Low-cost attacks on Ethereum 2.0 by sub-1/3 stakeholders

Michael Neuder   Daniel J. Moroz
Rithvik Rao   David C. Parkes

School of Engineering and Applied Sciences
Harvard University

Workshop on Game Theory in Blockchain (GTiB)

The 16th *Conference on Web and Internet Economics (WINE)*

December 11, 2020

Introduction
Context

- The Ethereum 2.0 beacon chain launched December 1, 2020.
- Approximately 1 Million ETH (500 Million USD) currently staked.
- High validator participation in the consensus.

- Outline two attacks that can be launched against the Ethereum 2.0 Beacon chain.
    1. Malicious reorgs
    2. Delaying finality
- Demonstrate that for a 30% stake attacker, these attacks are feasible and cheap.

- Intuition:
    1. The fork-choice rule decides between conflicting blocks with same parent by seeing which block has more votes.
    2. An attacker can create a private fork, during which the honest validators vote for the parent block.
    3. This allows the validator to use multiple sets of votes for their private chain and thus outweigh the next honest blocks.
- Impact:
    1. Potential to double spend.
    2. Potential to front run.

- Intuition:
  1. Finality gadget operates on special "checkpoint" blocks.
  2. In order to finalize new blocks, 2/3 of the validators need to agree on one of these checkpoint blocks.
  3. If the attacker is the proposer (block creator) for a checkpoint block, they can delay its release in order to ensure 2/3 threshold is not met.
- Impact:
  1. Temporary DoS on finalization mechanism.
  2. Less predictable network.

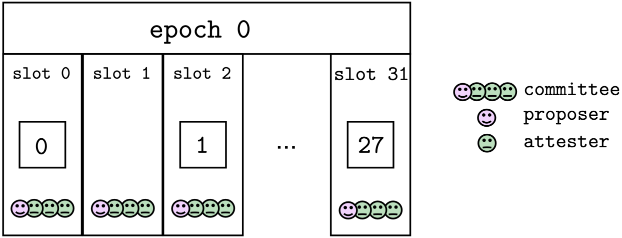
Introduction

Related Work

- Proof-of-Work selfish mining literature from Eyal and Sirer (2013), Nayak et al. (2016), Sapirshtein et al. (2016).
- Longest-chain Proof-of-Stake selfish mining by Brown-Cohen et al. (2019), Neuder et al. (2020).
- Attacks on the beacon chain.
    1. Ebb and Flow attack by Neu et al. (2020).
    2. Decoy Flip-Flop by Ryuya Nakamura (2019).
    3. Bouncing attack by Ryuya Nakamura (2019).

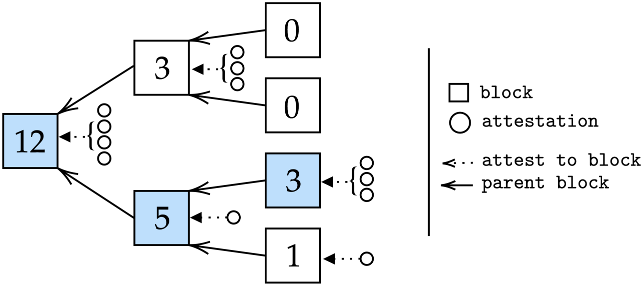
Ethereum 2.0

Proof-of-Stake Basics



- Time is divided epochs, which consists of 12 second slots.
- Each slot has a committee with a single proposer and many attesters.
- Proposers create blocks, and attesters vote.
- Validators are rewarded for participation (as of today 14% annually).

- Combination of two ideas:
    1. Fork-choice rule: HLMD-GHOST (Hybrid Latest Message Driven Greedy Heaviest Observed SubTree).
    2. Finality tool: Casper FFG (Friendly Finality Gadget).



- HLMD-GHOST† uses *weight* to determine the head of the canonical chain.
- Each block annotated with its weight.
- Blue blocks are heaviest branch at each fork and thus part of canonical chain.

† Hybrid Latest Message Driven Greedy Heaviest Observed SubTree

Definition

An <u>attestation</u> is the casting of a vote that contains:

A_1 – a source epoch boundary block

A_2 – a target epoch boundary block

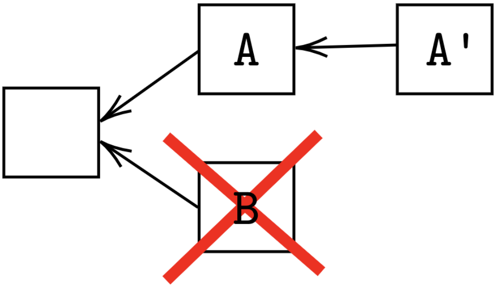
A_3 – the head of the chain according to HLMD-GHOST

- For now we only consider A_3, which is the result of HLMD-GHOST.

Malicious Reorgs

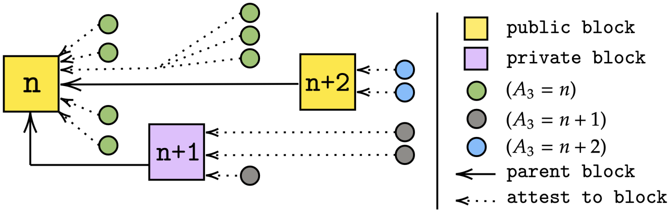
Definition

- Chain reorganizations, or reorgs, occur when a conflicting fork is determined to be dominant over the existing canonical chain.



Malicious Reorgs

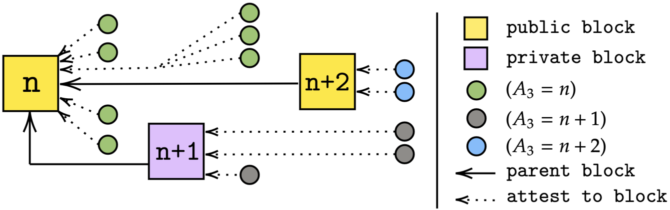
Strategy

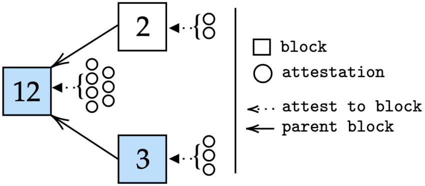


- The attacker privately proposes block $n + 1$ and attests with $(A_3 = n + 1)$. Honest validators instead attest with $(A_3 = n)$.
- At slot $n + 2$, an honest validator will propose a block whose parent is the slot n block.
- The attacker then releases private attestations and block $n + 1$, which is seen as the head of the chain by HLMD-GHOST.

Malicious Reorgs

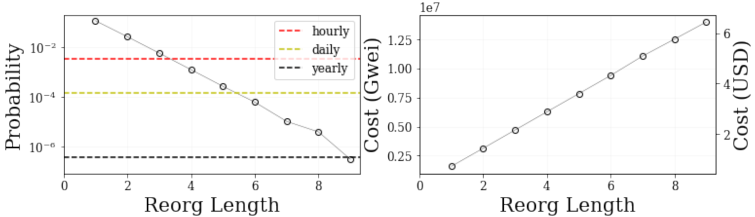
Strategy





Malicious Reorgs

Probability



- Use Monte Carlo simulation of 10^7 randomly generated epochs.
- In this case, we only consider reorgs that occur *within a single epoch*.
- Cost is the amount of reward lost, or the opportunity cost of playing this dishonest strategy.

Ethereum 2.0

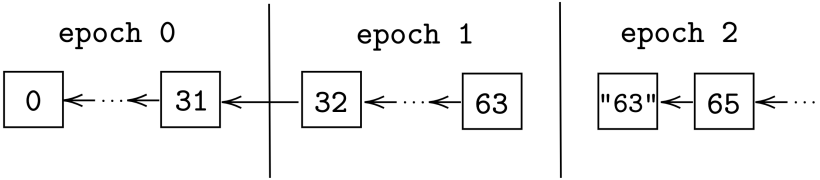
Finality

- Finality is a property of blocks.
- Casper FFG* from Buterin and Griffith (2017) operates on top of a blockchain and determines which blocks are finalized.
- The first block in the chain is finalized, and the gadget moves monotonically up in block height, using "checkpoint" blocks.
- Design rationale: More efficient to use checkpoints rather than finalizing each block individually.

*Friendly Finality Gadget

Ethereum 2.0

Epoch Boundary Blocks (EBBs)



- Block 32 is the EBB for epoch 1 because it is the first block of the epoch.
- Block 63 is the EBB for epoch 2 because the expected first block of the epoch, block 64, was not published.

Definition

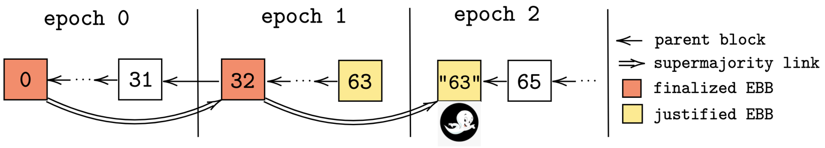
An <u>attestation</u> is the casting of a vote that contains:

A_1 − a source epoch boundary block

A_2 − a target epoch boundary block

A_3 − the head of the chain according to HLMD-GHOST

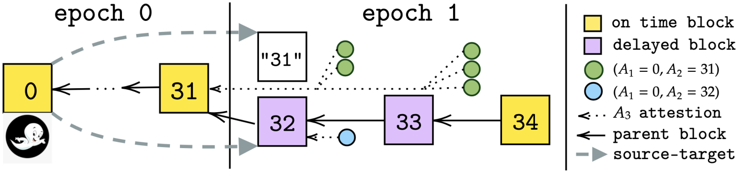
- An attestation with $(A_1 = \beta, A_2 = \gamma)$ means, "I want to move the finality gadget from EBB β to EBB γ".
- If $2/3$ of the validators attest with $(A_1 = \beta, A_2 = \gamma)$, we say a *supermajority link* is created, and the gadget is moved.



- An EBB becomes justified when the gadget *lands* on it.
- An EBB becomes finalized when the gadget is *moved* from that block to the next epoch's EBB.

Delaying Finality

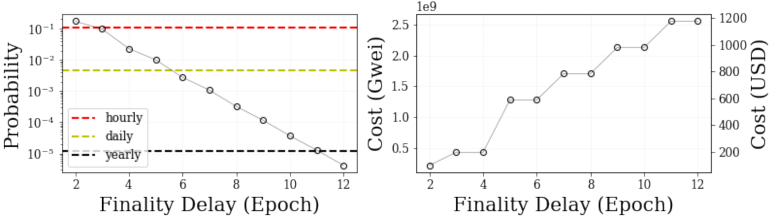
Strategy



- Attacker privately creates block 32, which is an EBB for epoch 1.
- Since the honest validators do not see the EBB (block 32), they attest with $(A_1 = 0, A_2 = 31) \implies$ "Move gadget from 0 to 31".
- Repeat previous two steps with block 33.
- Attacker releases private fork and withholds all remaining attestations.

Delaying Finality

Probability



- The 30% attacker has probability $(0.3)^2 = 0.09$ of forcing non-justified epoch.
- In order to ensure none of next *n* epochs are finalized on time, attacker needs to ensure that no two epochs in a row are justified.
- Cost is the amount of reward lost, or the opportunity cost of playing this dishonest strategy.

Conclusion

- Summary
  - Ethereum Proof-of-Stake
  - Malicious Reorgs
  - Delaying Finality
- Future work
  - Quantifying the impact of attacks.
  - Mitigation of attacks.

Thanks!