TL; DR:

this article begins by assessing the safety and liveness of the beacon chain, eth1 and one-way bridge. From there, an adjustment to the one-way bridge is suggested to accommodate the finality gadget. Finally, safety and liveness is re-analyzed with the finality gadget included.

### Background reading

- Eth1 data voting in the spec
- The finality gadget by Alex Stokes
- Two-way bridges between eth1 and eth2 by Vitalik

## Analysis of Eth1 -> Eth2 bridge

### Eth1 data voting overview

Eth1 data voting mechanism is implemented to enable beacon chain validators deposited on Eth1 chain. This is achieved by bringing Eth1 block hash and root of deposit tree to the beacon chain. During period of time called voting period deposit root is being updated in the beacon state and after that proposers start processing a new portion of deposits.

Since Eth1 has no re-org limitation on the protocol level there is a reasonably high ETH1_FOLLOW_DISTANCE = 1024

to protect Eth1 voting mechanism from accidental re-org happening inside of voting period. Under honest majority model probability of re-org with depth equal to 1024

is effectively a 0

. In other words this distance is safe enough.

A number of proposers in a voting period is parametrised by SLOTS_PER_ETH1_VOTING_PERIOD = 1024

. Each proposer should include its Eth1 vote into a block. In order to do that an honest

proposer reads Eth1 data submitted by its precursors and tries to match it with its own view of Eth1 chain. Depending on the match it submits either its own view or the view that has already been included. Finally decision around winning Eth1 data is made according to WINNER_THRESHOLD = n/2 + 1

parameter.

With mean time of Eth1 block and seconds per beacon slot we can calculate a delay between inclusion of data into Eth1 chain and reaching consensus around this data on beacon chain side. The formula is straightforward
ETH1_BLOCK_MEAN_TIME*ETH1_FOLLOW_DISTANCE +*
*SECONDS_PER_SLOT*SLOTS_PER_ETH1_VOTING_PERIOD

. Suppose Eth1 block mean time is 14

seconds then we get 14*1024 + 12*1024 = ~7.4 hours

.

A side effect of this voting mechanism is finalization of Eth1 blocks. When new beacon chain checkpoint gets finalized it implicitly finalizes all Eth1 blocks that were included into the chain prior to that checkpoint. In order to estimate time to Eth1 finality

we can base on the formula above and add a time required to finalized beacon chain checkpoint. In the best case time to beacon chain finality equals to 1

epoch. Thus, time to Eth1 finality equals to 14*1024 + 12*1024 + 12*32 = ~7.5 hours

or 1929

Eth1 blocks in the best case.

### Safety and Liveness

Eth1 and Eth2 are merged into a composite system by the one-way bridge. Safety and liveness of this system is shared between its components and parametrised by bridge's constants. In order to formulate safety and liveness assumptions we need to analyze both chains in the context of one-way bridge implementation.

We know that Eth1 as a regular PoW system tolerates f < n/2

failures. In practice, this boundary means that both safety and liveness are preserved unless adversary owns 51% of hash power.

Beacon chain safety and liveness thresholds are based on Casper FFG and inactivity leak properties. Casper FFG safety and liveness tolerates f < 2n/3

and f <= n/3

failures respectively. However, adversary that owns >= 1/2

of total stake is able to split the network and due to inactivity leak start finalizing conflicting checkpoints. Thus, inactivity leak reduces safety fault tolerance to f < n/2

. Sacrificing safety of the system inactivity leak significantly increases its liveness but diving deep into details about that is not a goal of this article.

We assume that beacon chain safety and liveness tolerates f < n/2

and f <= n/3

failures respectively. This is pretty enough for further analysis.

**Deep re-orgs in Eth1**

Having a majority of hash power adversary is able to build an alternative fork which eventually beats canonical chain and causes deep re-org in the system that could result in a successful double spend attack. This kind of attack could be run against the bridge in a following way.

Adversary deposits some amount of ETH and waits until the end of voting period. After that adversary reveals an alternative Eth1 chain which either creates new or top ups already onboarded validators. Depth of the revealed chain should be at least equal to ETH1_FOLLOW_DISTANCE

and adversary will have to keep this chain progressing for at least SLOTS_PER_ETH1_VOTING_PERIOD/2 + 1

to successfully finish the attack.

**Dishonest voting majority**

On the other hand if adversary owns a majority of proposers during the voting period it could slow down voting progress or vote for alternative Eth1 chain with no regards to the length of this chain. In the context of the bridge the former would be a liveness failure the latter – safety failure.

Probability of adversary to get a majority of voting power with respect to its total power is shown in a table below. It’s been calculated with SLOTS_PER_ETH1_VOTING_PERIOD = 1024

.

Adversary power

Chance to get a voting majority*

0.33

1.59E-28

0.38

2.49E-15

0.39

4.02E-13

0.40

4.06E-11

0.43

2.89E-06

0.45

5.97E-04

0.47

0.025

0.49

0.250

0.50

0.488

*

Estimated with Binomial distribution.

Considering 5.55E−15

as a safety threshold the bridge is safe as long as adversary owns no more than 38%

of total stake. On the other hand if adversary owns 41%

of stake it gets a majority of voting power once per 149,409

years which sounds like a reasonably safe threshold. It stays between beacon chain liveness and safety thresholds which are mentioned above.

There is a thing about Eth1 voting that makes analysis more sophisticated. Beacon chain safety and liveness are coupled with Casper FFG and inactivity leak which are limiting adversary power by a portion of total stake. While bridge's voting power is determined by a portion of validators' registry size. In practice it means that safety threshold of the bridge is not coupled with safety threshold of the beacon chain. Which with current constants opens an attack vector described in this issue.

Switching bridge's voting power from validator count to validators' effective balance would significantly simplify analysis of bridge's safety. As in that case bridge's safety would be tightly coupled with beacon chain once. The main problem with this change in the context of proposers voting is that voting balance will be spread across a number of epochs (currently 32

) which increases implementation complexity. Hence, doesn't address analysis complexity either.

### Safety and liveness statements

Taking the above in account, there are several statements that oultines bridge's safety and liveness.

#### Eth1 statement

Bridge safety is preserved unless re-org with depth ETH1_REORG_DEPTH_LIMIT

on Eth1 is possible.

ETH1_REORG_DEPTH_LIMIT = ETH1_FOLLOW_DISTANCE + (SLOTS_PER_ETH1_VOTING_PERIOD/2 + 1) * SECONDS_PER_SLOT / SECONDS_PER_ETH1_BLOCK = 1464

Note:

ETH1_REORG_DEPTH_LIMIT

depends on SLOTS_PER_ETH1_VOTING_PERIOD

.

#### Beacon chain statement

Given current voting parameters bridge safety and liveness are preserved until adversary power on the beacon chain is no more than 41%

of total stake.

Current voting parameters are:

- SLOTS_PER_ETH1_VOTING_PERIOD = 1024
- WINNER_THRESHOLD = 1/2

Including specific voting parameters in this statement is essential. Cause, for example, reduction of SLOTS_PER_ETH1_VOTING_PERIOD

increases a chance of adversary to exploit bridge in its attack on the beacon chain.

## Practical finality

The reasons behind introducing finality gadget on Eth1 arewell described by Alex Stokes. An in their context having a time to finality about 2000

blocks seems impractical. A target value for it should be something like 64

or even 32

blocks to preserve the same UX level for users of Eth1 applications.

First of all, it requires a significant speed up of the voting mechanism. A solution for this could be found in reduction of voting period length and Eth1 follow distance. But significant reduction of SLOTS_PER_ETH1_VOTING_PERIOD

drastically weakens bridge's security which is shown in a table below.

Probability of getting a majority of voting power by adversary holding 1/3 of total stake

SLOTS_PER_ETH1_VOTING_PERIOD

Probability*

Years to occurrence

1024

1.59E-28

2.45E+24

512

2.77E-15

7.03E+10

256

1.35E-08

7201

128

3.45E-05

1.41

64

0.002

0.01

*

Estimated with Binomial distribution.

We can see that voting period lengths starting from 128

slots significantly reduce bridge's security level. Therefore, we have to look for another solution for Eth1 data voting.

### Voting attesters

An obvious move is to switch voting burden from proposers to attesters. Attesters will have to include the same Eth1Data

into attestations as proposers do. Once epoch is finished Eth1Data

is updated with a sample that collects 2/3

of total amount of effective balance.

This approach would reduce time to Eth1 finality to two epochs in the best case. During the first epoch attesters come to consensus around new Eth1Data

and an epoch after it gets finalized. This solution reduces time to finality to something like 54

Eth1 blocks.

The main drawback with this approach is that it adds more stress to the network layer. It results into additional 64 bytes

per attestation which increases block size up to 8Kb

and a size of single attestation message by about 50%

of its current size.

## Voting committees

The other solution would be to treat Eth1 as a kind of system shard. This solution looks like a precursor of shard chain “crosslinking”, thus, sounds more organic to the beacon chain and its roadmap than the previous one.

Attesters of the committee assigned to Eth1 shard votes for a block that is ETH1_FOLLOW_DISTANCE

behind the block that was a head of Eth1 chain at the beginning of current slot. Eth1Data

getting 2/3

of effective balance of the committee wins in the same fashion as it was in a deprecated for Phase 0 crosslink mechanism.

During epoch processing Eth1Data

in the state is updated with the one that is elected by the committee of the most recent slot. This solution doesn’t introduce additional stress on the network and block size cause its data complexity is near the same as a shard crosslinking that we’re going to see starting from Phase 1.

This approach reduces time to Eth1 finality to SLOTS_PER_EPOCH + 1

which is about 29

Eth1 blocks in the best case.

## Reducing Eth1 follow distance

The other parameter requiring an adjustment on the way to reaching practical finality is ETH1_FOLLOW_DISTANCE

. With assumption of honest majority of Eth1 miners this parameter could be significantly reduced. Table below shows probability of inagreement of honest majority of miners for various chain lengths.

A number of blocks

Probability of no agreement*

1024

~0.0

128

9.48E-115

64

9.74E-58

32

3.12E-29

16

5.59E-15

8

7.47E-08

*

Estimated with homogeneous Poisson process. With 14

seconds as mean time of the block and 1

second as network delay.

16

blocks threshold looks like a reasonably safe margin to justify about agreement around canonical chain by honest majority. Which is much less than values that most of merchants use as a number of blocks to confirm transactions. An explanation of it is that the cost of double spend attack is proportional to a number of blocks required for confirmation.

Since finality gadget gives a protection from double spend attacks we can reduce follow distance to whatever sane value we decide upon.

By the end of the day with ETH1_FOLLOW_DISTANCE = 16

and a voting committees approach to one-way bridge we have 45

blocks as a time to Eth1 finality which looks pretty good.

## Revisiting Safety and Liveness

One of the previous sections touches safety and liveness properties of beacon chain and one-way bridge that provides a link between beacon chain and Eth1. In this section we analyze safety and liveness of Eth1 system featured with finality gadget. We assume that Eth1 miners respect finality produced by the bridge. Which results in a fact that honest miners never revert a block that has been finalized by the beacon chain.

#### Modified fork choice

Understanding a modified fork choice rule of Eth1 system is one of the keys to our further analysis.

Currently Eth1 uses GHOST as a fork choice rule with total difficulty as a function of weight. Finality gadget affects this rule in a following way.

In order to determine which chain is canonical one should pick the chain with the most recent finalized checkpoint and apply a GHOST rule for all sub-trees started from that checkpoint.

Now we're ready to determine safety and liveness properties for modified Eth1 system.

#### Safety and Liveness

Safety of modified Eth1 system is preserved unless two conflicting checkpoints are finalized.

To completely understand this statement we need to give a definition of conflicting checkpoints. Two checkpoints are considered to be conflicting if they belong to different sub-trees. It means that if a checkpoint is not an ancestor of the other one then they are considered as conflicting.

We know that agreement around finalized checkpoints is made by Eth1 data voting mechanism. In application to modified Eth1 system it means that Eth1 safety is supplied by that mechanism.

We, also, know that security level of the voting mechanism depends but not equal to security level of the beacon chain. Elimination of this dependency would be highly wanted as a prerequisite to finality gadget as it removes one of the moving parts from the field of protocol analysis.

Liveness of modified Eth1 system is preserved until new checkpoints are getting finalized. This statement implies that two aspects of the liveness. The first one is a progress of Eth1 chain itself. The other one is a progress of Eth1 chain finality which is provided by Eth1 data voting.

**Practical meaning**

In practice it means that Eth1 safety becomes no longer vulnerable to attacks driven by 51% of hash power that could result in a double spend. However, liveness could be still violated by such kind of attacks. An adversary is able to cencor users' transactions or sabotate voting process by flip-floping with two competing forks. Therefore, hash power owned by honest nodes should be significant enough to prevent liveness attacks by making them cost ineffective for attacker.

In the worst scenario adversary could be able to attack safety by exploiting liveness vulnerability. For example, it could censor deposit transactions and gain an amount of stake significant enough to violate beacon chain safety. If this would be the case then Eth1 safety could be violated too. However, this kind of attack requires pretty big amount of ETH and its a big question whether it would be cost efficient or not.

Readers might already notice that a huge peace of design is assumed here. Which is Eth1 <- Eth2 backwards bridge mechanism which is out of the scope of this write up. One obvious thing that worth noting is that properties of modified system will be shared with backwards bridge as well and their analysis is in high dependcy from particular approach to that bridge.

# When finality gadget?

Changing Eth1 voting scheme to somewhat that provides us with practical finality is a prerequisite for finality gadget. The other part of answer to this question lays in the field of safety and liveness. As we don't want a security flaw on Eth1 side caused by finality gadget introduction.

We assume that one-way bridge safety and liveness are tightly coupled with corresponding properties of the beacon chain. It could be achieved by solutions previously described in this write up or by any other way.

To make further estimation we, also, assume the following:

- two-way bridge scheme doesn't affect safety and liveness of the system
- a cost of 1 hour of 51% attack is about 10x

of the cost estimated by[https://www.crypto51.app/](https://www.crypto51.app/)

Basically, it's hard to estimate what the cost of 1 hour of attack would be if one decided to attack Eth1. It would depend on the duration of the attack and other factors like availability of hash power on the market. We just take reasonably high margin assuming that picked number is not less than the real one would be. To the point of time when this article has been written cost of 1 hour of attack is estimated as 6000 ETH

.

Table below matches beacon chain safety and liveness violation to Eth1 re-org of certain depth. We want to answer to the following question. What an attacker that able to violate beacon chain safety and liveness can do with Eth1 chain?

ETH at stake

Re-org depth worth of liveness failure*

Re-org depth worth of safety failure*

2M

29959

44938

3M

44938

67407

4M

59918

89877

5M

74897

112346

*

Evaluated with 6000 ETH as a cost of 1 hour attack and 14 seconds as a mean time of Eth1 block.

These numbers are far beyond any reasonable re-org depth required forsuccessful double spend attacks. Therefore, with initial amout of deposited ETH (2M

) beacon chain looks like a reliable safety supplier for Eth1 system.

However, this statement works only with assumption that there is no vulnerability in two-way bridge and the beacon chain that could diminish these boundaries. This assumption is partially relaxed by formal verification of the spec and formal proves of particular principles laying in the foundation of the beacon chain design. But by the end of the day it is community to decide when beacon chain is mature enough to take a burden of finality gadget.