Aspectos teóricos de las blockchain en sistemas de votación [*]

Índice

1. Introducción **2**

2. Sistemas blockchain **2**
2.1. Que son . . . 2
2.2. Sus vulnerabilidades . . . 2

3. Métodos formales en los sistemas de blockchain **3**
3.1. Formalizando consenso . . . 3
3.2. Creando mejores contratos inteligentes . . . 3

4. Voto electrónico **3**
4.1. Antecedentes . . . 3
4.2. Votando con blockhains . . . 3
4.3. Métodos Formales en la Votación Electronica . . . 3
4.3.1. Homomorphic . . . 3
4.3.2. MixNets . . . 3

5. Caso especifico **3**
5.1. El cálculo ρ . . . 4
5.2. Rholang . . . 4
5.3. PoS en Rholang . . . 4

6. Conclusión **4**

7. Bibliografía **4**

[*]Material desarrollado bajo el proyecto UNAM-2024.

1. Introducción

2. Sistemas blockchain

2.1. Que son

Los sistemas de blockchains o tecnologías de blockchain son el uso conjunto/simultáneo de: 1. la estructura de datos de cadena de bloques(*blockchain*), 2. algún algoritmo distribuido de consenso, 3. criptografía de llave pública y, recientemente, 4. contratos inteligentes (*smart contracts*); en una red *peer-to-peer*. Todos reunidos forman una red anonímа.

(*Nota) pendiente a desarrollar: Motivaciones economicas desde su fundación. Moneda digital y descentralizada.

Sabemos que para n agentes en una red distribuida el sistema puede tolerar hasta $\frac{n}{4}$ agentes maliciosos, dicho protocolo asume que se conocen todos los agentes. La *blockchain* como estructura de datos es creada con el fin de resolver el problema del consenso Bizantino tolerante a fallos para un número arbitrario de agentes en una red anónima.

El primer acercamiento al desarrollo de estas fue la creación de las marcas de tiempo (*timestamps*) que denoten el momento en el que algúna operación fue efectuada. La respuesta se encontró en la criptografía. **(*Nota) pendiente a desarrollar:** Que es un Hash

Como ya mencionamos la motivación inicial de los sistemas blockchain era crear una moneda digital. Así, llamamos bloque a la información de las operaciones/transacciones con la supuesta moneda. Para poder preservar o simular el tiempo de forma efectiva falta alimentar el hash con otra cosa. Cada marca de tiempo debe probar que la información de cada bloque existió y de algúna forma estar vinculada con el pasado. Salvo por el bloque inicial, único bloque cuya marca de tiempo es el hash con la información inicial, el resto de bloques creará su marca de tiempo al hacer un hash sobre la información del bloque y la marca de tiempo anterior. Al hash ser único y seguiendo este procedimiento para cada bloque/tiempo nuevo es que tenemos la estructura de datos blockchain.

(*Nota) pendiente a desarrollar: Mencionar que se pueden tener varias ramas pero por que la más larga es la confiable. Se pueden usar las blockchains sin tener sistemas distribuidos pero como ya vimos no es el motivo de su creación.

(*Nota) pendiente a desarrollar: Redes p2p con la estructura de datos y criptografía de llave pública nos permiten transmitir información cifrada y anónima para cada nodo. Ventajas sobre redes cliente-servidor: Cualquier agente puede desconectarse de la red, es descentralizado, encaminar la discusión al problema del consenso

(*Nota) pendiente a desarrollar: Introducir el rol de verificadores: encargados de crear nuevos bloques, y el llamado *zero block* con sus funcionalidades: primeros verificadores, configuraciones de red, primeras monedas. Ver que existen diferentes algoritmos de consenso en las BC: **PoW, PoS**, PoT

(*Nota) pendiente a desarrollar: *SmartContract: Programas que ejecutan los nodos de los sistemas de blockchain. Permiten automatizar procesos de las blockchains.* Hablar de sus diferencias con el paso de mensajes y por que Bitcoin funcionaba sin los contratos. Por que son utiles y hablar de Solidity.

Las notas de la pendiente sección se siguen de [9] y [1].

2.2. Sus vulnerabilidades

En los sistemas de blockchain existen dos tipos de vulnerabilidades. El primero esta asociado con el protocolo mismo. En este tipo se incluyen: uso de funciones criptográficas mal implementadas y formas de atacar los protocolos de consenso para controlar la red. Por otro lado el segundo surge de añadir smart contracts al sistema. Al ser Touring Completos crean problemas que dependen del protocolo de consenso pero también en las billeteras digitales y sus constantes interacciones.

(*Nota) pendiente a desarrollar: Hablar de las vulnerabilidades a las que estan expuestos los algoritmos de protocolos y escenarios donde se pierde el control de la blockchain.

(*Nota) pendiente a desarrollar: Hablar 1) del caso DAO, como es que se podía ejecutar código inseguro/no verificado, y 2) de como las primitivas de la EVM pueden crear bugs que un adversario puede utilizar.

Para ambas ideas pendientes de esta sección se utilizará. [8], [13]

3. Métodos formales en los sistemas de blockchain

3.1. Formalizando consenso

(*Nota) pendiente a desarrollar: Hablaremos de las ventajas ecólogicas de PoS sobre PoW, de como por la naturaleza de PoS el adversario se compromete financiaremente, y sobre la primera formalización de PoS en coq [12].

3.2. Creando mejores contratos inteligentes

(*Nota) pendiente a desarrollar: Clasificación que la academía le ha dado a los problemas, antes mencionados, en los smart contracts. 1) Creación de lenguajes más seguros, 2) Analísis de programa (low level, seguridad), 3) Criptografía y 4) Nuevos protocolos [8][13].

4. Voto electrónico

4.1. Antecedentes

(*Nota) pendiente a desarrollar: Motivaciones de voto electronico. Casos donde se ha votado electronicamente (casos cliente-servidor y supuestamente con blockchain), hablar brevemente de los sistemas donde se realizaron. Suiza chvote mixnets. Virginia y Russia con blockchain.

4.2. Votando con blockhains

(*Nota) pendiente a desarrollar: Hablar del esquema de votos, la llamada independencia de software y como las blockchains cubren de mejor forma el e-vote [3][6][7]

(*Nota) pendiente a desarrollar: Traer a discusión lo antes explicado con el articulo del MIT [11].

4.3. Métodos Formales en la Votación Electronica

(*Nota) pendiente a desarrollar: Los dos casos exitosos y completos que existen sobre verificación en herramientas de votación han sido en sistemas cliente-servidor. Hablaremos de ellos dos y de que manera podríamos llevarlos a la blockchain y como cubre problemas planteados por el MIT[5] [4][11].

4.3.1. Homomorphic

4.3.2. MixNets

5. Caso especifico

(*Nota) pendiente a desarrollar: Una vez vista la importancía de los smartcontracts y se sus vulnerabilidades en el sistema más popular; veremos el caso de Rholang, programa concurrente cuyo fundamento es el cálculo ρ, una extensión del cálculo π. El lenguaje inicialmente creo su sistema de blockchain, llamado RChain, pero también puede ser utilizado en otros sistemas. Se introducira el cálculo rho, veremos las primitivas del lenguaje/mostrar ejercicios que puede resolver (los del tutorial más unos propios) y se mostrara como se puede llegar a un consenso PoS (el consenso verificable en coq) [2][10]

5.1. El cálculo ρ

5.2. Rholang

5.3. PoS en Rholang

6. Conclusión

7. Bibliografía

Referencias

[1] Vitalik Buterin. Ethereum: A next-generation smart contract and decentralized application platform., 2014.

[2] Horatiu Cirstea and Claude Kirchner. ρ-Calculus. Its Syntax and Basic Properties. Intern report 98-R-218 —— cirstea98a, 1998. Rapport interne.

[3] Milad K. Ghale, Rajeev Goré, and Dirk Pattinson. A formally verified single transferable voting scheme with fractional values. In Robert Krimmer, Melanie Volkamer, Nadja Braun Binder, Norbert Kersting, Olivier Pereira, and Carsten Schürmann, editors, *Electronic Voting*, pages 163–182, Cham, 2017. Springer International Publishing.

[4] Thomas Haines, Rajeev Gore, and Bhavesh Sharma. Did you mix me? formally verifying verifiable mix nets in electronic voting. Cryptology ePrint Archive, Paper 2020/1114, 2020.

[5] Thomas Haines, Dirk Pattinson, and Mukesh Tiwari. Verifiable homomorphic tallying for the schulze vote counting scheme. In Supratik Chakraborty and Jorge A. Navas, editors, *Verified Software. Theories, Tools, and Experiments*, pages 36–53, Cham, 2020. Springer International Publishing.

[6] Uzma Jafar, Mohd Aziz, and Zarina Shukur. Blockchain for electronic voting system—review and open research challenges. *Sensors*, 21:5874, 08 2021.

[7] Uzma Jafar, Mohd Juzaiddin Ab Aziz, Zarina Shukur, and Hafiz Adnan Hussain. A cost-efficient and scalable framework for e-voting system based on ethereum blockchain. In *2022 International Conference on Cyber Resilience (ICCR)*, pages 1–6, 2022.

[8] Andrew Miller, Zhicheng Cai, and Somesh Jha. Smart contracts and opportunities for formal methods. In Tiziana Margaria and Bernhard Steffen, editors, *Leveraging Applications of Formal Methods, Verification and Validation. Industrial Practice*, pages 280–299, Cham, 2018. Springer International Publishing.

[9] Satoshi Nakamoto. Bitcoin: A peer-topeer electronic cash system.

[10] Rholang org. Rholang web page.

[11] Sunoo Park, Michael Specter, Neha Narula, and Ronald L Rivest. Going from bad to worse: from Internet voting to blockchain voting. *Journal of Cybersecurity*, 7(1):tyaa025, 02 2021.

[12] Søren Eller Thomsen and Bas Spitters. Formalizing nakamoto-style proof of stake. In *2021 IEEE 34th Computer Security Foundations Symposium (CSF)*, pages 1–15, 2021.

[13] Palina Tolmach, Yi Li, Shang-Wei Lin, Yang Liu, and Zengxiang Li. A survey of smart contract formal specification and verification, 2021.