2017

Bitcoins

SEGURIDAD INFORMÁTICA

ORQUÍDEA SEIJAS

Tabla de contenido

1. Introducción

Este trabajo monográfico permite la primera toma de contacto de la alumna con algunas tecnologías o temas directamente relacionados con la seguridad informática. En este caso, se estudiará la criptomoneda, protocolo y software: bitcoin.

En este documento se describirán las principales características del bitcoin en el apartado 2. Características Generales. A continuación, se conocerá el funcionamiento de las transacciones en 3.Transacciones. Con el fin de explicar más en profundidad el funcionamiento interno del protocolo Bitcoin, se han añadido los apartados 4. Timestamp Server, 5. Sistema de prueba de trabajo (Proof-of-work system), 6. Red, 7. Espacio de disco necesario, 8. Verificación de pago simplificada y 9. Privacidad. Finalmente, se ha realizado una conclusión sobre lo aprendido a través de la realización de la memoria en el apartado 10. Conclusión.

2. Características generales

Bitcoin es una palabra utiliza para describir una criptomoneda, un protocolo y un software. Bitcoin se caracteriza por ser descentralizado, lo que supone que no está respaldado por ningún gobierno o emisor central. Así pues, se diferencia de las monedas tradicionales como el euro, ya que este depende del Banco Central Europeo y está respaldado por todos los gobiernos de los miembros de la Unión Europea en cuyas naciones se usa. Este proyecto se concibió como uno con capacidad de cambiar el paisaje económico mundial, ya que antes de su puesta en funcionamiento, en 2009, no era posible realizar pagos en comercio electrónico si no era a través de entidades centralizadas de confianza.

3. Transacciones

Una moneda electrónica es una cadena de firmas digitales. El uso del resumen digital y de claves públicas es fundamental para el funcionamiento de la moneda. Esto es porque cada propietario transfiere la moneda al siguiente propietario a través de la firma del hash de la transacción previa y la clave pública del siguiente, añadiendo esta información al final de la moneda. El receptor del pago puede verificar las firmas con el fin de verificar la cadena de propiedad de la moneda. Esto se ve ilustrado en la siguiente imagen:

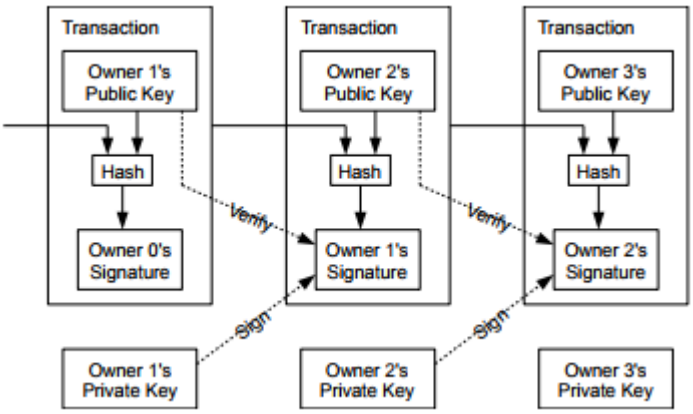


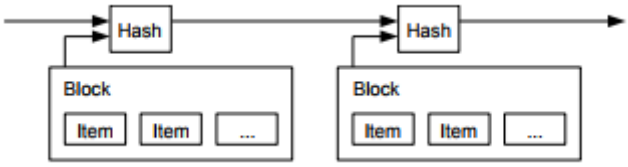
ILUSTRACIÓN 1

Esta moneda tendría asociado el problema de que es imposible saber si el propietario que realiza la transferencia ya gastó lo que pretende transferir. Con el fin de evitar que una autoridad central

realice esta comprobación, lo que haría que los bitcoins fueran como las monedas dependientes de bancos, se buscó una nueva solución. Esta solución simplemente supone comprobar que el propietario inicial no se gastó previamente lo que pretende transferir. La única forma de realizar esta comprobación es poder conocer todas las transacciones. Para ello, normalmente sería necesaria la autoridad central, que es la conocedora de todas las transacciones. Sin embargo, si las transacciones son anunciadas públicamente y existe un sistema que permite a los propietarios establecer el orden en el que se reciben las transacciones, no es necesario. Así pues, el receptor de una transacción únicamente necesita que, en el momento de la transacción, los participantes coincidan en que fue el primero en recibirla.

4. Timestamp Server

Para solucionar el problema descrito en el apartado anterior, los creadores de bitcoin proponen una solución basada en el uso de una red peer-to-peer que empieza con un *Timestamp Server*. Este servidor funciona realizando un resumen digital sobre un conjunto de ítems a los que se les marca con el *timestamp*. Esta marca sirve para probar que los datos existen en el momento de la realización del resumen digital: de lo contrario, sería imposible que estuvieran en él. Cada *timestamp* incluye a los *timestamp* previos, lo que forma una cadena que permite que cada *timestamp* adicional refuerce a los *timestamps* previos:



5. Sistema de prueba de trabajo (*Proof-of-work system*)

Para implementar un servidor *timestamp* distribuido en una arquitectura p2p, es necesario utilizar un sistema de prueba de trabajo. En general, un sistema de prueba de prueba de trabajo es un sistema en el que, para evitar comportamiento no deseado, se requiere que el cliente realice algún tipo de trabajo (factible) que tenga cierto coste y que pueda ser verificado fácilmente por el servidor.

En el caso concreto de los bitcoins, se utiliza el sistema *Hashcash* para la generación de bloques. Para que un bloque sea aceptado, se debe completar una prueba de trabajo que cubra toda la información del mismo. La dificultad de esta prueba se ajusta para que una red se vea limitada a generar, como mucho, un bloque cada diez minutos.

La prueba más usada está basada en SHA-256 y fue introducida como parte de Bitcoin. Esta prueba consiste en encontrar el número requerido de ceros al inicio del hash en una cadena concreta, añadiendo un valor entero a través de un nonce, que es un campo de los bloques, y sumando uno cada vez que se realiza el hash. Es decir, partiendo de una cadena de caracteres y modificándola añadiendo, e incrementando de cada vez, un entero, es posible obtener un valor hash con un número determinado de ceros al principio, que es el que solicita el servidor.

La prueba de trabajo, además, permite identificar cada CPU como un nodo. Puesto que es necesario que la mayoría de los nodos (o participantes) coincidan en si una transacción ha sido realizada previamente o no, se le concede un voto a cada CPU.

Cada bloque contiene el hash predecesor, por lo que cada bloque tiene una cadena de bloques que, en conjunto, son una enorme cantidad de trabajo. Cambiar un bloque supone volver a generar todos sus predecesores y rehacer el trabajo que contienen, lo que previene alteraciones.

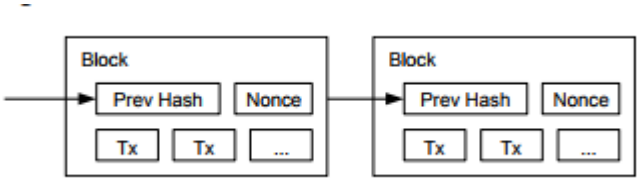


ILUSTRACIÓN 2: CADENA DE BLOQUES

6. Red

El funcionamiento de la red se comentará a continuación. En primer lugar, las transacciones nuevas se emiten a todos los nodos de la red. A su vez, cada nodo recoge nuevas transacciones en un bloque y trabaja en encontrar una prueba ce trabajo difícil para su bloque. Cuando un nodo encuentra una prueba de trabajo, la emite a todos los nodos. Los nodos aceptan el bloque solo si todas las transacciones son válidas y no han sido gastadas. Los nodos expresan su aprobación trabajando en crear el siguiente bloque en la cadena, utilizando el hash del bloque aceptado como el resumen digital anterior.

Los nodos siempre consideran que la cadena más larga es la correcta y continuarán trabajando con el fin de seguir extendiéndola. Si dos nodos emiten a los demás de diferentes versiones del mismo siguiente bloque simultáneamente, los nodos receptores trabajarán en el primer bloque que recibieran, pero guardarán la otra cadena, por si esta se vuelve más larga. Los nodos cambiarán de cadena si, cuando se descubra el siguiente trabajo, la cadena resultante se vuelve más larga.

La emisión de nuevas transacciones no tiene porqué alcanzar a todos los nodos, con que alcance a un número considerable es suficiente, ya que una vez que esté en los nodos se convertirá en parte de un bloque. La emisión de bloques es tolerante a fallos, puesto que si un nodo no recibe un bloque, cuando reciba el siguiente lo pedirá, ya que se dará cuenta de que le hace falta.

7. Espacio de disco necesario

Una vez que la última transacción de una moneda se vea oculta por suficientes bloques, las transacciones previas pueden ser descartadas para salvar memoria. Con el fin de realizar esta tarea sin romper el resumen digital del bloque, las transacciones se resumen digitalmente en un árbol Merkle. En el resumen del bloque únicamente se incluye la raíz de este árbol. Los bloques con más antigüedad se pueden compactar recortando las ramas del árbol. Además, los resúmenes digitales externos no necesitan ser almacenados. Así pues, tras aplicar el árbol de Merkle y tras recortar hashes innecesarios, se obtendría la siguiente ilustración:

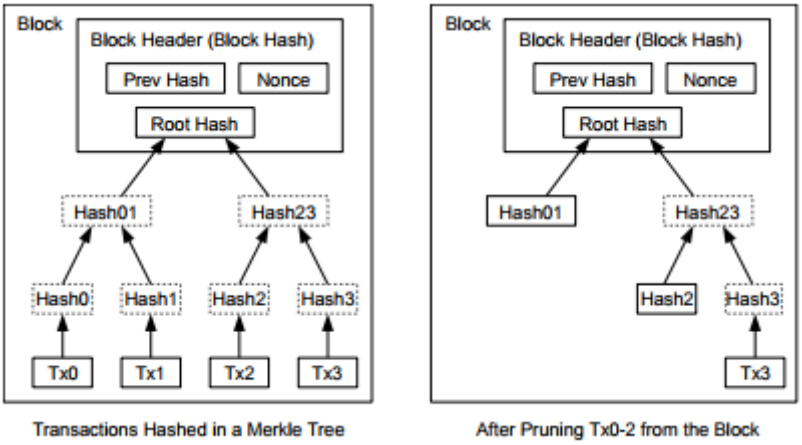


ILUSTRACIÓN 3

a. Descripción del funcionamiento de un árbol Merkle

Con el fin de comprender plenamente el funcionamiento del proceso explicado en la *Ilustración* 3 es necesario comprender el funcionamiento de un árbol binario de hash Merkle. Para este caso concreto, los nodos se obtienen siguiendo los pasos que se explicarán a continuación. En primer lugar, las hojas del árbol se obtienen calculando los hashes dobles de cada transacción que va a formar parte del árbol, tal y como se muestra en la siguiente ilustración:

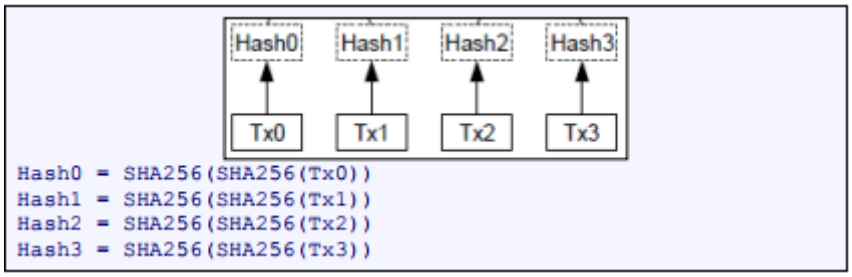


ILUSTRACIÓN 4

En segundo lugar, para obtener los nodos intermedios, los resúmenes digitales calculados en el nivel inferior se agrupan por parejas y se calcula el hash correspondiente a la concatenación de los hashes. Este proceso se repite recursivamente, lo que implica que cada nivel tendrá la mitad de nodos que el anterior. Finalmente, el nodo raíz se calcula de la misma forma que los anteriores. Gracias a esta estructura de datos es posible tener un bloque sin transacciones, lo que permite almacenar con más facilidad la información.

b. Espacio ocupado por un bloque sin transacciones

El cabecero de un bloque sin transacciones ocupa unos 80 bytes. Puesto que se genera, como mucho, un bloque cada 10 minutos, anualmente se producen unos $80 * 6 * 24 * 365 = 4204800$ bytes (4.2 MB). Dado el tamaño medio de la RAM de los ordenadores actuales, entre 2 y 8 GB, este gasto en almacenamiento es despreciable.

8. Verificación de pago simplificada

Para verificar pagos sin el uso de un nodo de red entero, un usuario únicamente necesita mantener una copia de los cabeceros de los bloques de la cadena de prueba de trabajo más larga que pueda obtener y obtener la rama del árbol Merkle enlazando la transacción con el bloque que tiene el *timestamp* en él. El usuario deberá enlazar la transacción a la cadena, con el fin de saber si un nodo de la red la ha aceptado y, por lo tanto, ha añadido bloques posteriormente.

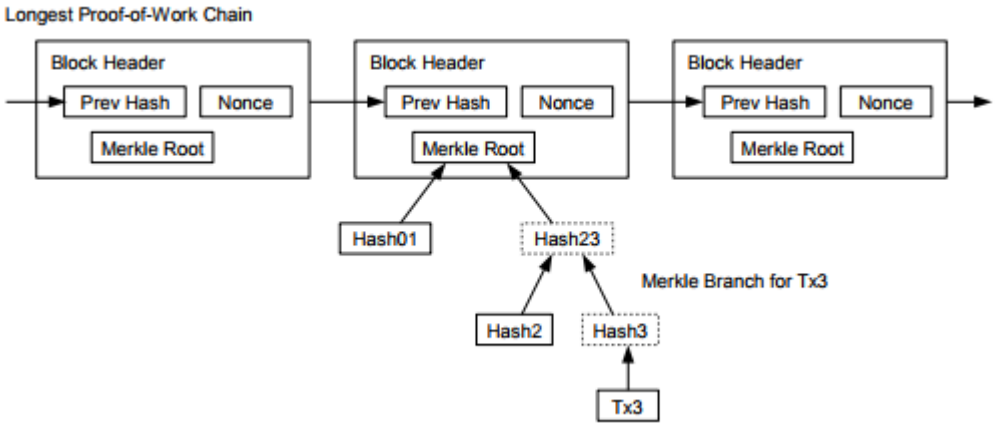


ILUSTRACIÓN 5: CADENA DE PRUEBA DE TRABAJO PARA REALIZAR LA VERIFICACIÓN DE PAGO SIMPLIFICADA

Este método de verificación es seguro siempre y cuando nodos honestos controlen la red. Esto es porque este método simplificado puede verse engañado por transacciones falsas, fabricadas por un atacante mientras este cuente con poder en la red. Para protegerse de este atacante, se podrían aceptar alertas de los nodos de la red conforme se han detectado bloques inválidos.

9. Privacidad

La necesidad de anunciar todas las transacciones públicamente podría dar la impresión de que limita la privacidad de los usuarios. Sin embargo, la privacidad se mantiene ya que no es necesario que las claves públicas se asocien a alguien. Es decir, se sabe que un usuario *x* está mandando una cantidad *y* a un usuario *z*, pero no se sabe quiénes son estos usuarios. Como medida de prevención adicional, se debería crear un nuevo par de claves para cada transacción con el fin de evitar la conexión de una clave pública a un usuario concreto.

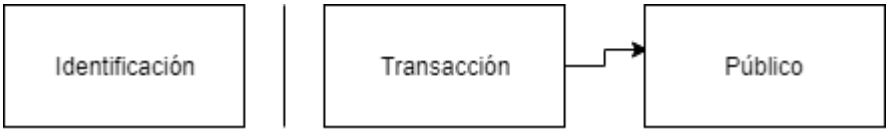


ILUSTRACIÓN 6

10. Conclusión

Las firmas digitales son fundamentales para que el protocolo Bitcoin funcione apropiadamente. En primer lugar, permiten tener control sobre la propiedad de la moneda. En segundo lugar, a través

de ellas se construye la arquitectura distribuida, puesto que es necesario realizar comprobaciones sobre las transacciones, ya que el sistema de pruebas de trabajo está basado en SHA-256. Además, la red de nodos involucrados se crea a partir de transacciones y bloques formados por las mismas.

Por el momento, los *bitcoins* no han cambiado radicalmente el panorama económico, pero han demostrado que es posible realizar transacciones de una forma completamente nueva, sin intermediarios tradicionales como la banca. Es por ello por lo que es muy difícil negar que han supuesto un antes y un después. Sin embargo, de no existir funciones *hash*, o funciones de resumen digital, no podría existir esta criptomoneda ni su protocolo asociado tal y como la conocemos.

Referencias

- J. Brito. A. Castilllo. *BITCOIN: A primer for Policymakers*. 2013. https://www.mercatus.org/system/files/Brito_BitcoinPrimer.pdf [Última visita: 23/11/2017]
- S. Nakamoto. Bitcoin: *A Peer-To-Peer Electronic Cash System*. 2009. https://bitcoin.org/bitcoin.pdf [Última visita: 23/11/2017]
- J. Díaz Vico. A. Sánchez Aragó. INTECO. *Bitcoin: una moneda criptográfica*. 2013 https://www.certsi.es/sites/default/files/contenidos/estudios/doc/int_bitcoin.pdf [Última visita: 24/11/2017]
- Wikipedia. *Sistema de prueba de trabajo*. 2017. https://es.wikipedia.org/wiki/Sistema_de_prueba_de_trabajo [Última visita: 25/11/2017]
- Bitcoinwiki. *Nonce*. 2017 https://es.bitcoin.it/wiki/Nonce [Última visita: 23/11/2017]
- Bitcoinwiki. *Proof of work*. 2017. https://en.bitcoin.it/wiki/Proof_of_work [Última visita: 25/11/2017]
- Wikipedia. *Bitcoin*. 2017. https://es.wikipedia.org/wiki/Bitcoin [Última visita: 25/11/2017]