

Analisi di Bitcoin

Anonimato, Sicurezza e Sviluppi Futuri

Paoluzzi Matteo

Università degli Studi di Udine

Relatore:
Dott. Ivan Scagnetto

IV Sessione di Laurea AA 2012/2013, Aprile 2014

Di cosa si tratta

Una valuta elettronica basata su crittografia a chiave pubblica progettata per:

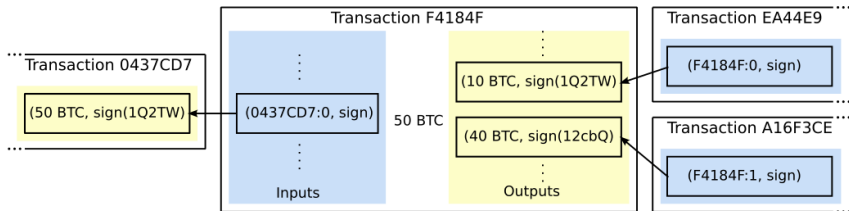
- Proteggere l'identità degli utenti sfruttando **indirizzi** anonimi.
- Essere indipendente da qualsiasi istituto di credito.
- Essere immune dal rischio di inflazione.
- Funzionare su base P2P in modo pubblico, sicuro e verificabile.

L'anonimato dell'utente viene implementato tramite stringhe di testo note come indirizzi.

- Un indirizzo viene generato a partire da una coppia di chiavi pubbliche e private.
- Tutte le **transazioni** di BTC avvengono da e verso indirizzi, senza divulgare informazioni sull'identità delle parti coinvolte.
- Le chiavi di cifratura consentono di spendere il denaro ricevuto e impedire che il denaro inviato finisca ad un destinatario diverso da quello desiderato.
- Ogni utente è incoraggiato ad avere molteplici indirizzi, mantenendo sicure le porzioni private delle chiavi appositamente generate.

Transazioni

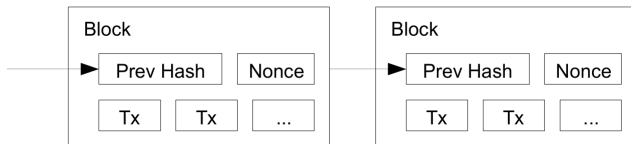
- Sono identificate da un hash calcolato su un sottoinsieme di dati in modo da fissare alcune proprietà specifiche.
- Gli output contengono le BTC inviate e lo script che il destinatario deve eseguire per spendere le BTC.
- Con gli input si firmano gli output di transazioni precedentemente ricevute dimostrando di esserne il proprietario.
- Se l'input è maggiore dell'output, il resto diventa una **transaction fee** donata a chi trova il **blocco** contenente la transazione.



Le transazioni vengono raccolte in blocchi che risultano validi se il loro hash non supera un determinato target:

- Trovare tale hash è un'operazione detta **mining** configurato ridimensionando il target ogni 2016 blocchi in modo da generare in media un blocco ogni 10 minuti.
- Il nodo che trova il blocco viene premiato con una quantità fissa di BTC e con le transaction fee.

Nell'insieme i blocchi formano una **blockchain** il cui scopo è fissare le transazioni nel tempo rendendone computazionalmente quasi impossibile la modifica.



How a Bitcoin transaction works

Bob, an online merchant, decides to begin accepting bitcoins as payment. Alice, a buyer, has bitcoins and wants to purchase merchandise from Bob.

WALLETS AND ADDRESSES



Bob and Alice both have Bitcoin "wallets" on their computers.



Wallets are files that provide access to multiple Bitcoin addresses.



An address is a string of letters and numbers, such as 1A1zP1eP5QGTHi3kxL9nF4z3R7J21



Each address has its own balance of bitcoins.



Bob creates a new Bitcoin address for Alice to send her payment to.

CREATING A NEW ADDRESS



Public Key Cryptography 101

When Bob creates a new address, what he's really doing is generating a "cryptographic key pair" composed of a private key and a public key. You sign a message with a private key (which only you know). It can be verified by using the matching public key (which is known to anyone). Bob's new Bitcoin address represents a unique public key, and the corresponding private key is stored in his wallet. The public key allows anyone to verify that a message signed with the private key is valid.

It's tempting to think of addresses as bank accounts, but they work a lot differently. Bitcoin users can create as many addresses as they wish and in fact are encouraged to create a new one for every new transaction to increase privacy. So long as no one knows which addresses are Alice's, her anonymity is protected.

Gary, Gertrude, and Glenn are Bitcoin miners.



The miners' computers are set up to calculate cryptographic hash functions.

VERIFYING THE TRANSACTION

The computers bundle the transactions of the past 10 minutes into a new "transaction block."



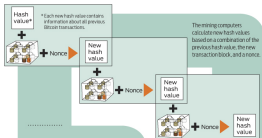
SUBMITTING A PAYMENT

Alice tells her Bitcoin client that she'd like to transfer the purchase amount to Bob's address.



Alice's wallet holds the private key for each of her addresses. The Bitcoin client signs her transaction request with the private key of the address she's transferring bitcoins from.

Anyone on the network can now use the public key to verify that the transaction request is actually coming from the legitimate account owner.



Cryptographic Hashes

Cryptographic hash functions transform a collection of data into an alphanumeric string with a fixed length, called a hash value. Even tiny changes in the original data drastically change the resulting hash value. And it's essentially impossible to predict which initial data set will create a specific hash value.

- Thousand of all odd: 256-bit (32 byte, 64 more characters)
- Thousand of all odd: 486C (64 byte, 1000)
- Thousand of all odd: 1000 720 8102...

The word of all odd

Creating hashes is computationally trivial, but the Bitcoin system requires that the new hash value have a particular form—specifically, it must start with a certain number of zeros.

0000 0000 0000

The miners have no way to predict which nonce will produce a hash value with the required number of leading zeros. So they're forced to generate many hashes with different nonces until they happen upon one that works.



Nonces

To create different hash values from the same data, Bitcoin uses "nonces." A nonce is just a random number that's added to data prior to hashing. Changing the nonce results in a widely different hash value.

Each block includes a "coinbase" transaction that pays out 50 bitcoins to the winning miner—in this case, Gary. A new address is created in Gary's wallet with a balance of newly mined bitcoins.



TRANSACTION VERIFIED

As time goes on, Alice's transfer to Bob gets buried beneath other, more recent transactions. For anyone to modify the details, he would have to redo the work that Gary did—because any changes require a completely different winning nonce—and then redo the work of all the subsequent miners. Such a feat is nearly impossible.



Double-Spending Inoltrare più transazioni con lo stesso input in modo che solo quella più conveniente per il mittente venga inclusa in un blocco senza che il destinatario se ne accorga. Senza contro-misure e con adeguata preparazione, l'attacco ha una probabilità di successo del 90%.

History Revision Creare una blockchain alternativa che modifica l'intera storia delle transazioni da un dato blocco in poi. Una blockchain alternativa può verificarsi spontaneamente con una probabilità del 1.78% ma si risolve automaticamente.

Portafogli Contengono le chiavi private necessarie per spendere le proprie BTC e sono quindi un obiettivo sensibile da tenere il più possibile sicuro e lontano da pericoli.

Double-Spending Collegandosi ad un gran numero di nodi fidati, bloccando le comunicazioni in ingresso e attendendo numerose conferme di ogni transazione è possibile di fatto annullare le probabilità di successo di un attacco doppia-spesa.

History Revision È possibile dimezzare la frequenza di blockchain alternative spontanee velocizzando la propagazione dei messaggi in rete. Le blockchain arbitrarie non possono essere bloccate, ma è estremamente improbabile (per ora) che un tale attacco possa avere luogo.

Portafogli Oltre le classiche opzioni di cifratura, è possibile salvare il portafogli in computer mai collegati in rete (**cold storage wallet**) oppure usare indirizzi "receive main - send once" stampati unicamente su carta (**paper wallet**).

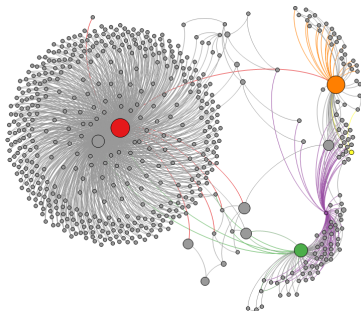
Il sistema di indirizzi adottato è simile a quello dei conti bancari in Svizzera: numeri di conto non collegabili direttamente a persone.

Ma il sistema di cifratura a chiave pubblica permette di stabilire se indirizzi diversi hanno il medesimo proprietario:

- Una transazione contenente più indirizzi in input indica che tali indirizzi appartengono tutti al creatore della transazione stessa.
- Se una transazione contiene un output molto piccolo e destinato ad un indirizzo mai visto prima, probabilmente tale indirizzo è stato creato appositamente dal creatore della transazione per raccogliere il resto.

Creando un grafo delle transazioni pubblicamente disponibile e applicando le aggregazioni descritte, è possibile creare un secondo grafo approssimato rappresentante il flusso di monete tra utenti invece che tra indirizzi. Unendo questi due grafi ad altre fonti esterne potrebbe essere possibile identificare effettivamente un utente e tracciarne le attività.

Esempio reale: un furto



: *

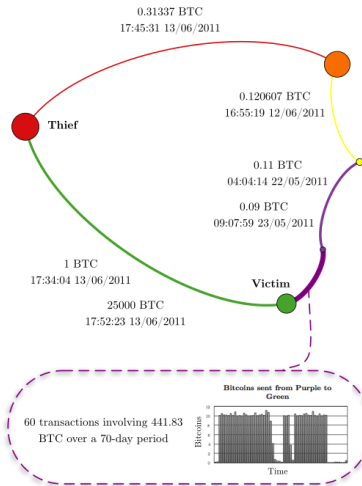
Vittima

Presunto ladro

Slush Pool

LulzSec

Sconosciuto



Furto: transazioni seguenti

