colombia,» p. 99, 2020.

[5] A. Acar, «A survey on homomorphic encryption schemes: Theory and implementation,» ACM Computing Surveys (CSUR), pp. 1-35, 2018. https://doi.org/10.1145/3214303

[6] L. Udstuen, «Facultades de Ciencias Económicas, Cs. Exactas y Naturales e Ingeniería,» p. 42, 2021.

[7] R. Rivest, «Shamir, a. and Adelman,» L." On Digital Signatures and Public Key, 1978.

[8] A. Bravo, «Selección e implementación de librería Java de algoritmo para E-Voting,» pp. 1-233, 2018.

[9] Y. ZHENG, «Residuosity problem and its applications to cryptography,» IEICE TRANSACTIONS (1976-1990), pp. 759-767, 1988.

[10] Barkee, «Why you cannot even hope to use Gröbner bases in cryptography: an eternal golden braid of failures,» Applicable Algebra in Engineering, Communication and Computing, pp. 235-252, 2020. https://doi.org/10.1007/s00200-020-00428-w

[1 1] Caboara, «Lattice polly cracker cryptosystems,» Journal of Symbolic Computation, pp. 534-459, 2011. https://doi.org/10.1016/j.jsc.2010.10.004

[12] Z. Peng, «Danger of using fully homomorphic encryption: A look at microsoft SEAL,» arXiv preprint arXiv:1906.07127, 2019.

[13] Y. Ouyang, «Quantum homomorphic encryption from quantum codes,» Physical Review A, 2018. https://doi.org/10.1103/PhysRevA.98.042334

[14] J. L. G. Pardo, «Cifrado homomórfico: ejemplos y aplicaciones,» Gaceta de la Real Sociedad Matematica Española, pp. 697-712, 2012.

BIOGRAFÍAS

Raul huillca huallparimachi, Maestro en Ciencias con Mención en Informática, Maestro en Administración. Docente del Departamento Académico de Ingeniería Informática de la UNSAAC. A la fecha, Estudiante del Doctorado en Ingeniería de Sistemas en la Universidad Nacional Mayor de San Marcos.

Ecler Mamani Vilca, Dr. en Ciencias de la Computación, desarrollador de aplicaciones multimedia y Software Educativo Intercultural, docente nombrado a Tiempo completo en la Universidad Nacional Micaela Bastidas de Apurímac regenta los cursos de Algorítmica, Computación Gráfica y Seminario de Tesis, docente de la Escuela de Post Grado UNAP, trabajó en diferentes universidades privadas e institutos superiores técnicos del Perú.

Karla Catherine Rojas Pedraza, Ingeniera de Sistemas, Egresada de la maestría en Ciencias con Mención en Informática de la Universidad Nacional San Antonio Abad del Cusco, Auditor Senior de Tecnología de la Información en la Gerencia de Auditoría Interna de la Entidad Financiera Caja Municipal Cusco.

Johel Cárdenas Solano, Docente de la Universidad Tecnológica de los andes y personal administrativo nombrado de la Universidad Nacional José María Arguedas y con estudios de materia en informática educativa y tecnológica de la información