BREWFORCE

Staketable Review

Summary

Project Name	Espresso
Language	Solidity & Rust
Codebase	https://github.com/EspressoSystems/espresso-network
Delivery Date	11/08/2025
Team	0xKato, Jarred Parr
Commit(s)	https://docs.google.com/spreadsheets/d/1YmwJxqkZg8HPsDTBFnzreiOztJZyjEiW3cL3e-A1nqg/edit?gid=0#gid=0

Vulnerability Summary

Vulnerability Level	Total	Pending	Declined	Acknowledged	Partially Resolved	Resolved
● Critical	0	0	0	0	0	0
● High	1	0	0	0	0	1
● Medium	4	0	0	2	0	2
● Low	3	0	0	1	0	2

Stake-0 | delegatedAmount used incorrectly

Category	Severity	Location	Status
Implementation Error	● High	StakeTableV2.sol	Confirmed

Description

The `staketablev2` contract has a variable named `delegatedAmount` this is meant to showcase the economic security of the entire espresso network, the problem is in the way that the espresso network decides on which validators gets to participate in the network, currently only the 100 highest delegated validators get to participate in the network which means that the economic security of the espresso network is equal to the top 100 highest delegated validators `delegatedAmount` and not every nodes `delegatedAmount`.

Recommendation

Consider either specifying weather or not `delegatedAmount` should be the metric for economic security of the entire espresso network or make sure sure to specify that any delegated amount only can be considered part of the economic security if the validator is in the top 100 highest delegated.

Resolution

Stake-1 | registerValidatorV2 can be called when paused

Category	Severity	Location	Status
Implementation Error	● Medium	StakeTableV2.sol	Confirmed

Description

The staketablev2 contract has is utilizing a pause functionality to stop operations of the smart contract, the registerValidatorV2 function does not have the whenNotPaused modifier which can lead to validators registering when they should not be allowed to.

Recommendation

Consider adding the whenNotPaused to the registerValidatorV2 function.

Resolution

Stake-2 | updateExitEscrowPeriod can be called when paused

Category	Severity	Location	Status
Implementation Error	● Medium	StakeTableV2.sol	Acknowledged

Description

The `staketablev2` contract has is utilizing a pause functionality to stop operations of the smart contract, the `updateExitEscrowPeriod` function does not have the `whenNotPaused` modifier which can lead to the case where a contract can be paused update `updateExitEscrowPeriod` can be called and change the `EscrowPeriod` from min to the max without users being able to act.

Recommendation

Consider adding the `whenNotPaused` to the `updateExitEscrowPeriod` function.

Resolution

Stake-3 | No grace period for unpausing the contract

Category	Severity	Location	Status
Implementation Error	● Medium	StakeTableV2.sol	Acknowledged

Description

There is no grace period on unpausing the contracts this can lead to a case where the contract is paused and before allowing users to act an upgrade or state change like stake-2 can take place without users having the ability to act.

Recommendation

Consider adding a grace period to the upgrade so users have time to act.

Resolution

Stake-4 | delegatedAmount is incorrectly wiped on exit

Category	Severity	Location	Status
Implementation Error	● Medium	StakeTableV2.sol	Confirmed

Description

The function deregisterValidator sets the validators[validator].delegatedAmount to zero when in reality the delegated amount is the full amount until users have exited, this can lead to incorrect tracking of the total delegated amount.

Recommendation

Consider decreasing the delegatedAmount when users exit rather than when the validator calls exit to always represent the correct value.

Resolution

Stake-5 | Missing schnorr signature length check

Category	Severity	Location	Status
Implementation Error	● Low	StakeTableV2.sol	Confirmed

Description

Functions like updateConsensusKeysV2 and registerValidatorV2 takes in schnorrSig but it does not ensure that the signature length is 64 bytes.

Recommendation

Consider adding a check to enforce the signature length is 64 bytes in length.

Resolution

Stake-6 | missing bounds on _exitEscrowPeriod in constructor

Category	Severity	Location	Status
Incorrect Violation	● Low	StakeTable.sol	Confirmed

Description

In the staketablev2 contract the Escrow Period is bounded between a min and max period but in the staketable contract there is no such restrictions imposed on the Escrow Period this could lead to a scenario where the EscrowPeriod is set to something that is out of the bounds leaving them pointless.

Recommendation

Add initial restriction on the Escrow Period

Resolution

Stake-7 | Missing check in FetchWithMajority

Category	Severity	Location	Status
Incorrect Violation	● Low	StakeTableV2.sol	Acknowledged

Description

Calling updateConsensusKeysV2 does not clear a user's old entry in the blsKeys mapping when updating to a new entry.

Recommendation

Add a check to enforce that more than 1 node is set.

Resolution

Disclaimer

This report is an internal review and should not be considered an “endorsement” or “disapproval” of any particular part of the codebase. It does not provide any warranty or guarantee regarding the absolute bug-free nature of the analyzed technology, nor does it reflect the economics, value, business model, or legal compliance.

This report should not be used to make investment or involvement decisions. It is not a substitute for external reviews and should not be taken as investment advice. Instead, it serves as part of an internal assessment process aimed at helping improve the quality of the code.

The goal is to help reduce attack vectors and risks associated with evolving technologies, we do not claim any guarantees regarding security or functionality of the technology analyzed. We do not guarantee the explicit security of the audited code, regardless of the findings.