Especificación Formal de Operaciones de Blockchain usando ℤ

Manuel Figueroa, *ITCR*, Esteban Leandro, *ITCR*

***Resumen*— Blockchain es en esencia una base de datos distribuida, una capa pública de todas las transacciones o eventos que han sido ejecutados y compartidos entre todas las partes participantes del sistema. Cada una de las transacciones es verificada por consenso entre la mayoría de los participantes. Una vez que se ingresa información en el sistema no puede ser removida, por lo que se considera que la tecnología de blockchain es útil para mantener registros confiables y verificables de cada una de las transacciones realizadas en algún momento [1]. La especificación formal de las operaciones de cada nodo participante en la red de blockchain nos permite determinar las operaciones requeridas para lograr consenso y registrar las transacciones realizadas en el sistema.**

***Index Terms*— Blockchain, Especificación, Formal, Z.**

I. Introduction

UN sistema blockchain introduce la capacidad de mantener de manera descentralizada y distribuida un registro inmutable, confiable y verificable de transacciones o eventos realizados por los participantes del sistema. Es importante destacar de que parte de la seguridad inherente del sistema se debe a las operaciones realizadas por los nodos participantes de sistema distribuido. El sistema ordena las transacciones colocándolas en grupos denominados bloques, estos bloques se conectan entre sí obteniendo el nombre de blockchain (cadena de bloques).

Para decidir cuando un bloque debe ser agregado a la cadena, el sistema crea un mecanismo de verificación criptográfica que se conoce como "*proof of work*" esto obliga a los nodos que participan a resolver un problema matemático usualmente difícil, cuya solución requiere de mucho poder computacional y que asegura que la incorporación de bloques fraudulentos sea prácticamente imposible, ya que el poder computacional requerido para lograr una cadena de bloques verificable y que sea aceptada por la totalidad de la red, sería inmenso.

II. Expresando El modelo en ℤ

Para simplificar la descripción del modelo y abstraer las particularidades de implementaciones orientadas al manejo de criptomonedas, el modelo descrito está basado en la implementación de Lauri Hartikka de NaiveChain, que implementa el algoritmo básico de blockchain sin las complejidades propias de otras plataformas como Bitcoin y Ethereum[3].

Vamos a definir las entidades que debemos conocer para implementar un sistema de blockchain.

Inicialmente definimos todos los mensajes posibles del sistema en el conjunto *Reporte* que consiste en la siguiente enumeración:

$$Reporte ::= Okay \mid BloqueInvalido \mid Rechazo$$

A. Bloques

4 correspondientes a las transacciones o eventos ocurridos en el sistema. Para lograr este encadenamiento cada bloque debe contener además de los datos, una firma única que lo identifica y la firma del bloque anterior.

Para definir un bloque nuevo requerimos los siguientes arreglos dados, para poder registrar los datos, la firma del bloque actual y anterior y la fecha de creación del bloque.

$$[FINGERPRINT, TIMESTAMP, DATA]$$

No requerimos saber más detalle de la estructura interna de los datos del bloque, ni de la representación de las firmas criptográficas de cada bloque.

Entonces un bloque va a estar definido por el esquema:

Bloque

$fingerprint : FINGERPRINT$
$prevblock : FINGERPRINT$
$indice : \mathbb{N}$
$timestamp : TIMESTAMP$
$data : DATA$

Se va a requerir de una función que nos calcula el fingerprint de un bloque, vamos a describir dicha función como:

$$getFingerprint : Bloque \longrightarrow FINGERPRINT$$

B. Nodos

Establecemos que un nodo en la cadena de blockchain es una entidad que mantiene una secuencia dada de bloques.

Nodo

$$cadena: \text{seq } Bloque$$

También requerimos definir una función que nos valide si el bloque se puede agregar a la cadena:

ValidarBloque

$$\Xi Nodo$$
$$bloque : Bloque$$
$$m! : Reporte$$

$$bloque.prevblock = (last\ cadena).fingerprint$$
$$getFingerprint(bloque) = bloque.fingerprint$$
$$bloque.indice = (last\ cadena).indice + 1$$
$$m! = Okay$$

III. Operaciones Del Nodo

Para establecer un conjunto de operaciones realizadas por los nodos individuales del sistema de blockchain, primero tenemos que definir que responsabilidades tiene cada una de estas entidades en el sistema.

Un nodo de blockchain además de mantener un registro de la cadena de bloques tiene como responsabilidad realizar la verificación de un nuevo bloque a ser incorporado en la secuencia. También es responsable de minar bloques, esta operación hace que el nodo tenga que establecer un valor de "proof of work" para un bloque para que este pueda ser enviado a la red y que sea aceptado por los demás nodos como el siguiente nodo válido en la cadena.

A. Agregar un nodo a la cadena

La operación para agregar un nuevo bloque a la cadena de bloques del nodo actual.

AgregarCadenaOk

$$\Delta Nodo$$
$$nuevo? : Bloque$$
$$m! : Reporte$$

$$nuevo?.prevblock = (last\ cadena).fingerprint$$
$$nuevo?.fingerprint = getFingerprint(nuevo?)$$
$$cadena' = cadena \frown \langle nuevo? \rangle$$
$$m! = Okay$$

B. Reemplazar cadena

Hay ocasiones en las que es necesario que el nodo escoja entre dos posibles cadenas de bloques debido a que dos nodos distintos han creado bloques con el mismo índice, en estos casos es necesario reemplazar la cadena completa por la cadena más larga de bloques disponible.

ReemplazarCadena

$$\Delta Nodo$$
$$nuevaCadena? : \text{seq } Bloque$$

$$\#nuevaCadena? > \#cadena$$
$$cadena' = nuevaCadena?$$

References

[1] M. Crosby *et al*, "BlockChain Technology: Beyond Bitcoin," *Applied Innovation Review*, *Berkeley*., no. 2, pp. 6-19, Jun. 2016.

[2] https://medium.com/programmers-blockchain/create-simple-blockchain-java-tutorial-from-scratch-6eeed3cb03fa

[3] L. Hartikka, "A blockchain in 200 lines of code," Medium, 29-Dec-2017. [Online]. Available: https://medium.com/@lhartikk/a-blockchain-in-200-lines-of-code-963cc1cc0e54. [Accessed: 20-Oct-2019].