ECE 595: Foundations of Blockchain Systems **November 16, 2020**

Lecture 13: Beyond Proof-of-Work

Lecturer: Sreeram Kannan *Scribe: Yonghun Lee*

Outline: This lecture answers three following questions:

1. What is Proof-of-Stake system?
2. How do PoS systems control the block production rate?
3. What is Nothing-at-Stake attack?

13.1 Why Proof-of-Stake?

13.1.1 Bitcoin scorecard

Bitcoin has a strong point on security, but as far as other attributes are concerned such as energy efficiency, latency, throughput as shown in Fig. 13.4. Bitcoin is not so efficient so the energy efficiency is a major dimension. In other words, the energy burning issue of Proof-of-work (PoW) is considered wasteful. We've discussed Prism which redeems the quality of Bitcoins in terms of latency. Prism also improves throughput by creating transaction blocks. In this lecture, we will take a deep dive onto a new design of systems which tries to alleviate the energy efficiency weakness.

	Security	Energy Efficiency	Latency	Throughput
Bitcoin	✓ 50% adversary	✗ ~ Switzerland	✗ 3 hours	✗ 10 Tx/Sec

Figure 13.1: Bitcoin scorecard

13.1.2 Energy waste of PoW systems

There are some misconceptions that people have about energy waste in blockchain system. The reason that energy-waste happens is not because the puzzle is some hard puzzle to solve, it's because many people are competing to solve the puzzle and every one is willing to spend as much power as the expected reward. Therefore, as more people compete to generate the next block, energy is been wasted.

13.1.3 Blockchain System Comparison: PoW vs. PoS vs. PoSpace

Proof-of-Work (PoW) systems are used for sybill resistance but have energy-wasted issues. Thus, the question is could we somehow solve the sybill. Here is alternator, Proof-of-Stake (PoS). For PoW systems, we can roughly think one approval of work translates to one unit of CPU, which implies the more CPU power you have, the more blocks you will be able to make. On the other hand, for PoS systems, if you hold one coin on $1 then you should get on vote. That is why it is called Proof-of-Stake. Therefore, you get participation mediated by how many coins you have in the system. The third system is Proof-of-Space (PoSpace) which allows 1 vote to corresponds to each one RAM unit. These are all alternative mechanisms to weigh participation. We can also imagine the most democratic version of systems, Proof-of-Personhood (PoPersonhood), in which one person trades to one vote like global democracy.

13.2 Proof-of-Stake systems

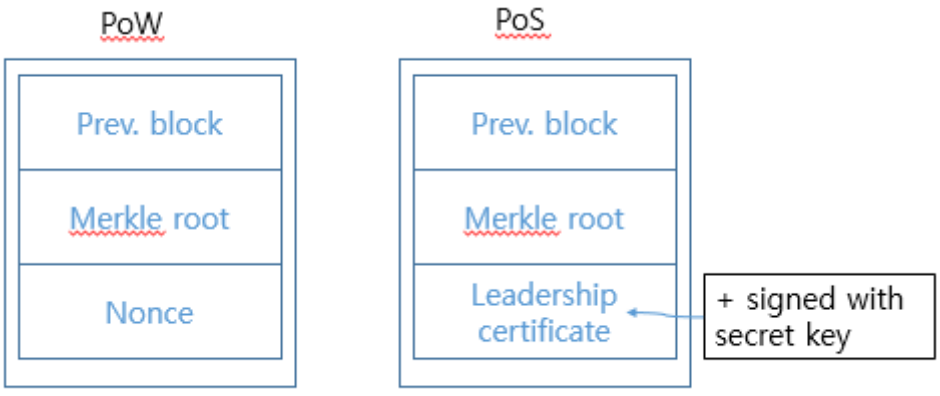

Figure 13.2: headers of PoW systems and PoS systems

PoW systems have three things in the header: previous block, merkle root, nonce. The hash of all these three things is less than a threshold.

$$H(\text{PB, MR, Nonce}) < threshold \tag{13.1}$$

On the other hand, PoS systems have a leadership certificate instead of nonce and the hash of these three things is less than a threshold.

$$H(\text{PB, MR, Public key}) < threshold \tag{13.2}$$

We assume that every valid public key contains 1$, which means that every public key is basically equal. Note that, A participant can have multiple public keys without limit. For PoS systems, nodes do not need to mine, you can immediately check if your public key satisfy this property. One advantage of the PoS system is that it is sybill resistant. However, there still exists a problem that we cannot have a vote if we have no coin. In this PoS system, More compute Power does not guarantee more participation ability.

13.2.1 Proposing a new PoS block

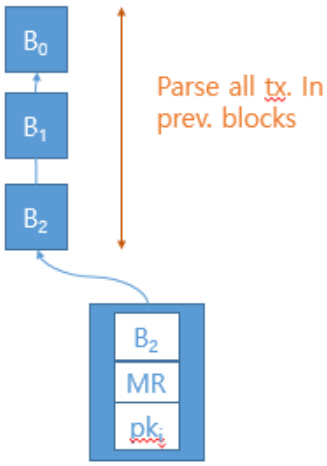

Figure 13.3: Example of Proposing a new PoS block

A new block parses all transactions in previous blocks and the header of a new block contains the previous block B_2, merkle root, and public key i. It proposes a new block if the hash of these three things less than the threshold, i.e.

$$H(PB, MR, pk_i) < threshold * stake(pk_i) \tag{13.3}$$

The right part of inequality above implies that every valid public key contains a stake(the amount of money you have within this public key). You'll increase the threshold and make it easier to solve the puzzle if your public key contains more money.

13.2.2 Types of possible attacks

Can I choose a public key to favor myself? This question leads two types of attacks: Dynamic stake and key grinding attack. In order to narrow down the discussion, however, we assume the static stake scenario where all public key and their stake are defined at genesis.

13.3 How to control block production rate

13.3.1 Differences in PoS

Can control the block production time by changing threshold? The threshold is the parameter we can change to control block production time. For example, given a block B_3, the average number of blocks that follow a block is determined by the threshold. While PoW systems control the block production rate by changing the hash rate, there is no such hash rate in PoS systems. It implies that a node can produce blocks at a very fast rate since there is nothing to throttle the rate.
Let's define the parameter γ,

$$\gamma = \text{(avg. \# of block that follow a block)} \tag{13.1}$$

then there is a forking if $\gamma > 1$, and the blockchain may stall if $\gamma < 1$. Even if $\gamma > 1$, the blockchain can stall since a block can have no child when all of us try out public keys to make a block and none of us are lucky. Note that, a node does not get a second attempt to make a block, but get just one attempt.
Let's summarize the differences in PoS:

1. there is no way to normalize block production rate
2. there is possible stalling of blockchain
3. there is an issue of transaction grinding

What we want to do is design a new system which has stable rate of mining block by solving (1) and (2).

13.3.2 A new parameter: Time t

Here is an idea. We can add one more variable in the hash function: time t. Let's assume you have one public key and every time you have to check one has to tkae this hash and have it less than the threshold. Then you get to the proposal block at that time.

$$H(\text{PB, MR}, pk, t) < threshold * stake(pk) \tag{13.2}$$

This implies that a new block will only be accepted if $t(block) < time(clock)$. This clearly prevents the stalling issue, which solves the issues of (1) and (2). Note that, this design requires at least weak clock synchronization. Error in clock synchronization can add to network latency.

13.3.3 Leadership certificate

How can we solve the issue of (3)? Let's remove the merkle root in the hash. Then it can solve the transaction grinding issue.

$$H(\text{PB}, pk, t) < threshold * stake(pk) \tag{13.3}$$

Note, adversarial nodes can still try to mine on different previous blocks.

13.3.4 Validity rule

First, timestamps on leadership certificate need to be ordered (in the increasing order). Next, it satisfies the hash condition. Further, there is the time constraint: $time(block) < time(clock)$.

13.3.5 Three-level chain structure

There are three parts of blockchain: chain of leadership certificate, chain of headers, and chain of blocks. Each header contains its leadership certificate, merkle root, the previous header hash, and the signature from the secret key; each blocks contain its header and transactions.

13.4 Nothing at Stake Attack

While honest nodes (blue nodes in Fig 13.5) mine only at tip of the longest chain, adversarial nodes (orange nodes in Fig 13.5) can mine on all past leadership certificate, i.e. at each time they can mine a block everywhere on the LC chain. This is Nothing-at-stake attack.

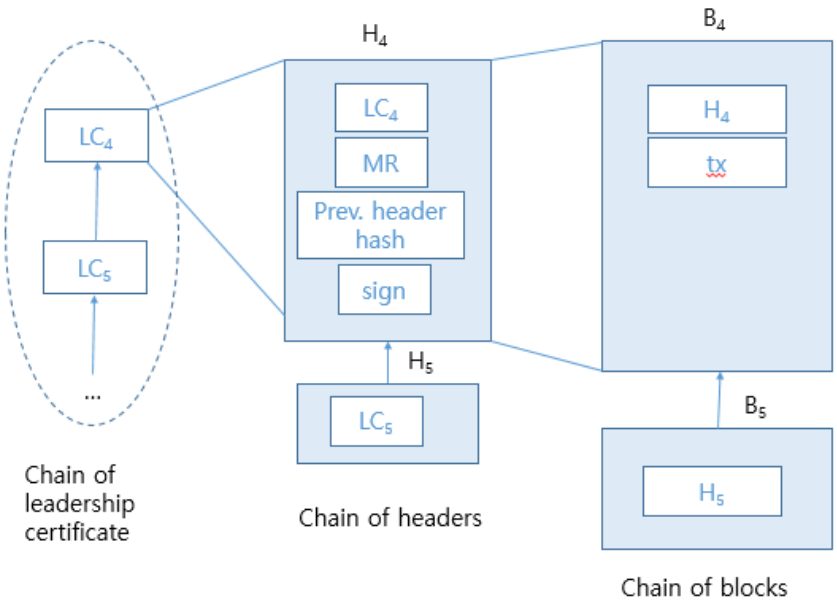

Figure 13.4: three level of chains

In Bitcoin system, you can mine at any blocks on all levels you want, but you have to split your computer power. It means each of them will grow slower. However, for PoS system, each attempts does not grow slower since you are not splitting your stake. There is no conservation of work in PoS. Therefore, adversarial nodes can min on several blocks without splitting their stakes.

References

[1] Ittai Abraham, Dahlia Malkhi, Kartik Nayak, Ling Ren, and Alexander Spiegelman. Solida: A blockchain protocol based on reconfigurable byzantine consensus. *arXiv preprint arXiv:1612.02916*, 2016.

[2] Vivek Bagaria, Sreeram Kannan, David Tse, Giulia Fanti, and Pramod Viswanath. Prism: Deconstructing the blockchain to approach physical limits. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 585–602, 2019.

[3] Yoad Lewenberg, Yonatan Sompolinsky, and Aviv Zohar. Inclusive block chain protocols. In *International Conference on Financial Cryptography and Data Security*, pages 528–547. Springer, 2015.

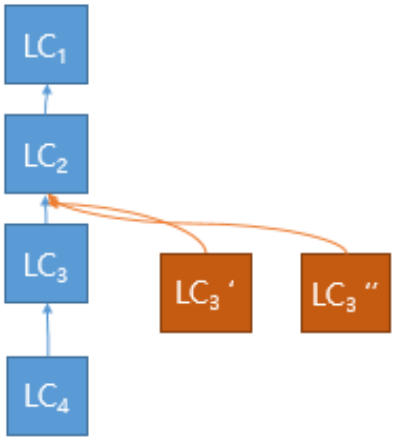

Figure 13.5: Nothing-at-stake attack

[4] Chenxing Li, Peilun Li, Dong Zhou, Wei Xu, Fan Long, and Andrew Yao. Scaling nakamoto consensus to thousands of transactions per second. *arXiv preprint arXiv:1805.03870*, 2018.

[5] Serguei Popov, Olivia Saa, and Paulo Finardi. Equilibria in the tangle. *Computers Industrial Engineering*, 136:160–172, Oct 2019.

[6] Yonatan Sompolinsky, Yoad Lewenberg, and Aviv Zohar. Spectre: A fast and scalable cryptocurrency protocol. *IACR Cryptol. ePrint Arch.*, 2016:1159, 2016.

[7] Yonatan Sompolinsky and Aviv Zohar. Phantom: A scalable blockdag protocol. *IACR Cryptol. ePrint Arch.*, 2018:104, 2018.