CSC2125: Homework #1

Due on September 24

Fan Long

Thomas Hollis

Problem 1

Section 4 argues that the longest chain rule represents the majority vote of the transaction history. This argument relies on the assumption that block propagation is fast enough, so that blocks generated by honest participants will form a chain. What would happen if this assumption is violated?

Solution

If the assumption is violated by block propagation being slow uniformly for all users in the network, then the consequence is that lots of useless work will be done. What would happen is that some users will start doing work to approve transactions based on an old outdated block. They will continue their own chain but eventually realise that a longer chain exists. This will cause them to abandon their shorter chain, and accept this longer chain. This essentially means that all their work since the outdated block was wasted as it is now irrelevant (the outdated block being the most recent block common to both chains).
If the assumption is violated differently though, i.e. if block propagation is slow only for a subset of honest users, then another effect could occur. Blocks generated by (potentially dishonest) participants with faster propagation rates could form a separate chain and become an unofficial/unwanted fork of the main Bitcoin blockchain. This is because with slow propagation, a malicious attacker could create a seemingly longer chain causing neighbouring users to attach to this malicious fork of the original chain instead of the official chain. The reason why these new users join the malicious chain is because the official chain has not reached them yet. Thus, they assume that the longest chain represents the legitimate global majority vote since it is the chain that took the most computational power to create (as far as they know). It is worth noting that this would enable a temporary attack (until the official chain reaches everyone) even without the attacker needing 50 percent of the CPU power.

Problem 2

Can you think of any other assumption the above claim relies on? Write down these assumptions and discuss them.

Solution

The above claim relies on the assumption that over 50 percent of the CPU power is from honest and well-intentioned actors (as opposed to a few dishonest actors with very large CPU power). If this is not the case a 50 percent attack could be launched whereby users are brought along an undesired malicious fork where an attacker could steal funds by double spending, for example.
Another assumption that the above claim relies on is that the hash function used for the proof-of-work is secure. Otherwise a malicious actor would be able to avoid having to use as much CPU power as the others and would be able to redo the work required for the blocks hence be able to modify the past by making a change to the transaction ledger.
The above claim also assumes that no user can force other users to disconnect. If this was possible he would become the largest CPU power and would automatically get the majority vote violating the previous statement.
Finally, the above claim assumes that the proof-of-work difficulty increases at the same rate as interest and increasing hardware speed. If not, a small number of large entities would have control of the blockchain and the system would no longer be decentralised. In fact, many modern critics of BTC argue that this is currently the case as most of the CPU hash power is concentrated in BTC mining farms which has effectively

Problem 2 continued on next page...

transformed the decentralised Bitcoin network back to centralisation.

Problem 3

What would happen if Bitcoin removes the Proof-of-Work puzzle from block generation? Imagine a hypothetical system in which all transactions form a chain directly instead of being packed into blocks first. Anyone who wants to submit a transaction can append to the end of the transaction chain without PoW. In this hypothetical system, the longest transaction chain is considered the correct transaction history. What could go wrong for the system?

Solution

Bitcoin would become fundamentally flawed if the PoW puzzle was removed. This is because it would allow any malicious user to create a chain longer than the currently longest chain which is the desirable majority-voted chain. This would then cause all the nodes to attach to this malicious users new and longer chain (believing it is the majority-voted chain). This chain in turn could contain fraudulent transactions, essentially breaking the Bitcoin protocol.
The Bitcoin protocol would have to implement some other alternative puzzle to prevent corruption of the blockchain. Indeed, many new cryptocurrencies have found alternatives to PoW such as PoS (Proof-of-Stake as implemented in Peercoin), PoR (Proof-of-Research as implemented in Gridcoin), PoC (proof-of-capacity as implemented in SpaceMint) and many more.

Problem 4

Section 5 mentions that each transaction and each block are broadcasted to everyone else in the network. How is this actually implemented in Bitcoin? You may want to search additional materials on the web to answer this question.

Solution

Bitcoin, and in fact many other cryptocurrencies (such as Gridcoin for example), broadcast their transactions without needing the list of all the other users of the network. Instead, nodes are only connected to a handful (around 10) other neighbouring nodes to which they transfer any message that they receive. Assuming the network is fully connected, if each node tells a handful of other nodes what is occurring then the information will propagate and eventually all nodes will become aware. This is referred to in Bitcoin as confirmations. Transactions are only trusted once they have received a high number of confirmations.
This is implemented by communication over TCP and usually occurs on port 8333 but this can be modified. To connect to a peer, a version message is sent by person A and another version message will be received by person B (peer) if that peer is accepting connections from the version of person A. In response, a verack message is sent by person A to person B to confirm person A is accepting connections from the version of person B. Then the addresses of both people are exchanged using the messages getaddr and addr. Every 24 hours, each node broadcasts an addr message with their IP and nodes relay this to their peers such that everyone end up with a reasonably clear picture of connected IPs in the network. Finally, the transaction information is relayed to neighbouring nodes using the inv (broadcast by a node to its peers for every transaction) and getdata (peers request to receive full transaction) messages. Nodes are checked using a

Problem 4 continued on next page...

ping message to verify that they are still active every 30min by their peers and if no message is received after 90min that node is assumed to be disconnected.
In fact, Bitcoin allows certain nodes to disconnect from the network for a while and miss the publication of a blocks before reconnecting. It does so through the mechanism detailed earlier whereby a node can catch-up by simply downloading the missing blocks that follow the longest chain (i.e. the trusted one).

Problem 5

Based on the discussion in Section 11, how many blocks does the recipient of a transaction need to wait if 1) he or she assumes that the attacker controls up to 20% of computation powers and 2) he or she can only tolerate 0.1% of double- spending risk? How many blocks to wait, if the attacker controls 30% of power and he or she can only tolerate 0.01% of double-spending risk?

Solution

The number of blocks needed for the recipient of a transaction to wait for are:

- If attacker controls 20% of computational power, tolerable double spending risk of 0.1%: 11 blocks since q = 0.2 and P $<$ 0.001 (since 10 blocks give a P = 0.001066953 while 11 blocks give a P = 0.000560139, and indeed 12 blocks give a P = 0.000294330 as expected)

- If attacker controls 30% of computational power, tolerable double spending risk of 0.01%: 32 blocks since q = 0.3 and P $<$ 0.0001 (since 31 blocks give a P = 0.0001152 and 32 blocks give P = 0.000087262)

The above calculations are based on the original Bitcoin white paper and code written to calculate the answer is available at: `https://github.com/PsiPhiTheta/Blockchain-Labs`