#Bitcoin

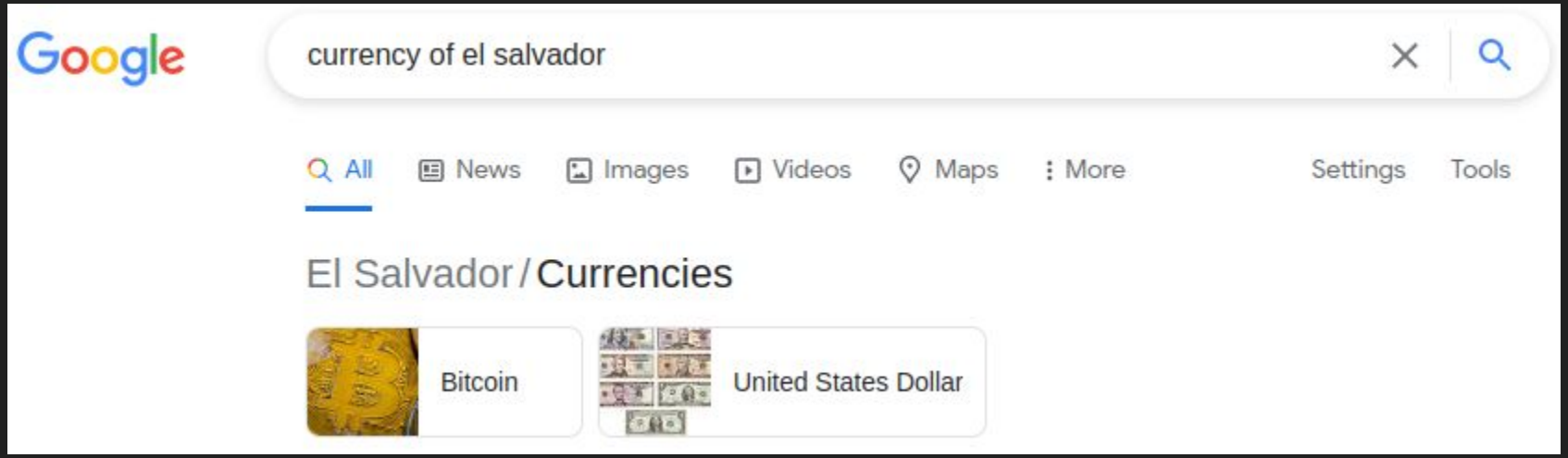
¿Cómo funciona el dinero mágico de internet?

Javier G. Montoya S.

¿Por qué nos debería importar?

¿Por qué nos debería importar?

Porque es una de las monedas oficiales de El Salvador 

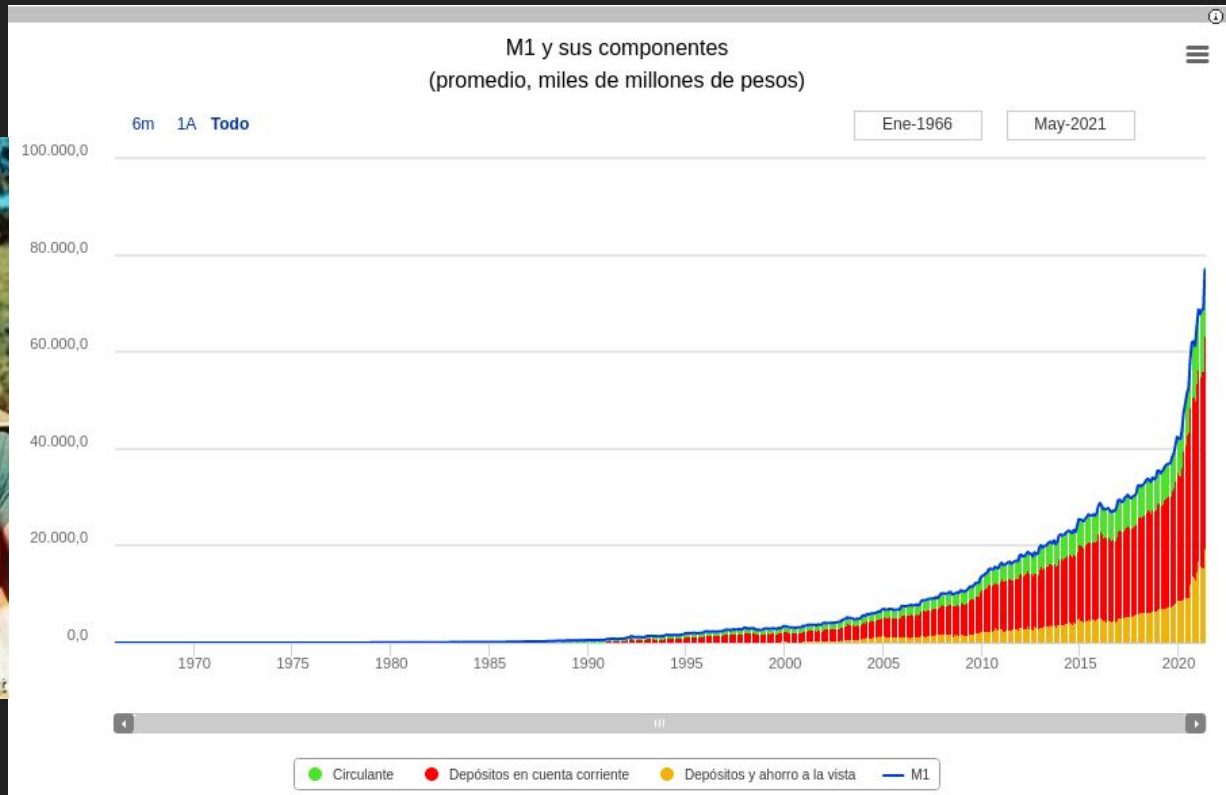


¿Por qué nos debería importar?

Porque es una de las monedas oficiales de El Salvador 

Devaluación de la moneda

Devaluación de la moneda



¿Por qué nos debería importar?

Porque es una de las monedas oficiales de El Salvador 

Devaluación de la moneda

Derechos humanos

Derechos humanos

Human Rights Foundation

Suena bonito pero, ¿cómo funciona?

Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

Suena bonito pero, ¿cómo funciona?

Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

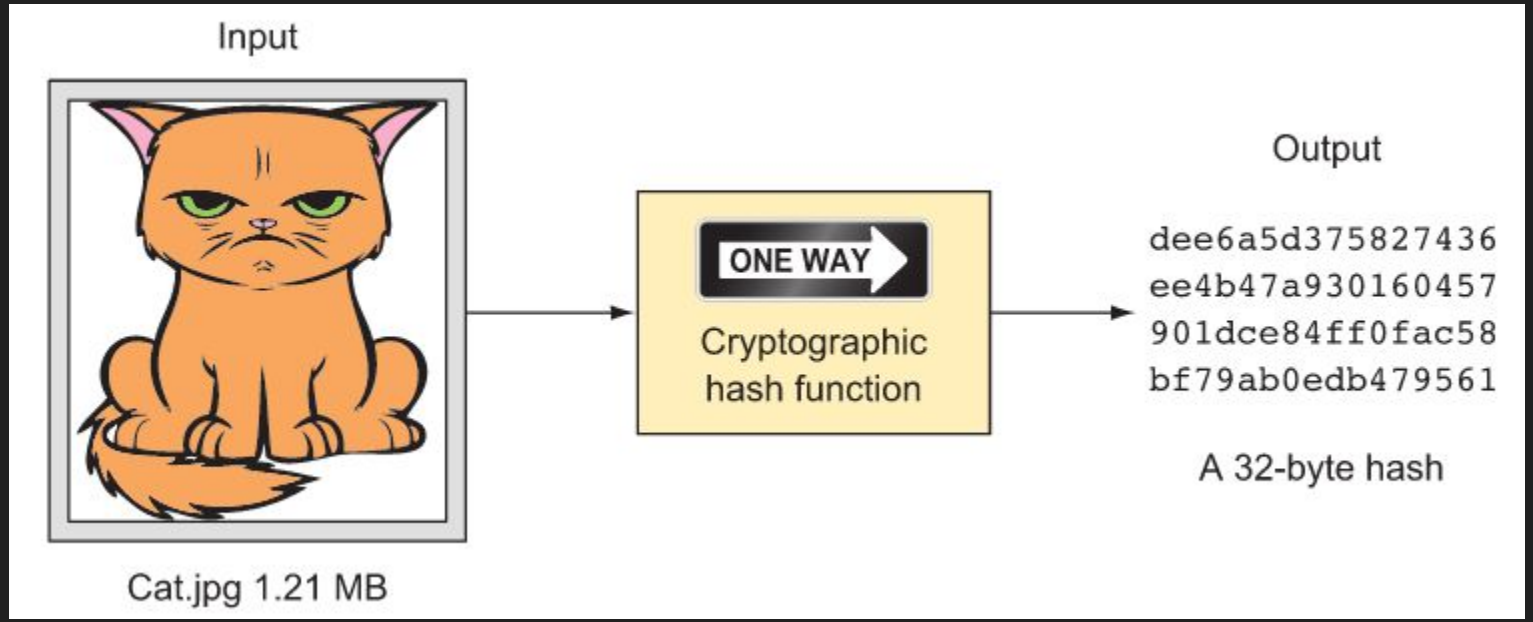
Ingredientes previos

1. Funciones de hash criptográficas
2. Firmas digitales

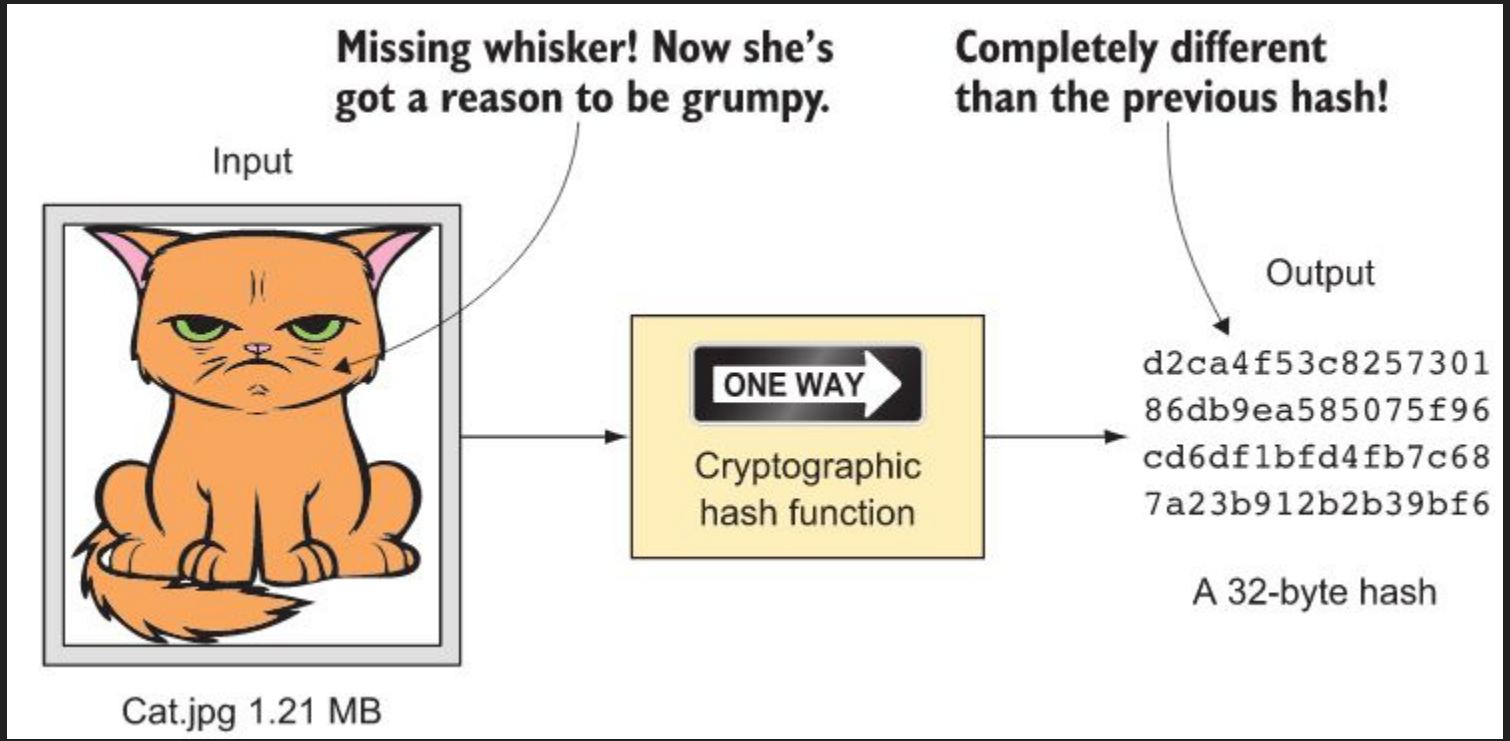
Ingredientes previos

1. Funciones de hash criptográficas
2. Firmas digitales

Funciones de Hash criptográficas



Funciones de Hash criptográficas



Hash anterior: dee6a5d375827436ee4b47a9301604579 01dce84ff0fac58bf79ab0edb479561

Funciones de Hash criptográficas

¿De qué me sirve eso?

Funciones de Hash criptográficas

¿De qué me sirve eso?

Un ejemplo: Verificar la integridad de un archivo (o sea, que no haya cambiado).

Funciones de Hash criptográficas

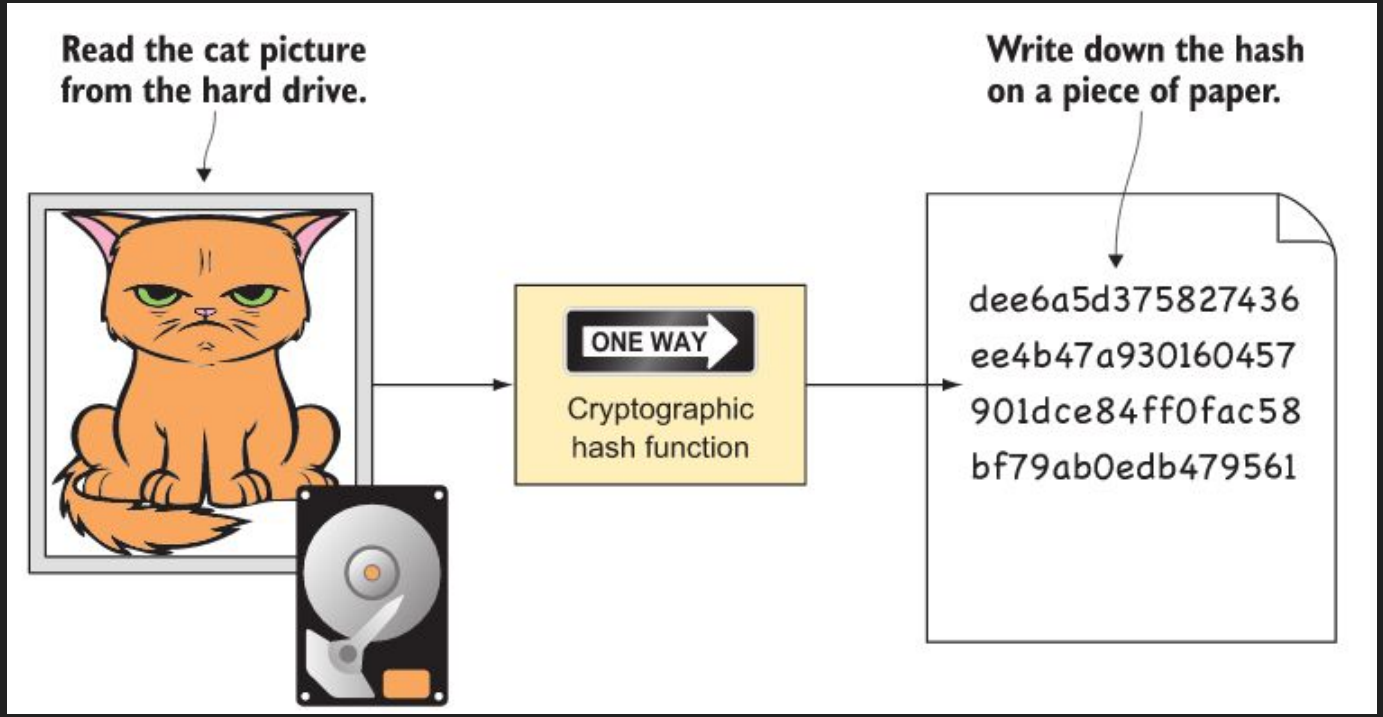
¿De qué me sirve eso?

Un ejemplo: Verificar la integridad de un archivo (o sea, que no haya cambiado).

¿Cómo?

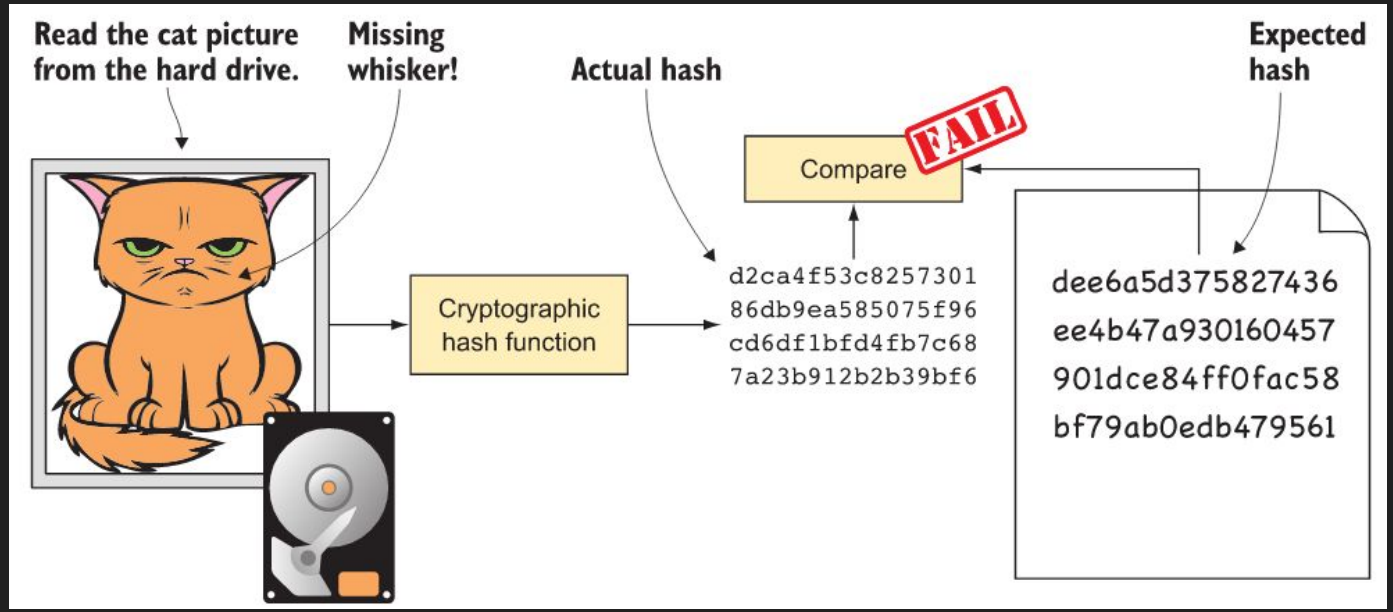
Funciones de Hash criptográficas

Primero calculamos el hash del archivo y lo anotamos en algún lado



Funciones de Hash criptográficas

Luego, cuando queramos ver el archivo, podemos verificar si ha sido modificado calculando el hash criptográfico de nuevo y comparándolo con el anterior



Funciones de Hash criptográficas

4 propiedades importantes de una buena función de hash criptográfica:

1. El mismo input siempre produce el mismo hash.
2. Inputs ligeramente distintos producen hashes muy diferentes.
3. El hash tiene un largo fijo (256 bits para SHA256).
4. La única forma de encontrar un input que resulte en cierto hash es a través de prueba y error.

Funciones de Hash criptográficas

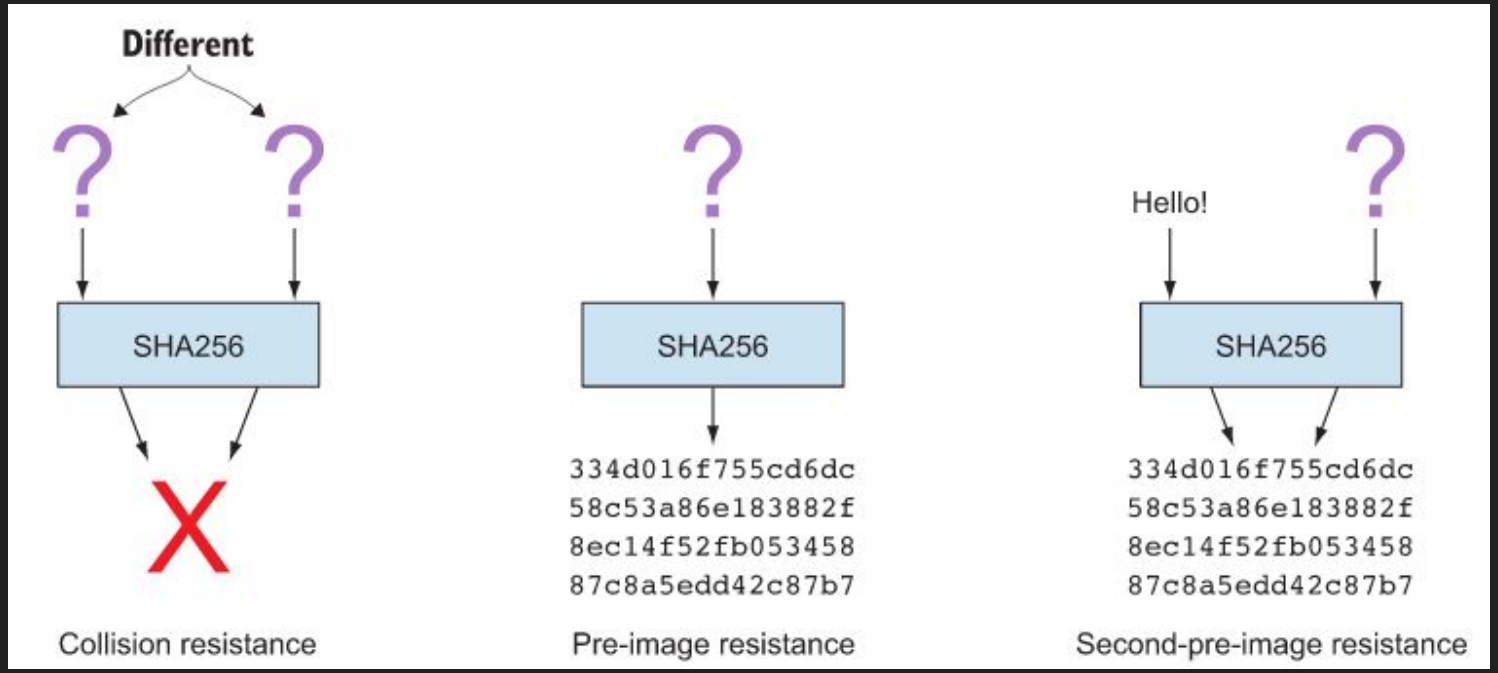
4 propiedades importantes de una buena función de hash criptográfica:

1. El mismo input siempre produce el mismo hash.
2. Inputs ligeramente distintos producen hashes muy diferentes.
3. El hash tiene un largo fijo (256 bits para SHA256).
4. La única forma de encontrar un input que resulte en cierto hash es a través de prueba y error.

Es decir que el output no entrega información acerca del input.

Funciones de Hash criptográficas

El output no entrega información acerca del input.

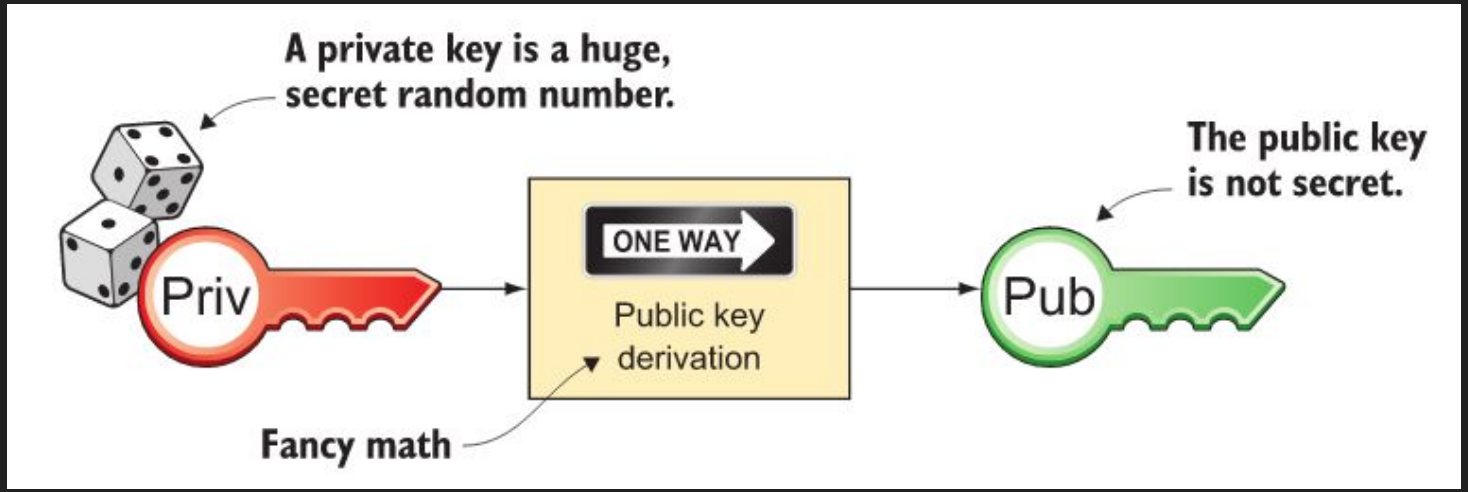


Ingredientes previos

1. Funciones de hash criptográficas
2. Firmas digitales

Firmas digitales

Pares de llaves: Llave privada y llave pública



Firmas digitales

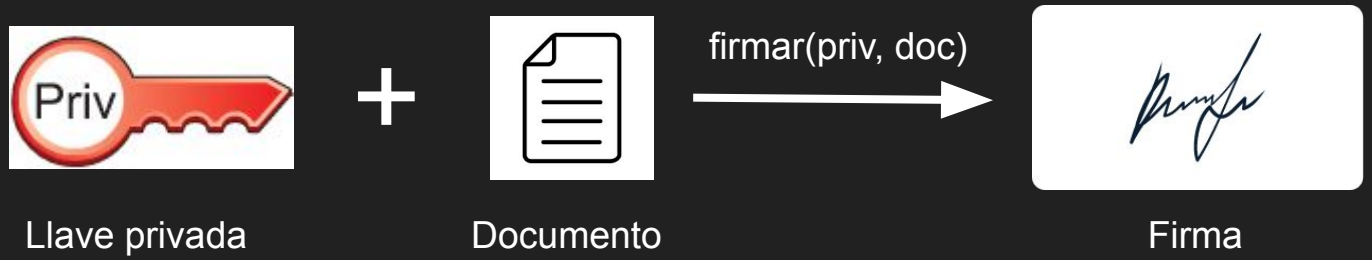
Dos funciones clave:

1. Firmar
2. Verificar

Firmas digitales

Dos funciones clave:

1. Firmar
2. Verificar



Firmas digitales

Firmar

El documento puede ser cualquier cosa.
No es posible producir la firma sin la llave privada.
La firma es única para el par (llave privada, documento): Si firmo otro documento con la misma llave privada, la firma resultante será distinta.

Firmas digitales

Dos funciones clave:

1. Firmar
2. Verificar

Llave pública

Firma

Si la firma fue producida para el documento con la llave privada correspondiente a ‘pub’

Firmas digitales

Dos funciones clave:

1. Firmar
2. Verificar

Firmas digitales

Firmar:

Llave privada

+

Documento

firmar(priv, doc)

Firma

Verificar:

Llave pública

+

Firma

verificar(pub, firma)

Firmas digitales

Firmar:

Llave privada | Documento | Firma

Verificar:

Llave pública | Firma

Es imposible producir una firma válida sin la llave privada, incluso teniendo la llave pública y acceso a otros mensajes firmados.

¿Qué podemos hacer con firmas digitales?

¿Qué podemos hacer con firmas digitales?

Alice

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Qué podemos hacer con firmas digitales?

Alice

Transfiere 100
a Bob

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Qué podemos hacer con firmas digitales?

Alice

Transfiere 100 a Bob

Verificar

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Qué podemos hacer con firmas digitales?

Alice

Transfiere 100
a Bob

Alice te
transfirió 100

Bob

Nombre	Monto	Llave pública
Alice	300	PKa
Bob	300	PKb

¿Y si eliminamos al intermediario?

Alice

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Y si eliminamos al intermediario?

Alice

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Y si eliminamos al intermediario?

Alice

Bob, te pago 100

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Y si eliminamos al intermediario?

A ver, déjame verificar

Alice

Bob, te pago 100

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Y si eliminamos al intermediario?

¿Es realmente Alice?
¿Alice tiene 100?

Alice

Bob, te pago 100

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Y si eliminamos al intermediario?

¿Es realmente Alice?
¿Alice tiene 100?

Alice

Bob, te pago 100

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Y si eliminamos al intermediario?

¿Es realmente Alice?
¿Alice tiene 100?

Alice

Bob, te pago 100

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Y si eliminamos al intermediario?

Genial, pago aceptado!

Alice

Bob, te pago 100

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿Y si eliminamos al intermediario?

Alice

Genial, pago aceptado!

Bob

Nombre	Monto	Llave pública
Alice	300	PKa
Bob	300	PKb

Nombre	Monto	Llave pública
Alice	300	PKa
Bob	300	PKb

¿Qué pasa si Alice no actualiza su tabla?

Alice

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

Bob

Nombre	Monto	Llave pública
Alice	300	PKa
Bob	300	PKb

¿Qué pasa si Alice no actualiza su tabla?

Y llega un nuevo participante

Alice

Charlie

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

Nombre	Monto	Llave pública
Alice	300	PKa
Bob	300	PKb

¿Qué pasa si Alice no actualiza su tabla?

Y llega un nuevo participante

Alice

Charlie

Bob

Nombre	Monto	Llave pública
Alice	400	PKa
Bob	200	PKb

¿A quién le creo?

Nombre	Monto	Llave pública
Alice	300	PKa
Bob	300	PKb

¿Qué pasa si Alice no actualiza su tabla?

Y llega un nuevo participante

Charlie

¿A quién le creo?

Bob puede mostrarle la transacción firmada

Bob

Nombre	Monto	Llave pública
Alice	300	PKa
Bob	300	PKb

¿Qué pasa si Alice no actualiza su tabla?

Y llega un nuevo participante

Charlie

Bob puede mostrarle la transacción firmada

Bob

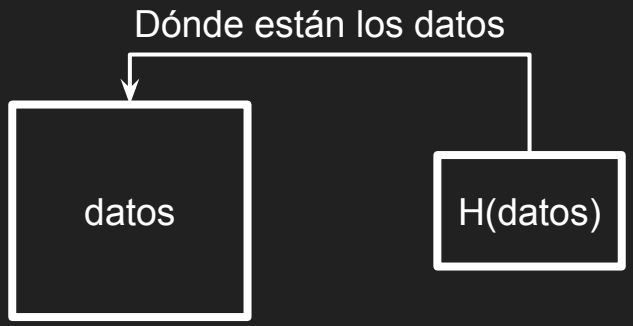
¿A quién le creo?

Pero Charlie no tiene cómo saber si Alice tenía ese monto en ese momento, ¿qué pasa si antes Alice le transfirió a otra persona?

Nombre	Monto	Llave pública
Alice	300	PKa
Bob	300	PKb

Nos falta algo

Punteros de hash

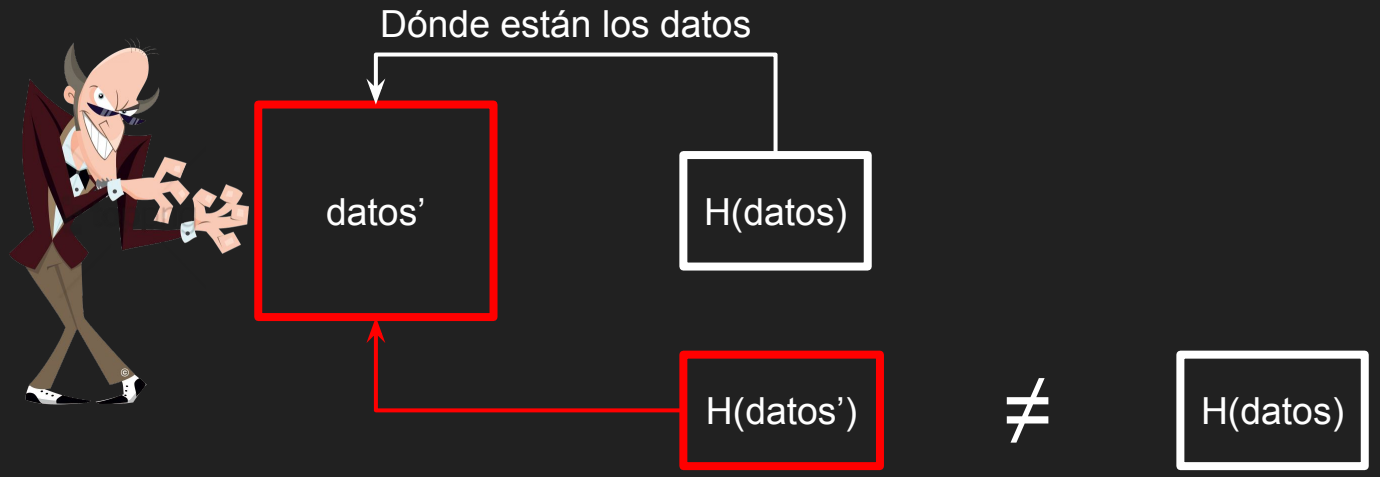


Punteros de hash

Punteros de hash

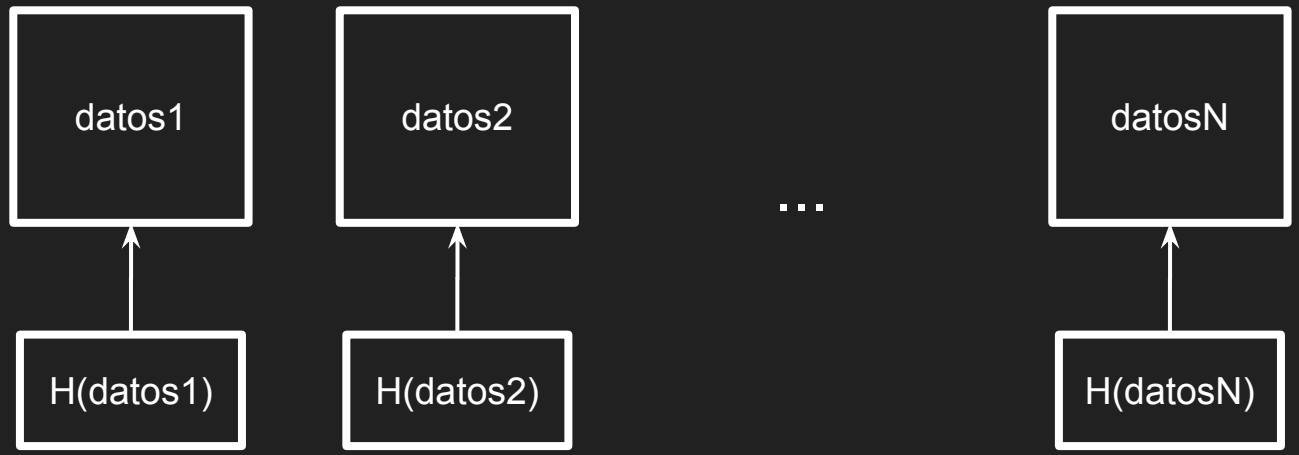
H(datos)

Punteros de hash

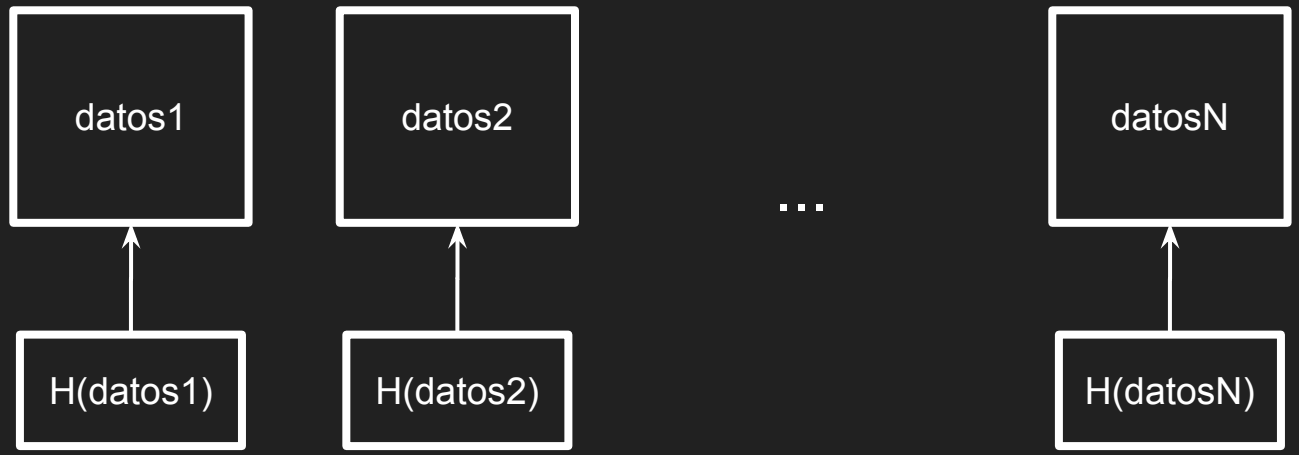


¿Qué pasa si tenemos que agregar datos de forma continua?

¿Qué pasa si tenemos que agregar datos de forma continua?



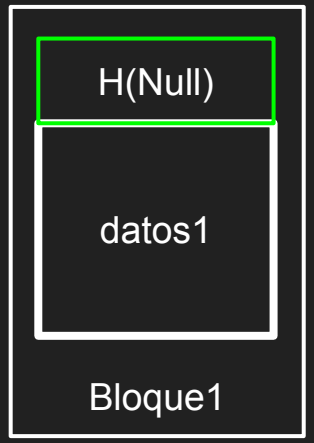
¿Qué pasa si tenemos que agregar datos de forma continua?



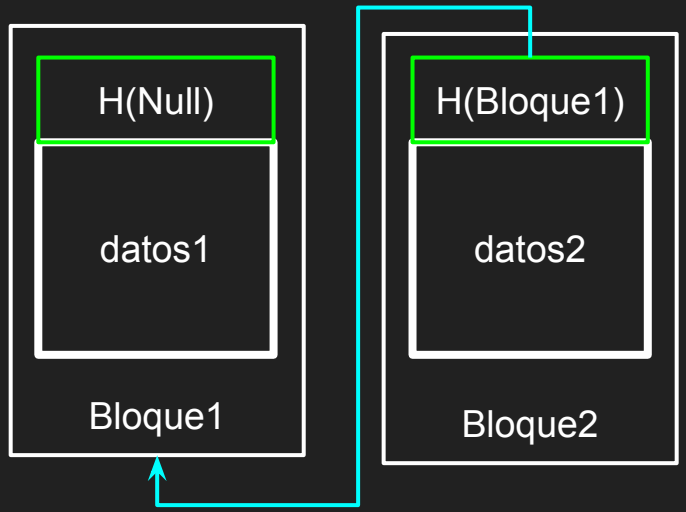
Necesitamos guardar N hashes para asegurarnos que nada cambió

Blockchain

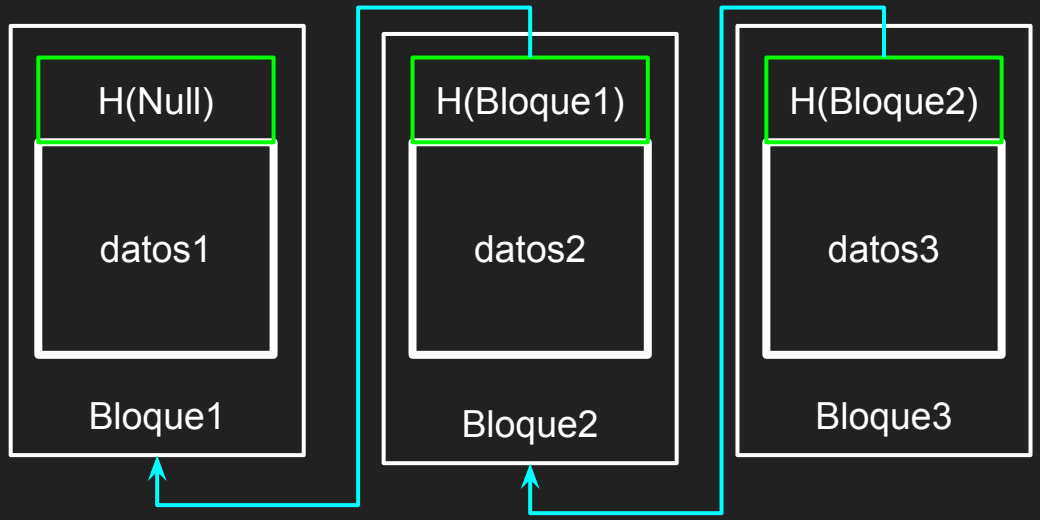
Blockchain



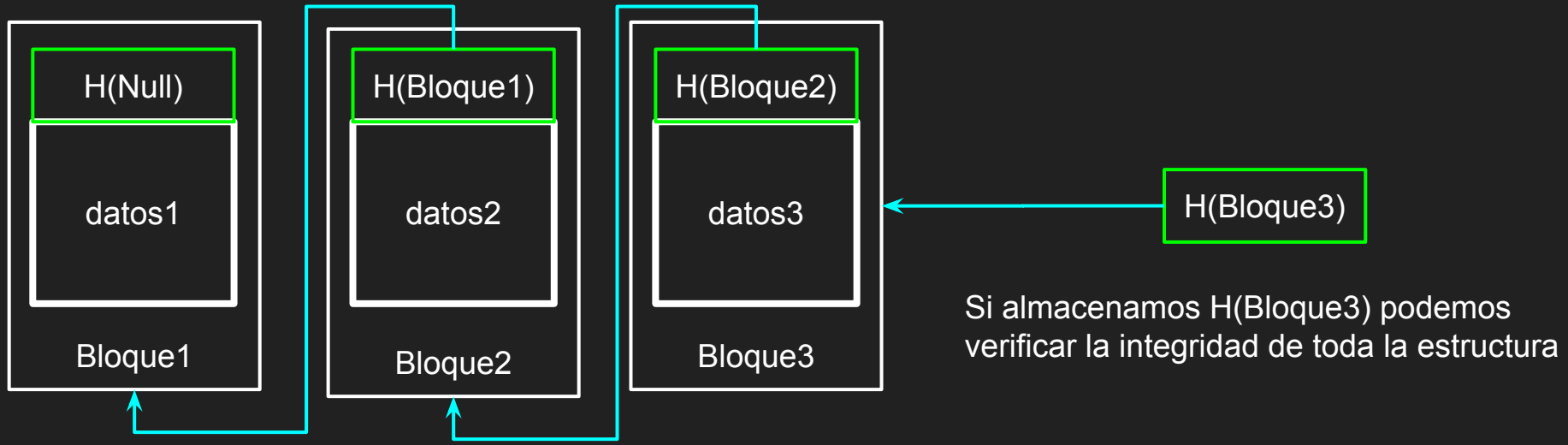
Blockchain



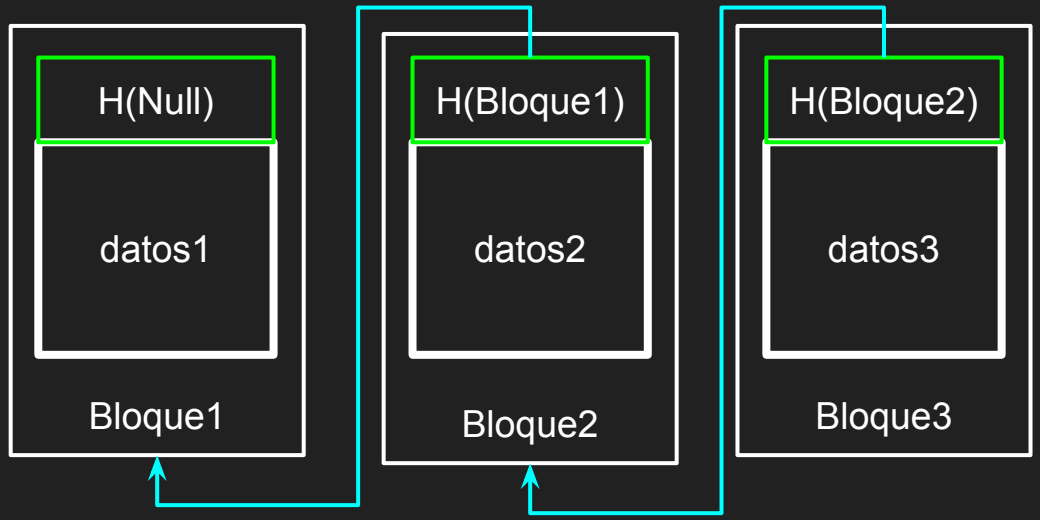
Blockchain


H(Null)
datos1
Bloque1
H(Bloque1)
datos2
Bloque2
H(Bloque2)
datos3
Bloque3

Blockchain

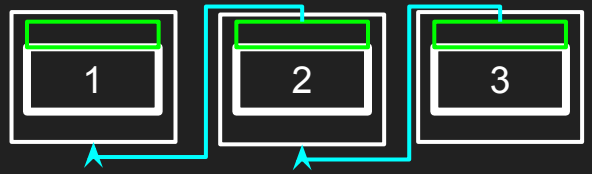


Si almacenamos H(Bloque3) podemos verificar la integridad de toda la estructura

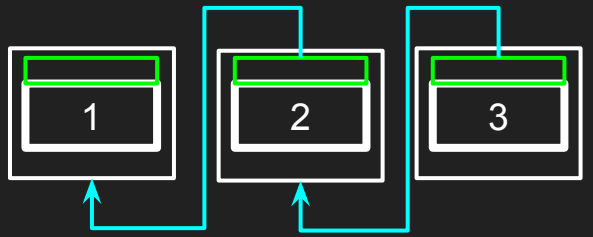


Los datos que guardamos en cada bloque son las transacciones.

Alice

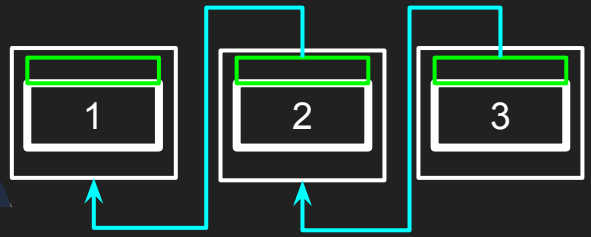


Bob

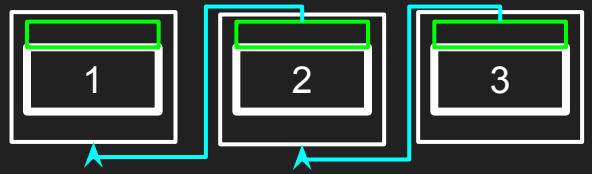


Reglas:
1. Cada participante guarda todos los bloques.
2. Cuando quieres pagar, le mandas la transacción a tus vecinos
3. Cuando te llega una transacción, la propagas a tus vecinos si es que es válida

Charlie

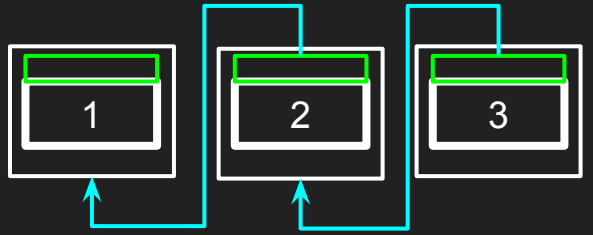


Alice



Supongamos cada 10 minutos un participante es elegido al azar para generar nuevos bloques.

Bob

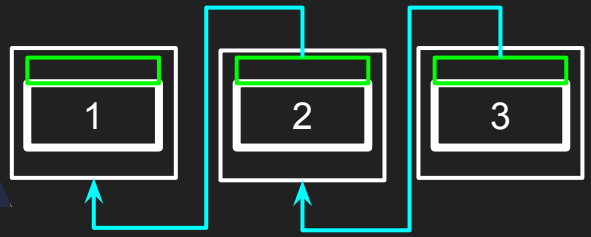


Reglas:

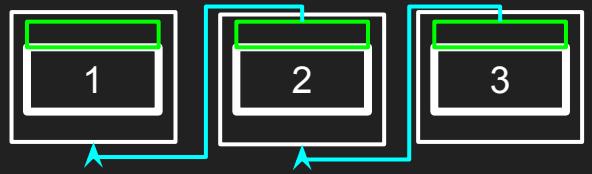
...

4. Cuando formas un bloque, lo propagas a tus vecinos
5. Si recibes un nuevo bloque, lo propagas a tus vecinos si es que es válido

Charlie

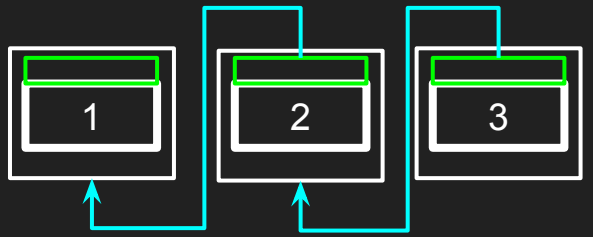


Alice



Para incentivar a que el que genera un nuevo bloque lo haga de manera correcta, el sistema lo premia con monedas nuevas (transacción *coinbase*). La cantidad está predefinida.

Bob

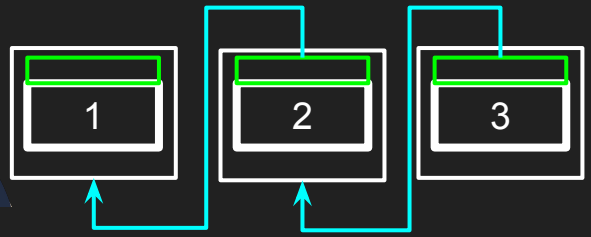


Reglas:

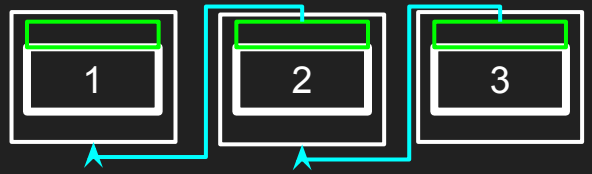
...

4. Cuando formas un bloque, lo propagas a tus vecinos
5. Si recibes un nuevo bloque, lo propagas a tus vecinos si es que es válido

Charlie

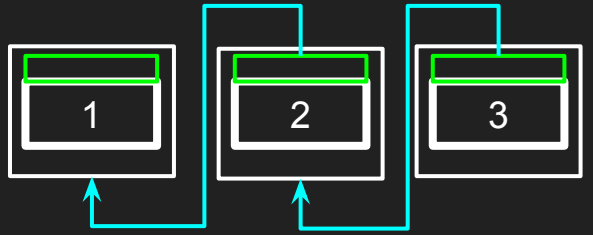


Alice



¿Qué significa que un bloque sea válido?

Bob

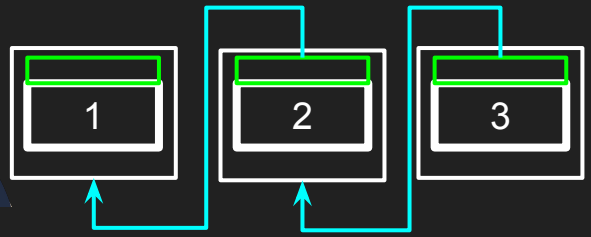


Reglas:

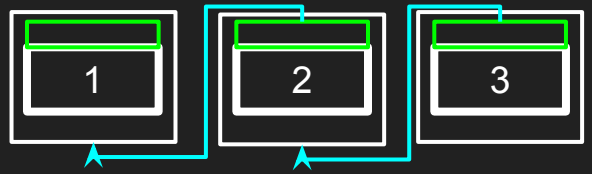
...

4. Cuando formas un bloque, lo propagas a tus vecinos
5. Si recibes un nuevo bloque, lo propagas a tus vecinos si es que es válido

Charlie



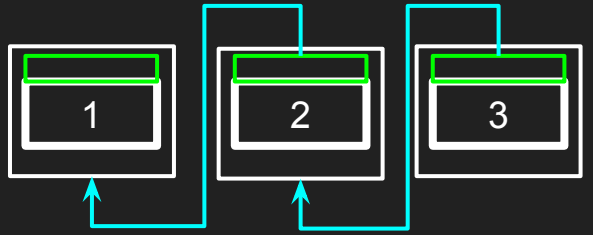
Alice



¿Qué significa que un bloque sea válido?

1. Todas las transacciones son válidas
2. Los montos son correctos
3. No hay doble gasto
4. Se verifica que efectivamente lo generó quien corresponde

Bob

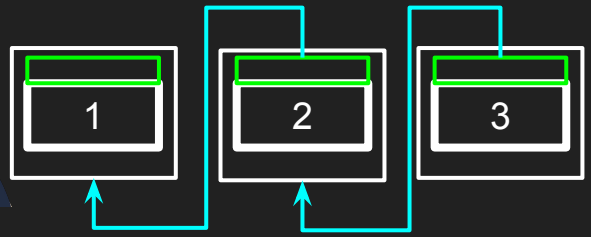


Reglas:

...

4. Cuando formas un bloque, lo propagas a tus vecinos
5. Si recibes un nuevo bloque, lo propagas a tus vecinos si es que es válido

Charlie



¿Cómo se elige quién genera un nuevo bloque?

¿Cómo se elige quién genera un nuevo bloque?

Agregamos un nuevo campo al bloque: Nonce

¿Cómo se elige quién genera un nuevo bloque?

Agregamos un nuevo campo al bloque: Nonce
Cualquiera puede agregar un nuevo bloque, si es que elige el nonce correcto

¿Qué significa que el nonce de un bloque sea correcto?

¿Qué significa que el nonce de un bloque sea correcto?

El nonce de un bloque es correcto, si el string resultante de hashear el bloque empieza con una cierta cantidad de ceros

Minería Bitcoin

El nonce de un bloque es correcto, si el string resultante de hashear el bloque empieza con una cierta cantidad de ceros

La cantidad específica de ceros se regula automáticamente cada 2016 bloques de forma que en promedio los bloques se encuentren cada 10 minutos

Minería Bitcoin

El nonce de un bloque es correcto, si el string resultante de hashear el bloque empieza con una cierta cantidad de ceros

La cantidad específica de ceros se regula automáticamente cada 2016 bloques de forma que en promedio los bloques se encuentren cada 10 minutos

¿Qué pasa si dos personas encuentran un bloque válido al mismo tiempo?

Minería Bitcoin

El nonce de un bloque es correcto, si el string resultante de hashear el bloque empieza con una cierta cantidad de ceros

La cantidad específica de ceros se regula automáticamente cada 2016 bloques de forma que en promedio los bloques se encuentren cada 10 minutos

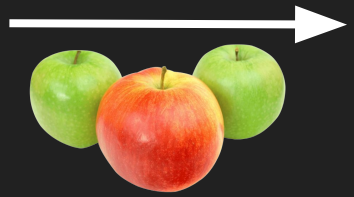
¿Qué pasa si dos personas encuentran un bloque válido al mismo tiempo?

Se aceptan y propagan ambos, hasta que un nuevo bloque sea generado. La cadena más larga es la que es considerada válida.

¿Cómo hacer trampa?

¿Cómo hacer trampa?

Alice



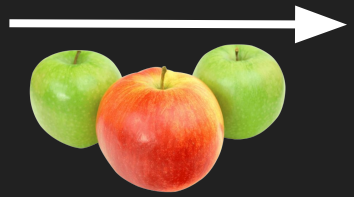
Alice, mira en el bloque 3, te pagué 100 por las manzanas

Bob

¿Cómo hacer trampa?

Alice

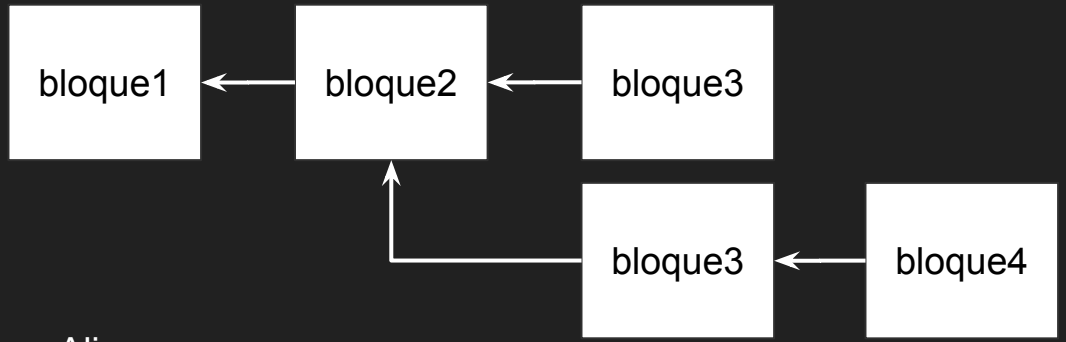
Genial, gracias!



Alice, mira en el bloque 3, te pagué 100 por las manzanas

Bob

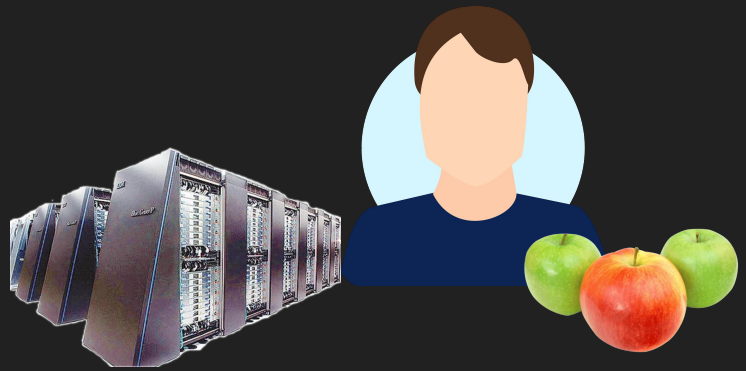
¿Cómo hacer trampa?



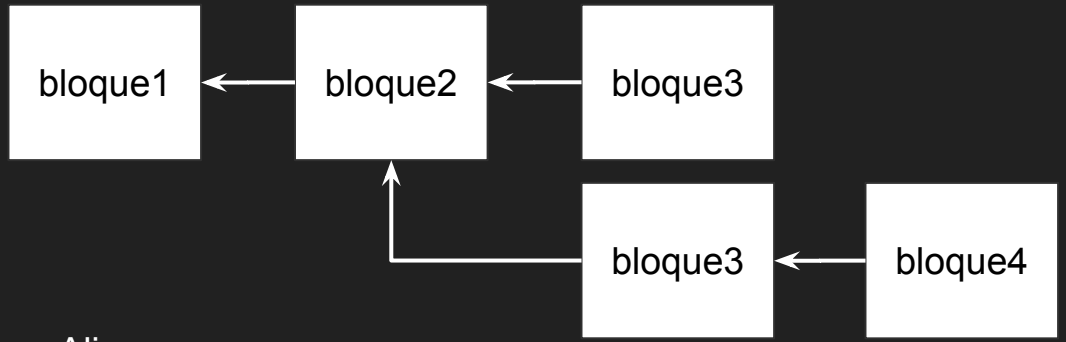
Nuevo bloque 3 sin la transacción a Alice, como la cadena más larga ahora no tiene esa transacción, es como si Bob no hubiese pagado.

Alice

Bob



¿Cómo hacer trampa?

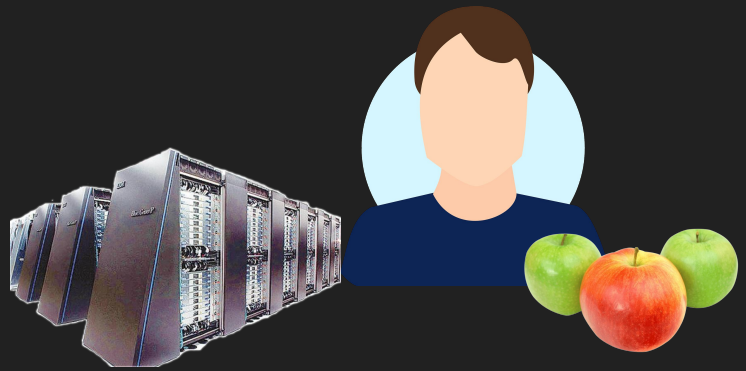


Nuevo bloque 3 sin la transacción a Alice, como la cadena más larga ahora no tiene esa transacción, es como si Bob no hubiese pagado.

Alice

De ahora en adelante esperaré a que las transacciones estén a mayor profundidad en el blockchain

Bob



Agradecimientos

Libro Grokking Bitcoin de Kalle Rosenbaum para la explicación e imágenes de funciones de hash criptográficas y firmas criptográficas.

Slides de Martín Ugarte en las que me basé para armar esta presentación.

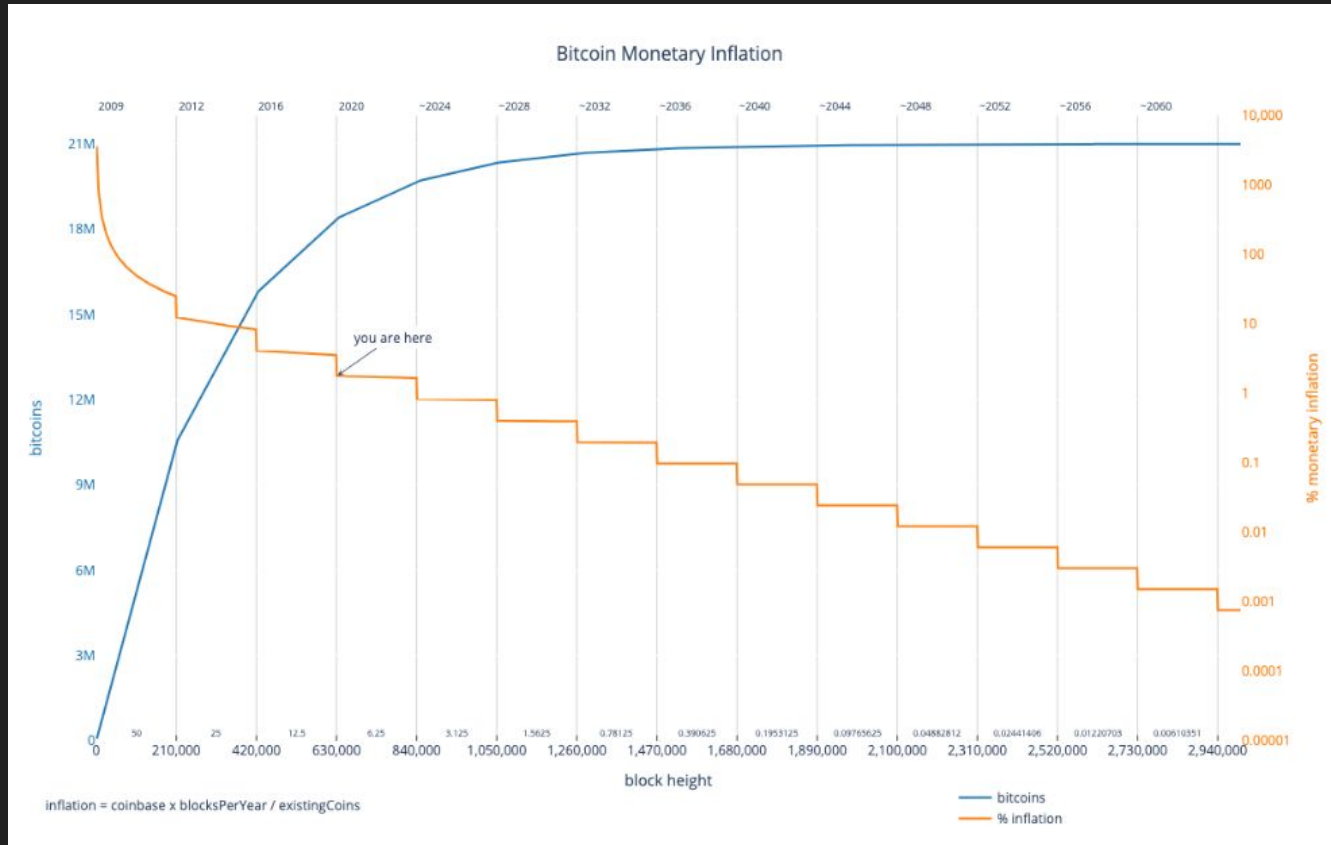
¿Preguntas?

¿Todavía aquí?

¿Por qué son solo 21 millones?

La cantidad máxima de nuevos bitcoins que se generan por bloque (en la transacción coinbase) se reduce a la mitad cada 210.000 bloques.

Actualmente solo se generan a lo más 6.25 nuevos bitcoins en cada bloque.



¿Por qué Bitcoin?