Blockchain and Cryptocurrencies

Week 7 — Chapter 9: Bitcoin as a Platform

Prof. Dr. Peter Thiemann

Albert-Ludwigs-Universität Freiburg, Germany

SS 2020

Contents

Non monetary uses

Bitcoin as a Log

Blockchain

- tamperproof ledger
- append-only
- history preserved

Bitcoin as a Log

Blockchain

- tamperproof ledger
- append-only
- history preserved

Log File (wikipedia, excerpt)

[...] a log file is a file that records either events that occur in an operating system or other software runs, or messages between different users of a communication software.
A transaction log is a [...] data collection method that automatically captures the type, content, or time of transactions made [...] with that system.

Secure Timestamping

Goal

prove knowledge of value x at time T

Method

- choose random r with high min-entropy
- publish $H(r\|x)$ at time T on the blockchain
- (e.g., using `OP_RETURN` or coin burn)
- if challenged later, we can produce r and x and point to the record on the blockchain

MIni transaction in which you can put an address

Applications of Secure Timestamping

- Prior knowledge of ideas

Applications of Secure Timestamping

- Prior knowledge of ideas
- Proof of submission

Applications of Secure Timestamping

- Prior knowledge of ideas
- Proof of submission
- Digital signature schemes (Guy Fawkes signature scheme)

Applications of Secure Timestamping

- Prior knowledge of ideas
- Proof of submission
- Digital signature schemes (Guy Fawkes signature scheme)

Overlay Currency

- uses Bitcoin as the transport and consensus layer
- (standard) Bitcoin nodes and miners are not aware of the overlay's transactions
- special nodes verify the overlay transaction, e.g., from `OP_RETURN` instructions
- example: counterparty

Contents

Fungibility

Definition (wikipedia)

In economics, fungibility is the property of a good or a commodity whose individual units are essentially interchangeable, and each of its parts is indistinguishable from another part.

- Are bitcoins fungible?

Colored Coins

- Bitcoins with additional metadata
- Implemented using special scripts
- Examples: tickets, shares, collectibles, subscriptions

Contents

Secure Multiparty Lotteries in Bitcoin

The offline version: Coin Flip

- Alice and Bob want to bet on the outcome of a coin flip
- They agree on the amount and the method to determine the winner
- Bob throws the coin, Alice shouts “heads” or “tails” while the coin is in the air
- When the coin lands, Alice wins if she correctly predicted the coin top

Secure Multiparty Lotteries in Bitcoin

The offline version: Coin Flip

- Alice and Bob want to bet on the outcome of a coin flip
- They agree on the amount and the method to determine the winner
- Bob throws the coin, Alice shouts “heads” or “tails” while the coin is in the air
- When the coin lands, Alice wins if she correctly predicted the coin top

Properties

- The outcome is random and cannot be influenced
- The winner is immediately known to Alice and Bob

Secure Multiparty Lotteries in Bitcoin

The offline version: Coin Flip

- Alice and Bob want to bet on the outcome of a coin flip
- They agree on the amount and the method to determine the winner
- Bob throws the coin, Alice shouts “heads” or “tails” while the coin is in the air
- When the coin lands, Alice wins if she correctly predicted the coin top

Properties

- The outcome is random and cannot be influenced
- The winner is immediately known to Alice and Bob

Drawbacks

- All participants have to be physically present
- They need to trust each other

Issues of Bitcoin Lotteries

Randomness?

- Can we get un-influentiable randomness on the blockchain?
- No random instruction in script

Issues of Bitcoin Lotteries

Randomness?

- Can we get un-influentiable randomness on the blockchain?
- No random instruction in script

Mutual trust?

- Betters are not (simultaneously / physically) present
- But want to guarantee payout

Issues of Bitcoin Lotteries

Randomness?

- Can we get un-influentiable randomness on the blockchain?
- No random instruction in script

Mutual trust?

- Betters are not (simultaneously / physically) present
- But want to guarantee payout

Building block for other applications

- Sealed bid auctions without trusted auctioneer
- Randomization (e.g., to break ties)
- Transfer of the auctioned (digital) asset

Coin Flipping (attempt)

Three party coin flip

Coin Flipping (attempt)

Three party coin flip

- Desired outcome: 0, 1, or 2

Coin Flipping (attempt)

Three party coin flip

- Desired outcome: 0, 1, or 2
- A, B, and C each pick a (large) random number a, b, ane c

Coin Flipping (attempt)

Three party coin flip

- Desired outcome: 0, 1, or 2
- A, B, and C each pick a (large) random number a, b, ane c
- Everyone publishes their number at the same time

Coin Flipping (attempt)

Three party coin flip

- Desired outcome: 0, 1, or 2
- A, B, and C each pick a (large) random number a, b, ane c
- Everyone publishes their number at the same time
- Everyone calculates $(a + b + c) \mod 3$

Coin Flipping (attempt)

Three party coin flip

- Desired outcome: 0, 1, or 2
- A, B, and C each pick a (large) random number a, b, ane c
- Everyone publishes their number at the same time
- Everyone calculates $(a + b + c) \mod 3$

Simultaneity

- cannot be guaranteed on the Internet
- if A publishes last, she can affect the outcome ...

Three Party Fair Coin Flip

Round One

- A, B, and C each pick a (large) random number a, b, ane c
- Everyone publishes the hashes $H(a)$, $H(b)$, $H(c)$
- Abort the protocol if two of the hashes are equal

Three Party Fair Coin Flip

Round One

- A, B, and C each pick a (large) random number a, b, ane c
- Everyone publishes the hashes $H(a)$, $H(b)$, $H(c)$
- Abort the protocol if two of the hashes are equal

Round Two

- Every party reveals their number a, b, and c
- Everyone can check consistency with the previously published hashes
- Everyone computes $(a + b + c) \mod 3$

Three Party Fair Coin Flip

Round One

- A, B, and C each pick a (large) random number a, b, ane c
- Everyone publishes the hashes $H(a)$, $H(b)$, $H(c)$
- Abort the protocol if two of the hashes are equal

Round Two

- Every party reveals their number a, b, and c
- Everyone can check consistency with the previously published hashes
- Everyone computes $(a + b + c) \mod 3$

Remark

It is sufficient that one participant picks a number at random!

Fairness and Commitment

Drawback of the coin flip protocoll

- Once C sees values a and b in round three, she can figure out whether she won
- She could choose to never reveal c and block the protocol indefinitely!

Fairness and Commitment

Drawback of the coin flip protocoll

- Once C sees values a and b in round three, she can figure out whether she won
- She could choose to never reveal c and block the protocol indefinitely!

What's needed

- a commitment to the stake
- automatic loss of the stake if c leaves the protocol (i.e., no reaction inside a time limit)

Timed Commitment in Bitcoin

Timed commitment between Alice and Bob

- Alice puts up a **bond** that vouches for value x:
- a bitcoin transaction with an output that can be spent in two ways
  1. transaction signed by both Alice and Bob
  2. transaction that includes the value x, but only needs Alice's signature

Timed Commitment in Bitcoin

Timed commitment between Alice and Bob

- Alice puts up a **bond** that vouches for value x:
- a bitcoin transaction with an output that can be spent in two ways
  1. transaction signed by both Alice and Bob
  2. transaction that includes the value x, but only needs Alice's signature

Step 2

- Alice and Bob sign a transaction that transfers the bond to Bob
- This transaction comes with a **lock time** t in the future
- Alice intends to reveal x before t, so this transaction will never be accepted

Timed Commitment in Bitcoin

Timed commitment between Alice and Bob

- Alice puts up a **bond** that vouches for value x:
- a bitcoin transaction with an output that can be spent in two ways
  1. transaction signed by both Alice and Bob
  2. transaction that includes the value x, but only needs Alice's signature

Step 2

- Alice and Bob sign a transaction that transfers the bond to Bob
- This transaction comes with a **lock time** t in the future
- Alice intends to reveal x before t, so this transaction will never be accepted

Possible Outcomes

- Alice reveals x before time t and her bond is returned
- Alice fails to reveal x before t and her bonds falls to Bob

Bitcoin script for timed commitment

Locking script for the output

```
OP_IF
  <AlicePubKey> OP_CHECKSIGVERIFY <BobPubKey> OP_CHECKSIG
OP_ELSE
  <AlicePubKey> OP_CHECKSIGVERIFY OP_HASH <H(x)> OP_EQUAL
OP_ENDIF
```

Bitcoin script for timed commitment

Locking script for the output

```
OP_IF
  <AlicePubKey> OP_CHECKSIGVERIFY <BobPubKey> OP_CHECKSIG
OP_ELSE
  <AlicePubKey> OP_CHECKSIGVERIFY OP_HASH <H(x)> OP_EQUAL
OP_ENDIF
```

Unlocking script; bond forfeited

```
<BobSignature> <AliceSignature> 1
```

Bitcoin script for timed commitment

Locking script for the output

```
OP_IF
  <AlicePubKey> OP_CHECKSIGVERIFY <BobPubKey> OP_CHECKSIG
OP_ELSE
  <AlicePubKey> OP_CHECKSIGVERIFY OP_HASH <H(x)> OP_EQUAL
OP_ENDIF
```

Unlocking script; bond forfeited

```
<BobSignature> <AliceSignature> 1
```

Unlocking script; bond returned

```
<x> <AliceSignature> 0
```

Application to Secure Lottery

n-party lottery protocol

- **timed** hash commitments
- n parties $\rightarrow n^2 - n$ commitments required
- to be effective, players have to escrow more than they are betting

Thanks!