

### Caso de estudio # 3 Seguridad

#### Objetivos

- Comprender el alcance y limitaciones de los algoritmos para generación de códigos criptográficos de hash y su aplicación en el manejo de integridad de datos.
- Construir un prototipo a escala de una herramienta que evalúa el esquema para manejo de integridad que usa la tecnología bitcoin.

#### Problemática:

Bitcoin registra las transacciones en bloques de un tamaño definido y encadena los códigos criptográficos de hash de dichos bloques para garantizar la integridad de los datos. Adicionalmente, bitcoin incluye un procedimiento conocido como “minado”; este procedimiento se debe ejecutar por cada bloque y se encarga de buscar un valor (v) que concatenado con las transacciones que ya están en el bloque, permite generar un código criptográfico de hash (ch) que cumple una condición específica; usualmente la condición se especifica como el número de ceros que el código generado (ch) debe tener al comienzo. Por las propiedades de los algoritmos de generación de códigos criptográficos de hash, el proceso de minado debe usar una aproximación de ensayo y error, intentando diferentes valores hasta encontrar uno que cumpla con la condición buscada.

*Nota:* en el artículo que propone bitcoin originalmente puede encontrar información adicional. La referencia se encuentra al final del enunciado.

#### A. [85%] Implementación del Prototipo.

Su tarea consiste en escribir un programa en Java que:

- Recibe como datos de entrada: (i) un algoritmo de generación de código criptográfico de hash, (ii) una cadena que representa los datos de una transacción y (iii) un entero que indica el número buscado de ceros.
- Explora un espacio de búsqueda, buscando un valor v tal que el código criptográfico de hash de la cadena de entrada concatenada con v tenga en su comienzo el número buscado de ceros.

Su programa debe cumplir con las siguientes condiciones:

- No puede usar las instrucciones `break`, ni `continue`. Vamos a contribuir al debate `break`, `continue`, `goto` vs. programación estructurada.
- No use librerías especiales, solo las librerías estándar de java (`java.security`). Si tiene alguna duda sobre una librería específica consulte con los profesores.
- El programa debe reportar la cadena de entrada usada, el valor v que permite cumplir la condición definida y el tiempo que tomó hacer la búsqueda.

Para simplificar un poco el problema:

- Escoja una cadena con datos de entrada con una longitud máxima de 32 caracteres.
- Los algoritmos para generación de código criptográfico de hash que usaremos son `SHA256` y `SHA512`.
- El número de ceros buscado puede ser: 24, 32, 40, 48, 56 o 64. Este número corresponde a la cantidad de bits en cero que deben aparecer al comienzo del código generado.
- El espacio de búsqueda para el valor v debe considerar cadenas con una longitud entre 1 y 7 caracteres (longitud máxima de 7 caracteres) y solamente consideraremos letras minúsculas (a-z). Tenga en cuenta que, en el caso de bitcoin, y otras aplicaciones que usan la técnica de minado, se trabaja con todas las combinaciones posibles de bits, no solo con los códigos correspondiente a letras.
- Adicionalmente, por limitar el tamaño del espacio de búsqueda, es posible que después de explorar todos los valores en el espacio no se encuentre una respuesta. Si no se encuentra respuesta, entonces el tiempo reportado

debería corresponder al tiempo de revisión de todo el espacio de búsqueda. Tenga en cuenta que, en un contexto real sería necesario continuar generando valores hasta encontrar uno que permita cumplir con la condición.

Adicionalmente, resuelva las siguientes tareas:

1. Corra su programa en diferentes escenarios contruidos a partir de las variaciones en los datos de entrada: para cada algoritmo, sobre dos cadenas diferentes y **para número de ceros 24, 32, 40, 48, 56 y 64 y construya una tabla con los datos recopilados.**
2. Construya una gráfica que muestre los tiempos registrados para los dos algoritmos usados de acuerdo con el número de ceros buscados. Use los datos del punto anterior.
3. Identifique la velocidad de su procesador, y estime cuántos ciclos de procesador toma, en promedio, generar y evaluar un valor para determinar si cumple o no con la condición buscada. Escriba todos sus cálculos.

#### **B. [15%] Análisis y Entendimiento del Problema.**

*Nota:* No copie contenido de otros autores, construya sus propias respuestas con base en las referencias consultadas (e incluya dichas referencias en el informe).

1. Busque información adicional sobre los algoritmos de generación de códigos criptográficos de hash: **(i) cuáles se usan hoy día y en qué contexto y (ii) por qué dejamos de usar aquellos que se consideran obsoletos.**
2. La tecnología blockchain se construyó a partir de la propuesta de bitcoin. Busque información sobre blockchain y presente un caso de uso en el que sería razonable incorporar la tecnología blockchain (diferente a bitcoin). Justifique su respuesta: use argumentos concretos (considere las preguntas: ¿qué problemas resuelve blockchain en el caso presentado? y ¿cómo los resuelve?).

#### **Entrega:**

- Cada grupo debe entregar un zip de un proyecto Java con: La implementación del prototipo. En el subdirectorío docs debe haber un archivo que incluya el informe con las respuestas a los puntos A.1 a A.3 y B.1 y B.2. **Al comienzo del informe, deben estar los nombres y carnés de los integrantes del grupo.** Si un integrante no aparece en el documento entregado, el grupo podrá informarlo posteriormente. Sin embargo, habrá una penalización: la calificación asignada será distribuida (dividida de forma equitativa) entre los integrantes del grupo.
- Recuerde incluir en su informe **todas las referencias** que use para resolver este proyecto.
- El trabajo se realiza en grupos de **2** personas. No debe haber consultas entre grupos.
- El grupo responde solidariamente por el contenido de todo el trabajo, y lo elabora conjuntamente (no es trabajo en grupo repartirse puntos o trabajos diferentes). Se puede solicitar una sustentación a cualquier miembro del grupo sobre cualquier parte del trabajo. Dicha sustentación será parte de la calificación de todos los miembros.
- El proyecto debe ser entregado por Sicua+ por uno solo de los integrantes del grupo.
- **La fecha límite de entrega es 26 de abril 2021 a las 23:50 p.m.**

#### **Referencias:**

- *Cryptography and network security*, W. Stallings, Ed. Prentice Hall, 2003
- *Computer Networks*. Andrew S. Tanenbaum. Cuarta edición. Prentice Hall 2003, Caps 7, 8.
- *To use or not to use the goto statement: Programming styles viewed from Hoare Logic*. Hidetaka Kondoh, Kokichi Futatsugi. Science of Computer Programming. Volume 60, issue 1. 2006
- *Bitcoin: A Peer-to-Peer Electronic Cash System*. Satoshi Nakamoto. 2009