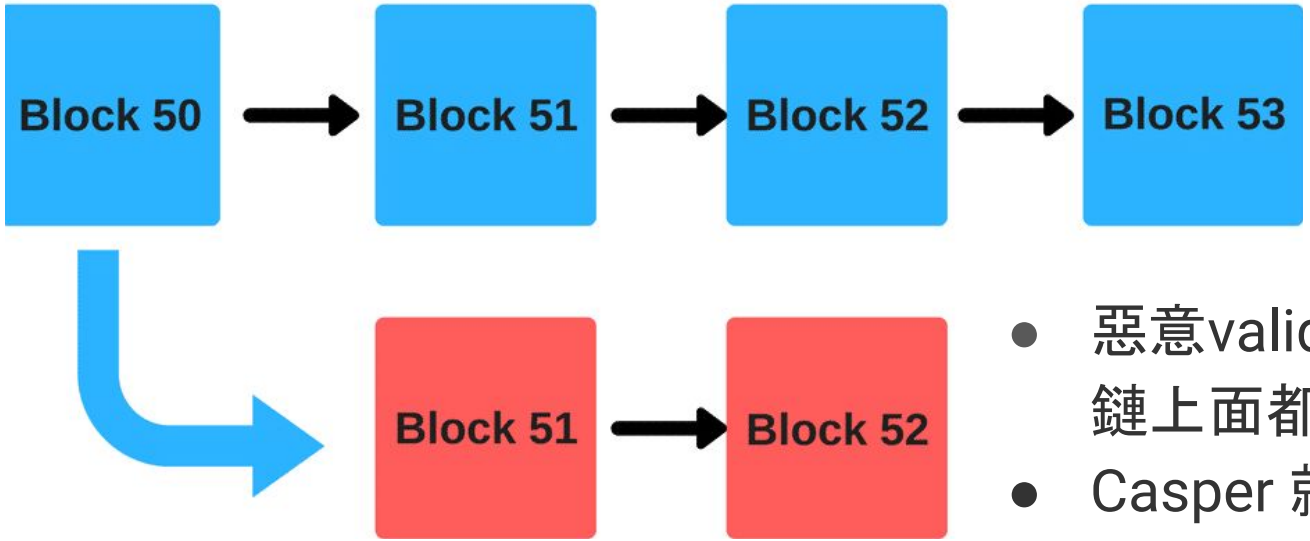
Casper Survey

By Gerber 黃冠博

Problem of POS -- Nothing at Stake



- 惡意validator就在紅鏈跟藍鏈上面都放stake
- Casper 就是想使用PoS，但同時又要避免這個問題

Always win and have nothing to lose,
despite how malicious the actions maybe.

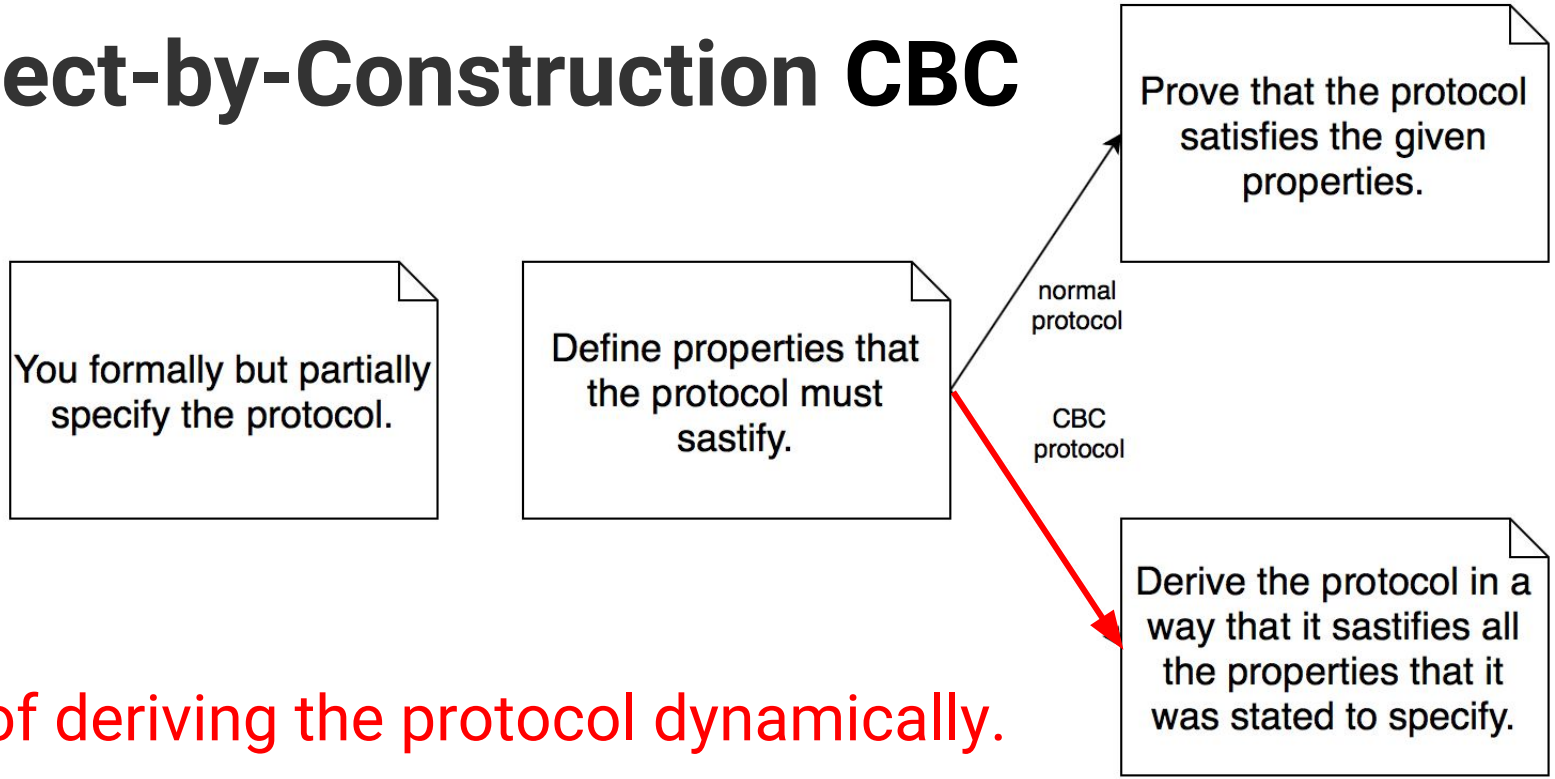
PoS under Casper

- The validators stake a portion of their Ethers as stake.
- When validators discover a block which they think can be added to the chain, they will validate it by placing a bet on it.
- If the block gets appended, then the validators will get a reward proportionate to their bets.
- However, if a validator acts in a malicious manner and tries to do a “nothing at stake”, they will immediately be reprimanded, and all of their stake is going to get slashed.
- 誠實的validator在藍鏈上會拿到獎勵，惡意的validator的stake則會因為下在紅鏈上而被沒收
- 亂離線的validator也會被處罰

Casper = FFG + CBC

- Casper the Friendly Finality Gadget (FFG)
- Casper the Friendly GHOST:
  Correct-by-Construction (CBC)

Correct-by-Construction CBC



Sort of deriving the protocol dynamically.

Casper the Friendly Finality Gadget (FFG)

- A mechanism which proposes blocks.
- While blocks are still going to be mined via POW, every 50th block is going to be a POS checkpoint where finality is assessed by a network of validators.
- Casper protects against finalizing two conflicting checkpoints, but ***the attackers could prevent Casper from finalizing any future checkpoints***.

> 標記粗斜體畫底線的情況為什麼會發生？

Casper 有的 BFT 沒有的

- Accountability
- Dynamic validators
- Defenses
- Modular overlay

Accountability

- If a validator violates a rule, we can detect the violation and know which validator violated the rule.
  - solves “Nothing at stake” problem
- The penalty for violating a rule is a validator’s entire deposit.
- Security of PoS is based on the size of the penalty.
  - The penalty should be greater than the gains from the mining reward.
  - stronger security incentives

Dynamic validators

- validators change overtime

Defenses

- long range revision attacks
- more than 1/3 of validators drop offline
- cost: very weak tradeoff synchronicity assumption

Modular overlay

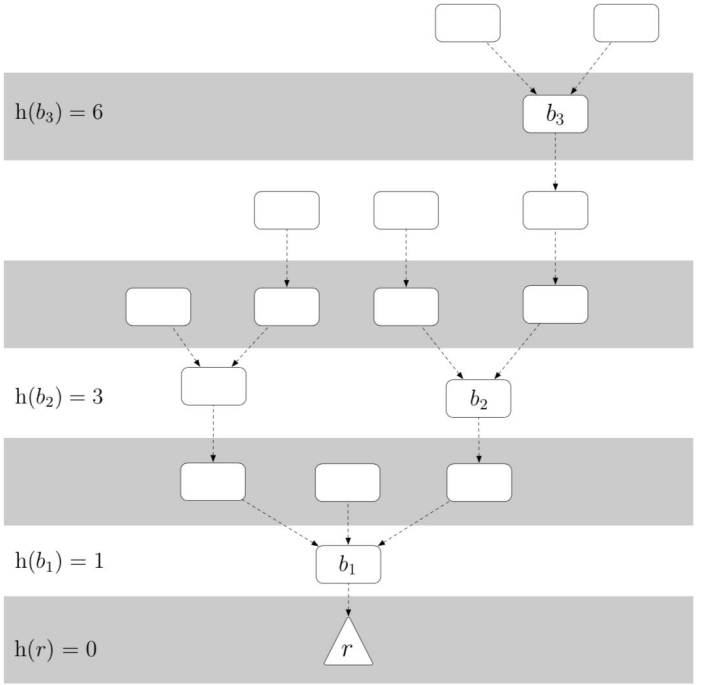
- Casper’s design as an overlay makes it easier to implement as an upgrade to an existing proof of work chain.

The Casper Protocol

- Assume a fixed set of validators and a proposal mechanism.
  - produces child blocks of existing blocks
  - ever-growing blocktree
- 理論上希望: propose blocks one after the other in a linked list
- 現實中: In the case of network latency or deliberate attacks, the proposal mechanism will inevitably occassionally produce multiple children of the same parent.
- Caspers job is to choose a single child from each parent, thus choosing one canonical chain from the block tree.
- 為了效率: Casper only considers the subtree of checkpoints forming the checkpoint tree.

Terms and Definition

- checkpoint:
  - Genesis block (root)
  - every block whose height in the block tree (or block number) is an exact multiple of 100
- height h(c) of a checkpoint c:
  - the number of elements in the checkpoint chain stretching from c all the way back to root along the parent links



(b) The height function

Terms and Definition

- Security of PoS derives from the size of the deposits, not the number of validators.
  - “2/3 of validators”: deposit weight fraction, that is, a set of validators whose sum deposit size equals to 2/3 of the total deposit size of the entire set of validators.
  - 簡單來說，是總錢的2/3，不是總票數2/3

Terms and Definition

- vote message 包含了：
  - two checkpoints s and t
  - $h(s), h(t)$
- **Invalid 的狀況**：
  - s isn't an ancestor of t
  - public key of validator v isn't in the validator set

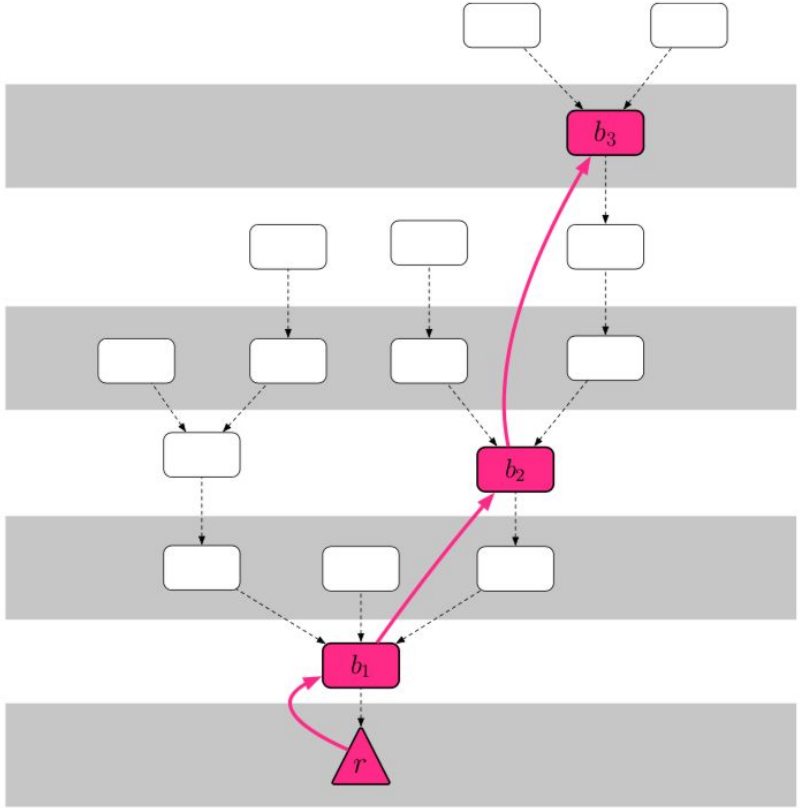
Notation	Description
s	the hash of any justified checkpoint (the "source")
t	any checkpoint hash that is a descendent of s (the "target")
$\mathrm{h}(s)$	the height of checkpoint s in the checkpoint tree
$\mathrm{h}(t)$	the height of checkpoint t in the checkpoint tree
$\mathcal{S}$	signature of $\langle s, t, \mathrm{h}(s), \mathrm{h}(t) \rangle$ from the validator ν's private key

Table 1: The schematic of a single **VOTE** message denoted $\langle \nu, s, t, \mathrm{h}(s), \mathrm{h}(t) \rangle$.

Terms and Definition

- supermajority link:
  - an ordered pair of checkpoints (a,b), also written a ⟶ b, s.t. at least 2/3 of validators (by deposit) have published votes with source a and target b.
  - can skip checkpoints,

    i.e., h(b) > h(a) + 1 is fine



(c) The justified chain $r \rightarrow b_1 \rightarrow b_2 \rightarrow b_3$

Terms and Definition

- **Conflicting**: Two checkpoints a and b are called conflicting iff. they are nodes in different branches, i.e., neither is an ancestor or descendant of the other.
- **Checkpoint**: A checkpoint c is justified if:
  - it is the root
  - there exists a supermajority link c’ → c where c’ is justified
  - Figure (c) shows a chain of four justified blocks.
- **Finalized**: A checkpoint c is called finalized if it is justified and there is a supermajority link c → c’ where c’ is a direct child of c.

Terms and Definition

- Impossible for two conflicting checkpoints to be finalized without ≥ 1/3 of the validators violating one of the two Casper Commandments/slashing conditions.(Figure 2)
- 違反 slashing condition的證據會被當成交易上鏈，押金會被沒收且一小部分會被拿來當作發現者 /舉報者的獎勵
    - 要阻止押金被拿走必須要 51%攻擊

AN INDIVIDUAL VALIDATOR ν MUST NOT PUBLISH TWO DISTINCT VOTES,

$$\langle \nu, s_1, t_1, \mathrm{h}(s_1), \mathrm{h}(t_1) \rangle \quad \text{AND} \quad \langle \nu, s_2, t_2, \mathrm{h}(s_2), \mathrm{h}(t_2) \rangle \ ,$$

SUCH THAT EITHER:

I. $\mathrm{h}(t_1) = \mathrm{h}(t_2)$.
Equivalently, a validator must not publish two distinct votes for the same target height.

OR

II. $\mathrm{h}(s_1) < \mathrm{h}(s_2) < \mathrm{h}(t_2) < \mathrm{h}(t_1)$.
Equivalently, a validator must not vote within the span of its other votes.

Figure 2: The two Casper Commandments. Any validator who violates either of these commandments gets their deposit slashed.

Proving Safety and Plausible Liveness

- Casper's two fundamental properties:
  - **accountable safety**: conflicting checkpoints cannot both be finalized unless ≥ 1/3 of validators violate a slashing condition (也就是說總押金會失去1/3)
  - **plausible liveness**: regardless of any previous events (e.g., slashing events, delayed blocks, censorship attacks, etc.), if ≥ 2/3 of validators follow the protocol, then it's always possible to finalize a new checkpoint without any validator violating a slashing condition.

Proving Safety and Plausible Liveness

Under the assumption that $\frac{2}{3}$ *of the validators by weight do not violate a slashing condition*, we have the following properties:

(i) If $s_1 \to t_1$ and $s_2 \to t_2$ are distinct supermajority links, then $\mathrm{h}(t_1) \neq \mathrm{h}(t_2)$.

(ii) If $s_1 \to t_1$ and $s_2 \to t_2$ are distinct supermajority links, then the inequality $\mathrm{h}(s_1) < \mathrm{h}(s_2) < \mathrm{h}(t_2) < \mathrm{h}(t_1)$ cannot hold.

From these two properties, we can immediately see that, for any height n:

(iii) there exists at most one supermajority link $s \to t$ with $\mathrm{h}(t) = n$.

(iv) there exists at most one justified checkpoint with height n.

Theorem 1 (Accountable Safety). Two conflicting checkpoints a_m and b_n cannot both be finalized.

Proof:

- Let a_m (with justified direct child a_{m+1}) and b_n (with justified direct child b_{n+1}) be distinct finalized checkpoints as in Figure 3.

1. Now suppose a_m and b_n conflict, and without loss of generality

$$h(a_m) < h(b_n)$$

   - if $h(a_m) = h(b_n)$, then it is clear that $\frac{1}{3}$ of validators violated condition **I.**

2. Let

$$r \to b_1 \to b_2 \to \cdots \to b_n$$

   be a chain of checkpoints, such that there exists a supermajority link

$$r \to b1, \cdots, b_i \to b_{i+1}, \cdots, b_n \to b_{n+1}$$

- We know that no $h(b_i)$ equals either $h(a_m)$ or $h(a_{m+1})$, because that violates property (iv).

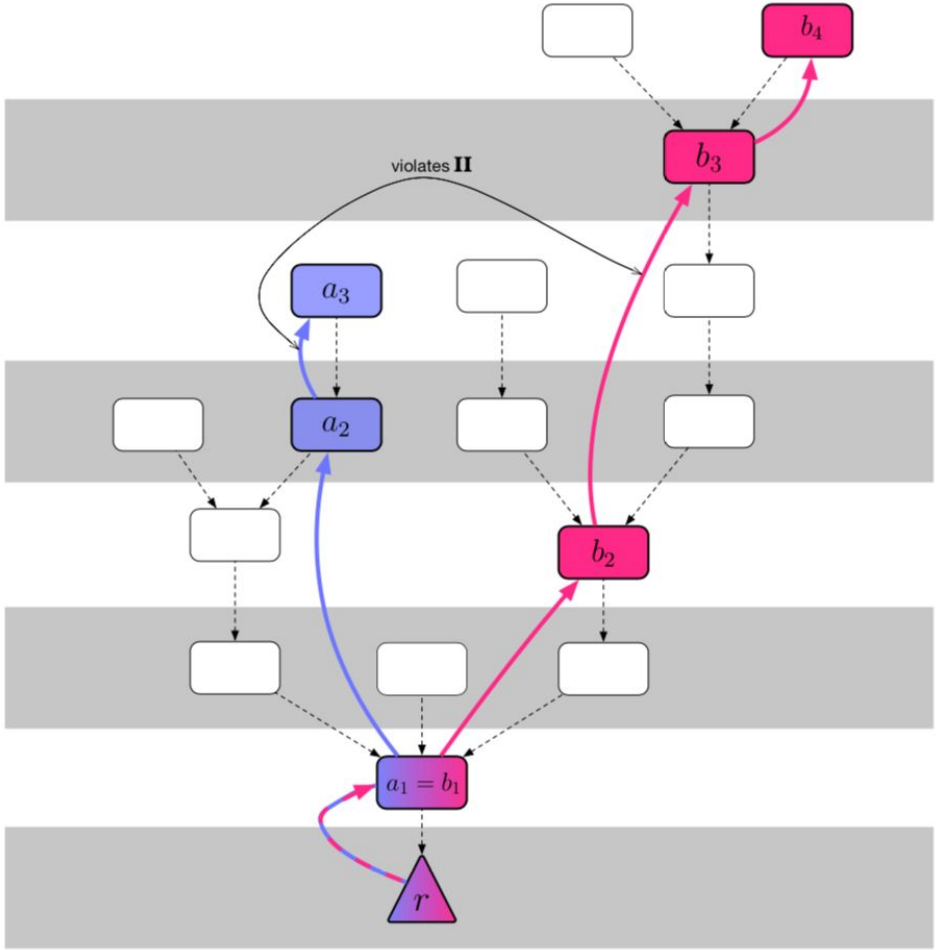


Figure 3: Figure for Theorem 1 (Accountable Safety).

Theorem 1 (Accountable Safety). Two conflicting checkpoints a_m and b_n cannot both be finalized.

3. Let j be the lowest integer s.t. $h(b_j) > h(a_{m+1}) \Rightarrow h(b_{j-1}) < h(a_m)$

> 無法理解第3.
> 他的意思應該是 $h(b_j) > h(a_{m+1}) > h(a_m) > h(b_{j-1})$
> 為什麼不會有這種狀況 $h(b_j) > h(a_{m+1}) > h(b_{j-1}) > h(a_m)$

4. However, this implies the existence of a supermajority link from a checkpoint with an epoch number less than $h(a_m)$ to a checkpoint with an epoch number greater than $h(a_{m+1})$, which is incompatible with the supermajority link from a_m to a_{m+1}.

第4.(紅色底線)是違反II嗎？

II. $\mathrm{h}(s_1) < \mathrm{h}(s_2) < \mathrm{h}(t_2) < \mathrm{h}(t_1)$.
Equivalently, a validator must not vote within the span of its other votes.

附上原文:

Theorem 1 (Accountable Safety)**.** *Two conflicting checkpoints a_m and b_n cannot both be finalized.*

Proof. Let a_m (with justified direct child a_{m+1}) and b_n (with justified direct child b_{n+1}) be distinct finalized checkpoints as in Figure 3. Now suppose a_m and b_n conflict, and without loss of generality $\mathrm{h}(a_m) < \mathrm{h}(b_n)$ (if $\mathrm{h}(a_m) = \mathrm{h}(b_n)$, then it is clear that $\frac{1}{3}$ of validators violated condition **I**). Let $r \to b_1 \to b_2 \to \cdots \to b_n$ be a chain of checkpoints, such that there exists a supermajority link $r \to b_1, \ldots, b_i \to b_{i+1}, \ldots, b_n \to b_{n+1}$. We know that no $\mathrm{h}(b_i)$ equals either $\mathrm{h}(a_m)$ or $\mathrm{h}(a_{m+1})$, because that violates property (iv). Let j be the lowest integer such that $\mathrm{h}(b_j) > \mathrm{h}(a_{m+1})$; then $\mathrm{h}(b_{j-1}) < \mathrm{h}(a_m)$. However, this implies the existence of a supermajority link from a checkpoint with an epoch number less than $\mathrm{h}(a_m)$ to a checkpoint with an epoch number greater than $\mathrm{h}(a_{m+1})$, which is incompatible with the supermajority link from a_m to a_{m+1}. □