A blockchain is a public immutable distributed ledger and has no central authority, eliminating a single point of failure. New transactions are recorded in a block that will be appended by miners to the blockchain. To maintain consistency of transactions, each block stores the hash of the previous block header, forming a hash chain. Since blocks are referred to by their hash values, tampering with their contents will cause discrepancies in the chain and is easily detected.

Transactions are recorded as a Merkle tree, in which the value of a node is the concatenation of the hash values of the children, and the Merkle root is stored in the block header. Merkle trees make it easier to check the integrity of the chain and is easily detected.

### PROBLEMS

Because of a lack of central authority, a blockchain system needs a consensus algorithm that allows mutually distrusting nodes to agree on block creation (transaction history). While different consensus algorithms have different block creation processes, they all must tackle the following problems:

- **Double-spending**: Double-spending means spending money more than once. The consensus algorithm has to be able to detect and reject blocks that contain double-spending. In the figure below, since Block 81A has an invalid transaction, no block should be built off Branch A.
- **Forking**: Because of propagation delays, some nodes might not have received the latest blocks and build a block off an older block, causing a fork as shown in the figure below (Branches B and C). The consensus algorithm must be able to decide which branch to extend on.

### EXPERIMENT

We implemented PoW in Scorex platform and conducted an experiment to investigate the impact of the change to the target on block creation by deploying 10 nodes on Docker containers with the difficulty adjustment interval set to 50. With an easy initial difficulty, the algorithm successfully achieved a 1 block per 10 minutes rate by reducing the target, which in turn increased the difficulty.

### CONCLUSION

As we have seen, each consensus algorithm has its own advantages and disadvantages. However, while Proof-of-Work and Proof-of-Stake have been around and proven to work, Proof-of-Activity hasn’t been used in practice. Therefore, we believe future work on consensus algorithms should focus on implementation and experimentation of Proof-of-Activity.

### REFERENCES
