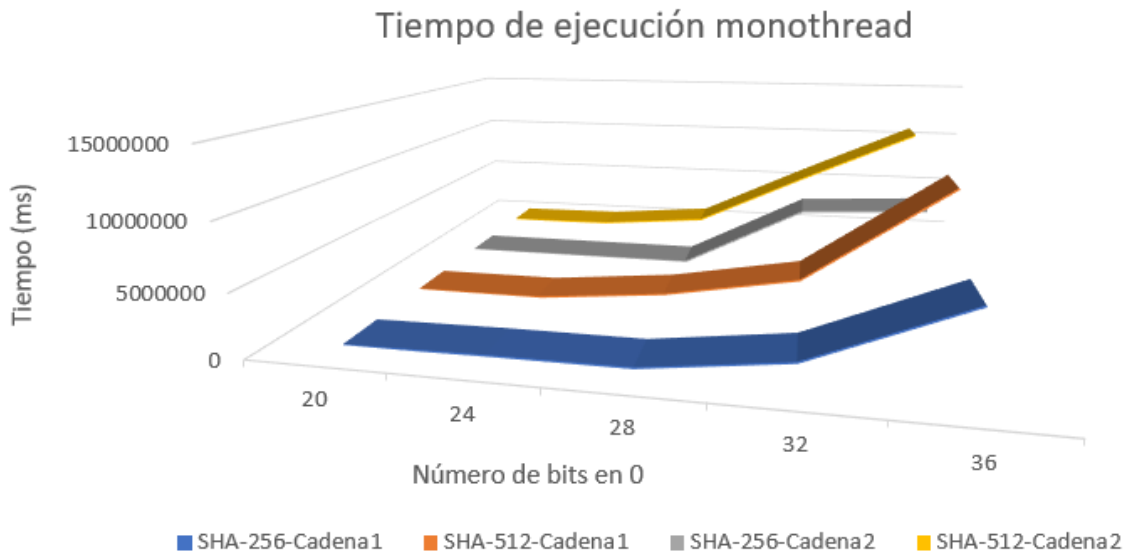
Documentación Caso 3 – Infraestructura Computacional

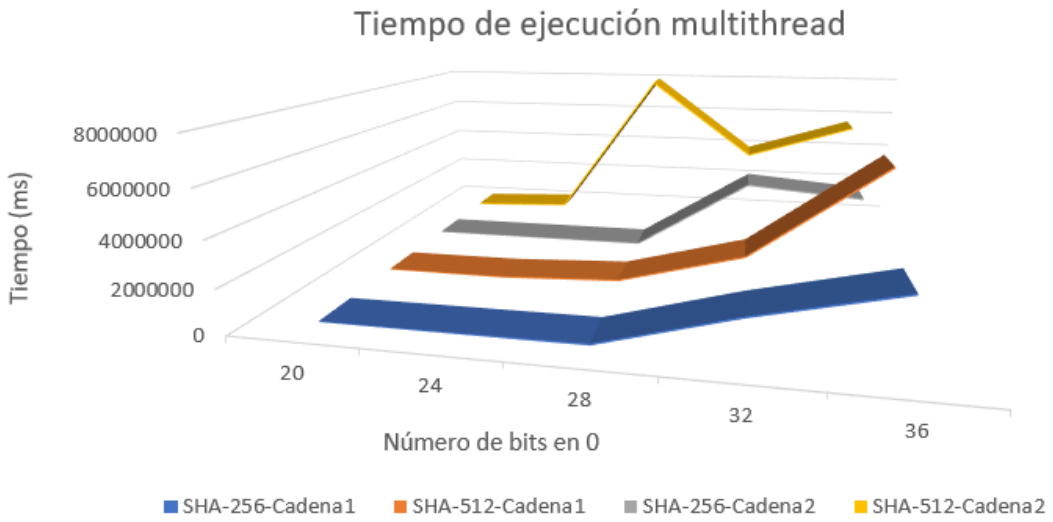
Pablo Pedreros – 202112491
Andrés Felipe Romero – 202013448
Juan Sebastian Urrea – 201914710

A. Implementación del Prototipo.
i) Tabla de resultados de ejecución con distintos parámetros:

TIEMPO EN MS MONOTHREAD				
Número de ceros	SHA256-Cadena1	SHA512-Cadena1	SHA256-Cadena2	SHA512-Cadena2
20	2062	6188	225	2166
24	134341	43147	13877	124091
28	160763	1073194	26359	1050002
32	1572736	3021571	5519700	6081892
36	6340455	10775439	6066478	10891531
TIEMPO EN MS MULTITHREAD				
Número de ceros	SHA256-Cadena1	SHA512-Cadena1	SHA256-Cadena2	SHA512-Cadena2
20	1374	3123	359	3047
24	24704	32291	14119	230468
28	52402	275171	26954	7930143
32	1692902	1813165	3558301	3880875
36	3053252	6070742	3089971	5623125

ii) Gráficas de comportamiento:





iii) Velocidad del procesador y cálculo de ciclos para una comprobación:
El procesador donde se hicieron las pruebas es de 2GHertz, o sea $2 * 10^9$ ciclos por segundo. Tomamos como muestra el tiempo con un solo thread, 36 bits en cero y cifrado con SHA-256. En este caso no se encontró ninguna solución v, lo que significa que se recorrió todo el espacio de búsqueda, por lo que podemos dividir el tiempo total entre el tamaño del espacio de búsqueda para hallar el tiempo que tarda en generar y comprobar un solo caso. El tamaño del espacio de búsqueda son todas las posibles combinaciones del alfabeto de longitud 7 más todas las combinaciones de longitud 6 y así hasta longitud 1. Así, el tamaño del espacio es de $26^7 + 26^6 + 26^5 + 26^4 + 26^3 + 26^2 + 26$. Como el tiempo de ejecución fue de 6340455ms, al dividirlo entre el espacio de búsqueda nos da un tiempo de $\frac{6340455}{(26^7+26^6+26^5+26^4+26^3+26^2+26)} = 0.00075ms$, o sea $0.75 * 10^{-6}$ segundos, que si lo multiplicamos por el número de ciclos por segundo nos da un total de 1500 ciclos aproximadamente por cada valor generado y comprobado.

iv) Tiempo del peor caso para un programa MonoThread:
Si son 1500 ciclos por cada valor y el tamaño del espacio de búsqueda es el antes mencionado entonces, al multiplicar ambos valores, tenemos que recorrer todo el espacio de búsqueda toma $1500 \cdot (26^7 + 26^6 + 26^5 + 26^4 + 26^3 + 26^2 + 26) = 1.25296 * 10^{13} ciclos$, que dividiendo por los ciclos que se hacen por segundo nos da un total de $\frac{1.25296*10^{13}}{2*10^9} = 6264.812\ segundos$ para recorrer todo el espacio. Esto es naturalmente aproximado al valor mediante el cual calculamos los ciclos en el punto anterior.

B. Análisis y Entendimiento del Problema.

Punto 1
i) ¿Cuáles se usan hoy en día? Hoy en día se utilizan familias de algoritmos entre las que se encuentran DH, E-DES, RSA, T-DES, ECC, EEE, RC4, RC2, BLOWFISH, DSA, RC6 y AES (Abood and Guirguis, 2018). Se destacan por su buena seguridad, aunque varían entre rápidos y lentos. Abood and Guirguis (2018) también destacan algunos algoritmos que no se usan más.
ii) ¿Por qué dejamos de usar aquellos que se consideran obsoletos? Algunos que se consideran obsoletos incluyen DES, SEAL, MD5, SHA-1. Los dos primeros debido a que no son lo

suficientemente fuertes (Abood and Guirguis, 2018), mientras que los dos últimos debido a que se requieren capacidades computacionales pequeñas para revertir su salida y se espera que que los avances futuros en hardware hagan esta requerimiento cada vez más pequeño (Jašek, 2015).
iii) ¿Qué referencias bibliográficas usó para responder esta pregunta? Se consultó la información desde un *survey* del estado del arte en criptografía y un artículo científico que realiza un análisis de seguridad sobre los algoritmos SHA-1 y MD5. Ambas fuentes provienen de revistas indexadas.
iv) ¿Por qué esas referencias tienen autoridad sobre este tema? Las fuentes se publicaron en revistas con prestigio científico y sus investigadores tienen una trayectoria reconocida en la disciplina.

Punto 2

Blockchain puede ser utilizado en la Universidad de los Andes para la verificación de los certificados académicos de los estudiantes, incluyendo certificados de estudios, actas de graduación, reconocimientos académicos, etc.

i) ¿Cuáles de los cuatro problemas de seguridad resuelve blockchain en el caso presentado?

Integridad: Blockchain garantiza la integridad de los certificados mediante su estructura de cadena de bloques. Cada bloque contiene un hash que depende del contenido del bloque anterior. Si se intenta modificar la información en un bloque, el hash cambiará, afectando a todos los bloques siguientes. Este vínculo criptográfico asegura que una vez que un certificado se registra en la cadena de bloques, cualquier intento de alteración resultará en una detección inmediata.

No-repudio y Autenticación: La capacidad de no-repudio, vinculada a la autenticación, se logra a través de la criptografía de clave pública. Cada entidad tiene un par de claves: pública y privada. La universidad, al emitir un certificado, firma digitalmente el contenido utilizando su clave privada. Cualquier persona externa puede verificar esta firma utilizando la clave pública de la universidad. Esto garantiza que el certificado fue emitido por la Universidad de los Andes y que la universidad no puede negar su emisión, proporcionando no-repudio.

ii) ¿Cómo los resuelve?

Estructura de Cadena de Bloques: Cada certificado académico se registra como un bloque en la cadena. Los bloques están enlazados mediante hashes criptográficos, asegurando la integridad. Cualquier cambio en un bloque afectaría a todos los bloques subsiguientes, lo que hace que la alteración sea fácilmente detectable.

Criptografía de Clave Pública: Se utiliza para lograr la autenticación y el no-repudio. La universidad utiliza su clave privada para firmar digitalmente los certificados. Cualquier persona externa puede verificar esta firma utilizando la clave pública de la universidad. Esto garantiza la autenticidad del emisor y proporciona evidencia de no-repudio.

Consenso Descentralizado: La naturaleza descentralizada de la cadena de bloques implica que la información se almacena en múltiples nodos de la red. El consenso entre estos nodos se alcanza para validar las transacciones. Esto asegura que la emisión y la veracidad de los certificados sean aceptadas por consenso, evitando manipulaciones maliciosas.

Bibliografía

Abood, O. G., Guirguis, S. K. (2018). *A Survey on Cryptography Algorithms*. International Journal of Scientific and Research Publications, Volume 8, Issue 7.

Jašek, R. (2015). *SHA-1 and MD5 cryptographic hash functions: Security overview*. Komunikácie, vol. 17, iss. 1, s. 73-80. ISSN 1335-4205.