Bits on Blocks

Thoughts on blockchain technology

Menu

A gentle introduction to immutability of blockchains

FEBRUARY 29, 2016 ~ ANTONYLEWIS2015

In the context of data security, the immutability of data stored on blockchains is important. What do people mean when they say "Blockchains are immutable"? In this post I try to explain the key concepts.

It may be useful to read introductions to blockchains and Bitcoin if you have just arrived here or are unfamiliar with them.

WHAT IS IMMUTABILITY?

Immutable means that something is unchanging over time or unable to be changed.

So in our context, it means **once data has been written to a blockchain no one, not even a system administrator, can change it**. This provides benefits for audit. As a provider of data you can prove that your data hasn't been altered, and as a recipient of data you can be sure that the data hasn't been altered. These benefits are useful for databases of financial transactions.

Immutability is relative. For example if I send an email to a large list of friends, that data is pretty immutable from my perspective. To change it, I'd have to persuade my friends each to delete the email (or persuade Gmail and the people running all the mailservers of my friends). From my perspective, and with the control I have, that email is immutable – I can't unsend or revoke it without collaboration and risk of detection.

So immutability is relative, and relates to how hard something is to change.

PRIVATE DATABASES

With a private database, an *end-user* may have read-only access. She will not be able to change the contents of a row in that database. However, someone with *higher privileged access* like a systems administrator may be able to change the data. So how do we currently manage the risk of a naughty systems administrator changing data to his advantage? In existing systems and organisations, we try to create segregations of responsibility, so that no single person can do something bad undetected.

For example an administrator may have access to change the database, but the logs may be stored on another system which is owned and managed by someone else. These organisational systems are put in place to deter that individual from making the changes. We need to trust that the organisational system works. However there is no control mechanism making the data immutable in the first place.

Enter blockchains.

BLOCKCHAINS

Blockchains are essentially databases with some inbuilt pre-agreed technical and business logic criteria, kept in sync via peer-to-peer mechanisms and pre-agreed rules about what new data can be added. With respect to immutability, there are two key ideas that help to make tampering easy to detect: hashes and blocks.

Hashes

A *hash function* is a type of mathematical function which turns *data* into a fingerprint of that data called a *hash*. It's like a formula or algorithm which takes the input data (any data, whether it's the entire Encyclopedia Britannica, or just the number '1') and turns it into an output of a fixed length, which represents the fingerprint of the data. There are many types of hash functions, and a common robust one is called SHA-256 (which stands for Secure Hash Algorithm – 256 bit)

When you mash the phrase "Hello from Bits on Blocks!" through this mathematical function, you get this fingerprint out: 389f9ef3822e5c88f4b140db82c459064711a52182a3e438b4ebc7ecda62b9bb. The fingerprint (389f…b9bb) is called the SHA-256 hash of the input phrase.

Two relevant properties of a good hash function are:

1. It's hard to back-calculate the original data from the hash
2. If the input data changes in the slightest, the hash changes in an unpredictable way

Hashes are the basis of the security and immutability of blockchains. You can play with them online here.

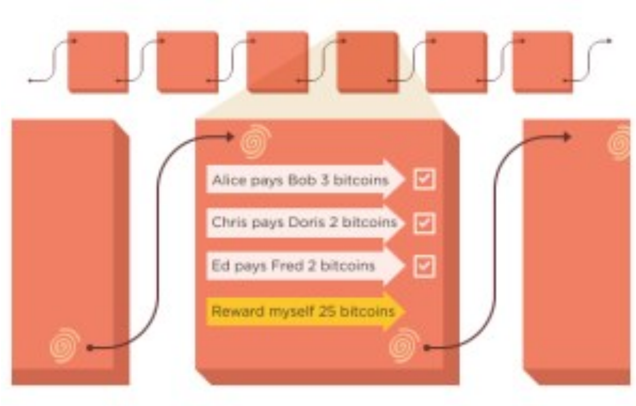
Blocks

An important idea in Bitcoin's blockchain is that transactions are bundled into blocks before being added to the blockchain database. Blocks contain some bitcoin transactions (payments) and also some other data including the previous block's hash. As each block includes the previous block's hash as part of its data, a **chain of blocks** is formed.

Creating a ledger of transactions with blocks that refer to previous blocks is a much better idea than numbering pages in a book. In a book ledger with numbered pages, 1, 2, 3, etc it would be easy to tear out page 40 and replace it with another page 40 with slightly different transactions. The book's integrity remains intact, with pages 39, 40, 41 becoming 39, 40, 41 – no change. Also there is nothing in the page number '40' that reflects any of the content in that page and the ordering of the pages is implicit from the page numbers.

However in a blockchain, instead of referring to block *numbers*, blocks are referenced by their *hash* and each block explicitly specifies which block (hash) it is building on. So it looks more like:

- block with hash 66a045b45 (building on block with hash a2c064616), followed by
- block with hash 8939a3c35 (building on block with hash 66a045b45), followed by
- block with hash a41f02e92 (building on block with hash 8939a3c35)


Alice pays Bob 3 bitcoins ☑
Chris pays Doris 2 bitcoins ☑
Ed pays Fred 2 bitcoins ☑
Reward myself 25 bitcoins

So, blocks are explicitly ordered by reference to previous block hashes, which reflect content, instead of being ordered implicitly by a numbering system (1, 2, 3) which is content-agnostic. See inside bitcoin's blockchain for more detail.

Key points

1. Each block's hash is derived from the contents of the block
2. Each block refers to the previous block's hash, not a sequential number
3. Data in a blockchain is internally consistent, that is you can run some checks on it, and if the data and hashes don't match up, there has definitely been some tinkering.

ISOLATED BLOCKCHAIN DATA

Let's first see what happens if you took Bitcoin's Blockchain and copied it onto a USB stick (it's about 55GB now so can still fit easily). What could you do with the data on the stick before passing it to someone else, like a regulator? Could you change the data and fool them?

Bitcoin's blockchain has nearly 400,000 blocks. Let's say you remove a transaction from block 200,000 which is about half way through the blockchain, trying to pretend that a specific payment never happened. What would happen?

1. The block's hash fails

The first thing the regulator could do when receiving the USB stick is **re-calculate all of the block hashes based on the block data**, and check that the block hashes supplied are valid and consistent with the contents of each block. If there is a discrepancy, then that means the transactions in the block don't match the block's hash, and block has been tampered. So to fool the regulator you would need to recalculate the block's hash to make it consistent with the altered contents.

2. The chain fails

However this breaks the chain. Remember that each block contains the previous block's hash. If block 200,000's hash changes, then 200,001 will be referencing a block hash which no longer exists. Block 200,001 will reference block 200,000's *old* hash instead of its *new* one. So the blockchain chain is broken, which is an obvious failure. To make this work, you will need to **rebuild and rehash each block** following the tampered block**,** replacing the contents of the previous-block-hash pointers.

However, there are safeguards to make it very hard or impossible to rebuild a blockchain. These safeguards differ based on the block-adding mechanisms and rules of different blockchains, and there are two dominant schemes: **target hashes** for proof-of-work public blockchains; and **specific signatures** for (some) private blockchains.

3. Chains are hard to rebuild

For **public proof-of-work blockchains** such as Bitcoin, there is a concept of *mining difficulty*. In Bitcoin, a block is only considered valid if the block hash follows a strict pattern – namely the hash has to be smaller than a target number, often described by "starting with a certain number of zeroes". See a gentle introduction to bitcoin mining for further detail.

So not only do you need to recalculate the block's hash, but you need to make sure that the recalculated hash is below a certain number. You need to *re-mine* the block by adjusting another part of the block's contents (called the nonce) repeatedly until you find a hash that is smaller than the target number. This takes some **significant computational power**. You then need to do this to every subsequent block. Colloquially, **you need to re-mine the entire blockchain from that block onwards.** Given the large amount of computational effort required to generate a valid hash that meets the criteria, this would be problematic, and furthermore, the earlier your block in the chain, the longer it will take you to do, as you have more blocks to re-mine.

For **private** blockchains, such as a Multichain, the block-adding mechanism tends to be a little different, and instead of relying on expensive proof-of-work, the rules can be set up where block-adders take it in turns to add blocks in a randomised round-robin fashion, and each block needs to be digitally signed by the block-adder. The blockchain is only valid and accepted if the blocks are signed by a defined set of participants. This means in order to recreate the chain, you'd need to know private keys from the other block-adders. Stealing these keys is a very different challenge to proof-of-work hashing.

Summary

If you did all of that then the data on the USB stick would be internally consistent and would look like a valid blockchain. But only to someone *who can't check it against any other copy of that blockchain*.

BLOCKCHAINS WITH MULTIPLE COPIES

All of the above assumes that the data on the stick is the *only version* that the regulator sees. Let's say you manage to create an internally consistent blockchain by removing the transaction *and* recreating all the block hashes to all conform to the validation criteria.

All it takes now for the regulator is to check other copies of the blockchain – and check **one single number – the hash of a recent block**. If the hash on the last block on the USB stick is different to the hash that they can find from any other (non-colluding) participant, then the regulator can immediately spot that something fishy is going on and the data on the USB stick is different to the data on the living blockchain.

The regulator doesn't even need to *see the data* in the live blockchain. They just need to see the hash of a recent block.

In other words, it is extremely difficult to try to create a fake blockchain.

CHANGING A BLOCKCHAIN MID-FLIGHT

What about trying to change the **existing data** in a blockchain that you are participating in? How would you try to get an amended block accepted by others in the network?

This is again difficult, because of the 'longest chain' rule which is the basis of consensus for most blockchains. The 'longest chain' rule broadly says that as a participant, if you see multiple competing valid chains, believe the one with more blocks.

So if you rebroadcast an amended block 200,000 you are in effect creating a blockchain 'fork' which is much shorter than the real chain (whose length is, say, 400,000). There are now two competing blockchains, one which is 200,000 blocks long and contains your amended block, and another which is 400,000 blocks long. Existing nodes will accept your block (if it's valid) but then immediately ignore it because they already know about the existing longer chain.

The only way is to make the change and create a longer chain, requiring lots of computing power or the private keys of other block-adders (depending on who is allowed to add blocks, and how) and push an entirely new lineage of blocks out, longer than the existing one. You need a significant amount of computing power to be in with a chance of outcompeting an existing proof-of-work chain like Bitcoin.

And even if you manage to do this, although technically your new chain would be valid, **realistically the community would notice** if there was a block re-organisation more than a few blocks deep; this would get investigated.

CONCLUSION

It is extremely difficult to change the data in an 'offline' blockchain, and even harder for a live blockchain.

When people say that blockchains are immutable, they don't mean that the data can't be changed, they mean it is extremely hard to change without collusion, and if you try, it's extremely easy to detect the attempt. This property of blockchains has positive and negative implications for the security of data stored, and by extension also for data privacy.

POSTED IN BITCOIN, BLOCKCHAIN, INTRODUCTIONS, MINING

BITCOIN BLOCKS CONSENSUS CRYPTOGRAPHY DISTRIBUTED LEDGERS HASHING IMMUTABILITY IMMUTABLE LONGEST CHAIN RULE PROOF OF WORK SHA

Published by antonylewis2015

View all posts by antonylewis2015

‹ PREVIOUS NEXT ›

In a nutshell: Eris (Epicenter Bitcoin interview – Jan 2016) *In a nutshell: MultiChain (Epicenter Bitcoin interview – Nov 2015)*

6 thoughts on “A gentle introduction to immutability of blockchains”

Brian Crain

FEBRUARY 29, 2016 AT 4:58 PM

Great post, Antony! How PoS-based, private chains differ in terms of mutability from PoW-based chains is crucial to understand.

Loading...

Reply

GC

FEBRUARY 29, 2016 AT 6:22 PM

Anthony,

Downloaded multichain and had a play with it today. Quite cool. Is anyone you know testing or using this so far?

Gavin

From: Bits on blocks Reply-To: Bits on blocks Date: Monday, 29 February 2016 at 8:22 AM To: Gavin Costin Subject: [New post] A gentle introduction to immutability of blockchains

WordPress.com antonylewis2015 posted: “In the context of data security, the immutability of data stored on blockchains is relevant. What do people mean when they say “Blockchains are immutable”? In this post I try to explain the key concepts. It may be useful to read introductions to blockcha"

Loading...

Reply

antonylewis2015

FEBRUARY 29, 2016 AT 6:34 PM

Watch this space... Will have a post up soon on Multichain which I agree is quite cool!

Loading...

Reply

sn0wy13

JUNE 5, 2016 AT 10:25 AM

"This property of blockchains has positive and negative implications for the security of data stored, and by extension also for data privacy."

I would like to start a discussion about this! What are the negative implications? Are you talking about transparency and the fact that there exist multiple copies of the data? Or are we talking negative implications against a bad player?

Loading...

Reply

antonylewis2015

JUNE 8, 2016 AT 9:07 AM

Immutability isn't always desired – or good. Certain censorship is sometimes useful. See Vlad's excellent commentary: http://spectrum.ieee.org/computing/networks/ethereum-developer-explores-the-dark-side-of-bitcoininspired-technology

Loading...

Reply

sn0wy13

JUNE 8, 2016 AT 9:19 AM

Thanks Antony, that's what I thought. I have a feeling things will get messy. But the old system ain't working no more. Switching out the engines while trying to keep the plane in the air isn't easy, let's hope we don't crash before that.

Loading...

Reply

Leave a Reply

Enter your comment here...

Follow Blog via Email

Enter your email address to follow this blog and receive notifications of new posts by email.

Email Address

Follow

Join 8,635 other subscribers.

Get The Basics

THE BASICS OF
BITCOINS AND BLOCKCHAINS
An Introduction to Cryptocurrencies and the Technology that Powers Them
ANTONY LEWIS

If you like my blog, please buy my book. It will make you taller, funnier, better looking and richer. Guaranteed. Click it. Buy it. Be cleverer.

Secure your cryptocurrencies!

For your own sake, protect your cryptocurrencies with a hardware wallet like a Ledger Nano. NOW is the right time to do it!

Search

Sort by Relevance

Top Posts & Pages

- Metaverse Musings Part One
- A gentle introduction to Ethereum
- A Gentle Introduction to Blockchain Technology
- Bits on Blocks
- A gentle introduction to bitcoin mining
- Inside Bitcoin's blockchain
- The pros and cons of internal blockchains
- A gentle introduction to bitcoin
- A gentle introduction to smart contracts
- My story

Categories

- banking (11)

bitcoin (20)

blockchain (50)

central banks (8)

Corda (15)

digital tokens (23)

distributed ledgers (18)

economics (4)

Epicenter Bitcoin (3)

ethereum (15)

Events (2)

Fabric (1)

financial inclusion (1)

fintech (12)

ICO (4)

identity (5)

industry workflow tools (7)

infographics (2)

interview (2)

introductions (14)

Iroha (1)

kyc (4)

law (1)

Libra (1)

mining (4)

money (14)

nutshell (6)

payments (8)

programmable money (2)

Ricardian contracts (1)

Sawtooth Lake (1)

smart contracts (15)

stablecoins (5)

thought (8)

tokens (13)

Uncategorized (6)

Blogroll

Dave Hudson's blog

Emin Gun Sirer's blog

Epicenter podcasts

Gavin Andresen’s blog

Gideon Greenspan's blog

Organ of Corti

Richard Gendal Brown's blog

Robert Sams' blog

Rusty Russell’s blog

Tim Swanson's blog

Vitalik Buterin's blog

Creative Commons

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

POWERED BY WORDPRESS.COM.