ECE 443/518 – Computer Cyber Security
Lecture 16 Cryptocurrency

Professor Jia Wang
Department of Electrical and Computer Engineering
Illinois Institute of Technology

October 24, 2022

Outline

Reading Assignment

- This lecture: Cryptocurrency
- Next lecture: Smart Contract, Oblivious Transfer

Outline

Proof of Work

Proof of Stake

BFT Meets Cryptocurrency

- Consensus on what the next block should be is a must for cryptocurrency.
  - Otherwise adversaries can create branches for double spending.
- Many BFT protocols are too weak to be useful here.
  - E.g. both the ones without or with digital signatures need to know how many traitors are there, but in our case one adversary can create arbitrary number of "traitor" accounts.
- The BFT propocol need to weight the participants differently.
  - So adversaries cannot simply overwhelm the protocol by creating more accounts, a.k.a. Sybil attack.

Proof of Work (PoW)

- Each account willing to generate the next block needs to perform certain amount of work before it could join the BFT protocol.
  - Typical work: for a block of hash s, find x so that $h(s||x)$ is smaller than a threshold.
  - If h is preimage resistant, one can only find such x via brute-force.
  - Block time: the expected time for some account to find a solution x.
  - When more computational power are available, the threshold is reduced such that the block time remains unchanged.
- PoW consensus: the branch with the most of work is the valid one.
  - The consensus can be reached as long as honest account owners can provide majority of work.

Obtaining the Blockchain

- Consider an account that want to generate the next block.
  - By using a program together with its private key.
- The program connects to Internet to query the blockchain.
  - However, there could be adversaries so the program must decide if the chain it receives is valid or not.
- The genesis block: the first block of the blockchain.
  - The genesis block is assumed to be well-known, usually coded into the program directly.
  - The genesis block could contain data like cryptocurrency parameters and initial balances for certain accounts.
- The program need to validate past transactions.
  - It is necessary to accumulate balances for all accounts to decide if transactions are valid – this is possible now since our computers are actually quite powerful.
- However, recall that valid transactions along cannot prevent branches (and thus double spending).

PoW Fork Choice

- Fork: the program may receive multiple valid blockchains all with the same correct genesis block.
  - They diverge somewhere back in the history.
  - Resulting from a temporary network partitioning or an attack.
- Choice: with PoW, the program should pick the blockchain with the most of the work as the correct one.
  - The work is measured as the total effort to solve the problems for all the blocks along the chain.
- 51% attack: attackers controlling more than half of the computational power could collude to cause a successful fork.
  - Suppose currently the honest accounts are at the chain $A \rightarrow B \rightarrow \cdots \rightarrow C$ from an earlier block A.
  - The attackers make a fork $A \rightarrow B'$ and continue.
  - No matter how many blocks are there between A and C, the attackers can eventually reach D' as $A \rightarrow B' \rightarrow \cdots \rightarrow D'$, that contains more work than the chain created by the honest accounts now as $A \rightarrow B \rightarrow \cdots \rightarrow C \rightarrow \cdots \rightarrow D$

PoW Finality

- With 51% attack, powerful attackers can revert transactions by creating successful forks.
  - It may take some time but 100% the attack will be successful.
- Can attackers with less computational power revert a block?
  - Finality: we need to define when a block is considered "final" and thus is not supposed to be changed or reverted.
  - Fake check scams are classical examples of attacks on finality for our banking system.
- Consider an attacker controlling 25% of computational power
  - Suppose the current chain is $A \to B$ and honest accounts are working on the block C.
  - If B is considered final immediately, the attacker will attempt to make a fork $A \to B' \to C'$ when A was ready.
  - If C' can be generated ahead of C in time, honest accounts may simply follow the chain $A \to B' \to C'$.
  - With 25% of computational power, this may happen with a probability of $(\frac{25\%}{75\%})^2 = \frac{1}{9}$.

Economic Incentives

- How could PoW cryptocurrencies actually survive when finality is always probablistic, and when powerful adversaries could have the majority of computational power?
- Economic incentives to attract honest accounts to participate in the BFT protocol.
  - Transaction fees: the account creates the next block will take all the transaction fees.
    - When there is more transactions than what the next block can hold, payers compete by paying more transaction fees.
  - Mining: the account creates the next block is allowed to award itself a predefined amount of money.
    - As a transaction with no payer.
- As a consequence, powerful adversaries have economic incentives to not cheat.
  - It is more rewarding to participate honestly than to make the cryptocurrency useless by attacking it.

Outline

Proof of Work

Proof of Stake

Proof of Stake (PoS)

- PoW consensus achieves a great success and enables a lot of honest account owners to participate.
  - For a fixed block time, need to increase complexity of work.
  - So more energy is needed to generate one block.
  - hardware depreciation + energy cost + profit = mining income
- Proof of stake: accounts stake a certain amount of the cryptocurrency itself to participate in the consensus process.
  - Without the need of computing complex works (and thus consume less energy) to resist Sybil attacks.
  - Honest accounts are rewarded with transaction fees.
  - Attackers may have their staked cryptocurrency burned.

GASPER

- PoS consensus mechanism for Ethereum, consisting of
  - Finality: Casper the Friendly Finality Gadget (Casper-FFG)
  - Fork choice: the LMD-GHOST algorithm
- Validators
  - An account who want to participate in PoS consensus need to deposit 32 ETH first to become a validator.
  - Each validator will need to vote for the next block.
- Unlike PoW where block time is an expectation, time in PoS is devided into 12-second slots and 32-slot epochs.
  - For each slot, a randomly chosen validator will create a block, and a randomly chosen committee of validators will verify the block.

Casper the Friendly Finality Gadget (Casper-FFG)

- Checkpoint blocks are blocks created at the first slot of each epoch (every 12*32=384 seconds).
- A checkpiont block is marked as "justified" when two-thirds of the total staked ETH vote so.
- When another block is marked as justified after a previous justified block, the previous block is marked "finalized".
  - So it takes two-thirds of the total staked ETH in 32 blocks to vote to finalize a block.
- Attackers controlling two-thirds of staked ETH can manupulate the block, e.g. to include or exclude transactions.
- Attackers controlling one-thirds of staked ETH can double-vote two different justified blocks to cause a fork.
  - But double-voting can be detected and punished.
- Attackers controlling more than one-thirds of staked ETH can simply remain silent to prevent any progress.
  - Punish inactivities when there is no progress for more than four epochs until a two-thirds majority can be reached.

Fork Choice with LMD-GHOST

- Latest Message-Driven Greedy Heaviest Observed Sub-Tree
- When fork happens, the situation can be much more complicated than where a block is forked into two – there could be multiple levels of fork forming a tree of many levels of branches from a root block.
- The algorithm picks a path from the root block by always choosing the justified block with more votes whenever there is a branch.
  - If a validator justifies multiple blocks in the tree, only the last one counts.
  - Ties are broken deterministically, e.g. by comparing the hash of the block.

Summary

- Both proof of work (PoW) and proof of stake (PoS) work as the consensus mechanism for cryptocurrencies.