Blockchain code development pitfalls

Stories from TrustChain development

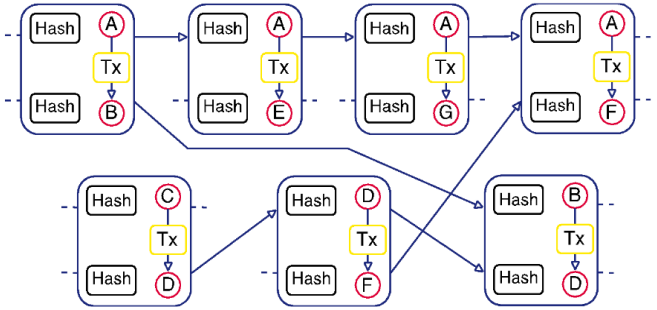
Vadim Bulavintsev
Distributed Systems
2019

TU Delft

#1 unexpected blocks diagram

Theory

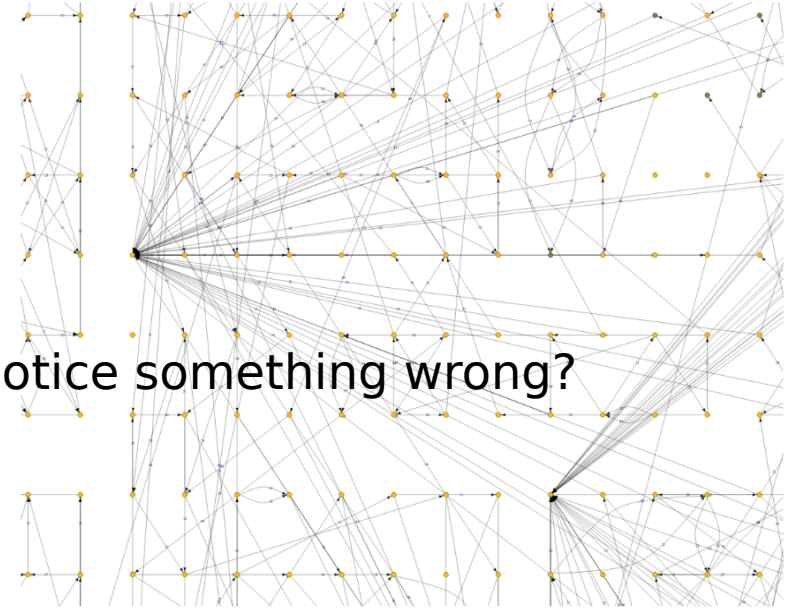
Diagram from the original TrustChain paper



Each block has exactly 2 parents and 2 children

Practice

Block diagram from an early version of TrustChain



Notice something wrong?

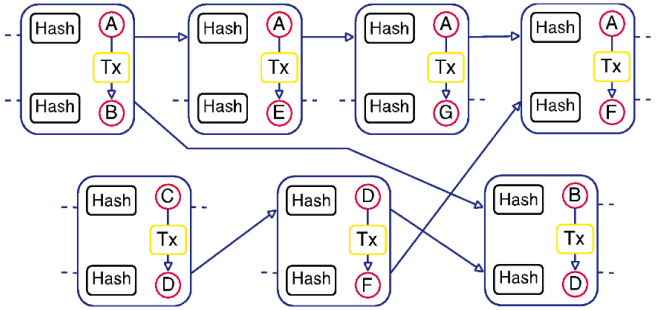
DELFT BLOCKCHAIN LAB

TU Delft

#1 unexpected blocks diagram

Theory

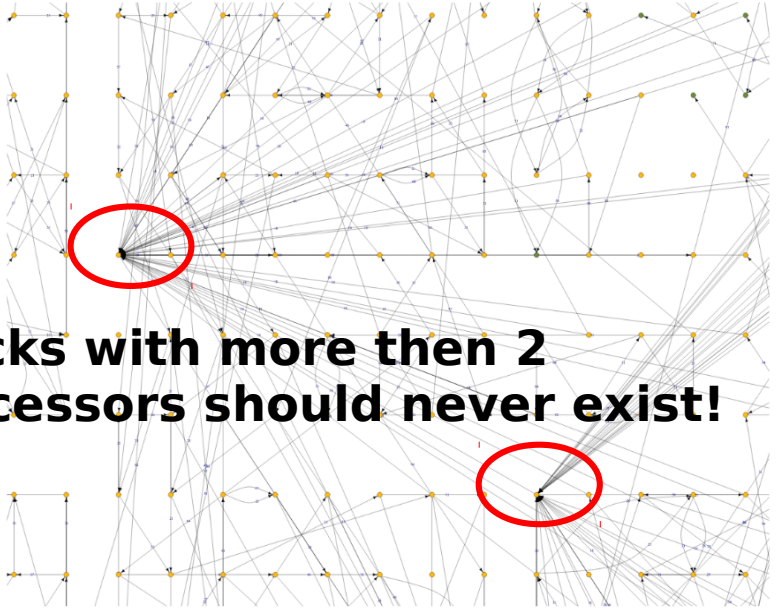
Diagram from the original TrustChain paper



Each block has exactly 2 parents and 2 children

Practice

Block diagram from an early version of TrustChain



Blocks with more then 2 successors should never exist!

DELFT BLOCKCHAIN LAB

TU Delft

#1 take-home message

- Humans think in visual images, not lines of code. Make proper use of your visual cortex.

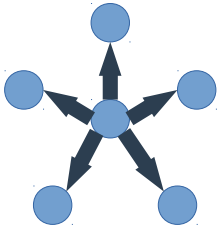
Visualize everything **from the beginning.**

DELFT BLOCKCHAIN LAB

TU Delft

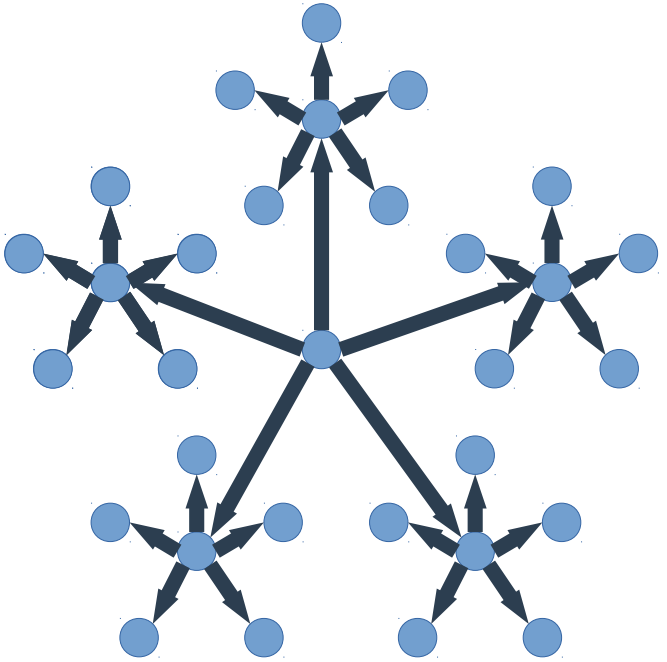
#2 excessive block broadcasts

Expected
(**and** unit tested!)



Packet time to live **1**.
Broadcast to **25** peers.

Reality



Packet time to live **2**.
Broadcast to **1110** peers.

DELFT BLOCKCHAIN LAB

TU Delft

#2 take-home message

- There were 2 errors: in the main code **and** in the test.
- The risk of a double error is higher in a complex system. Double-check everything.

Tests can have errors too.
Cross-check your tests.

DELFT BLOCKCHAIN LAB

TU Delft

#3 contagious crash

- One day, **all** app instances begin to crash 1 minute after connecting to the network.
- The crash spreads **like a disease** among peers.
- Who could have created this devious «message of death»? **Is this an attack?**

DELFT BLOCKCHAIN LAB

TU Delft

#3 take-home message

- A broken block was put on the TrustChain. Peers trying to re-send it **could not handle the value**.
- You guessed it: no attack, just a student playing around ;-)

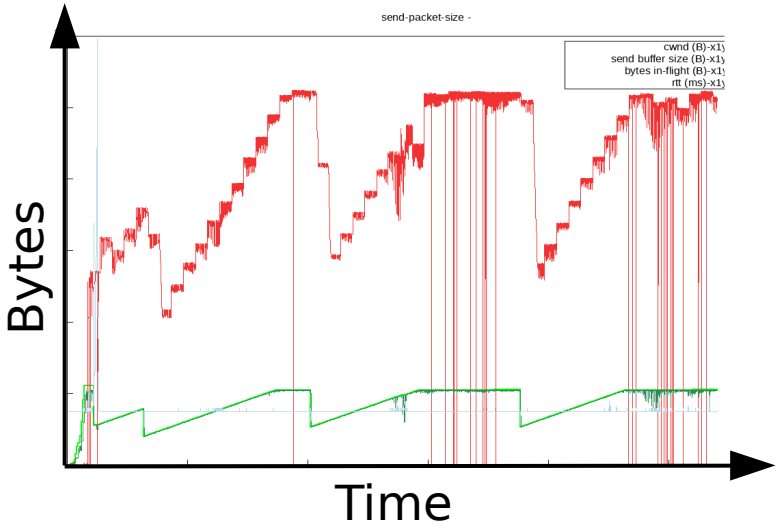
Sanitize incoming values
thoroughly.

DELFT BLOCKCHAIN LAB

TU Delft

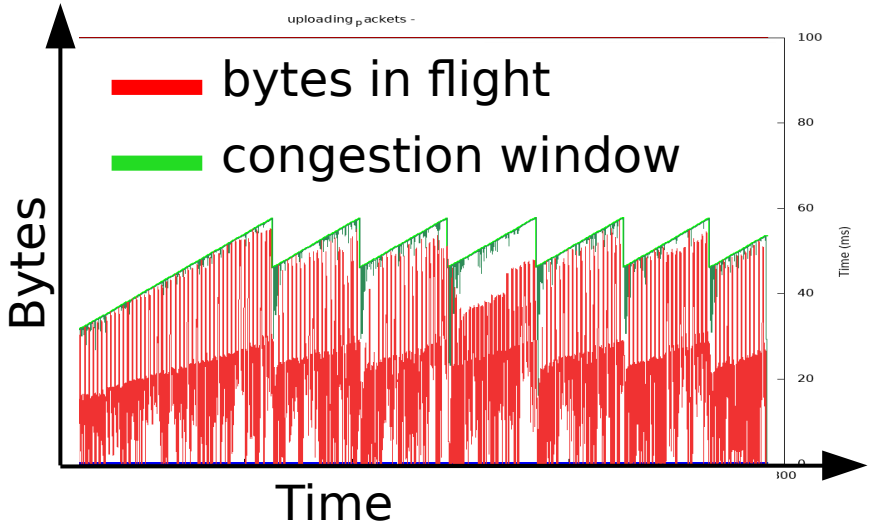
#4 confirmation bias

Wrong



Libtorrent developers were using using this chart to detect anomalies in network packet flow **for 5 years.**

Correct



Same chart, after a certain single-line bug was fixed. **The bug was there from the beginning.**

DELFT BLOCKCHAIN LAB

TU Delft

#4 take-home message

- Never take anything for granted.

Use scientific thinking:

try to **falsify** your basic assumptions.

DELFT BLOCKCHAIN LAB

TU Delft

Recap

1. Visualize everything from the beginning.
2. Cross-check your tests.
3. Thoroughly sanitize incoming data.
4. Always try to prove yourself wrong first.

DELFT BLOCKCHAIN LAB

TU Delft

Blockchain code development pitfalls

Stories from TrustChain development

Vadim Bulavintsev
Distributed Systems
2019

DELFT BLOCKCHAIN LAB

TU Delft