SOLIDIFIED

Audit Report for Hermez on October 18, 2020

Summary

Audit Report prepared by Solidified covering the Hermez protocol smart contracts (and their associated components).

Process and Delivery

Three (3) independent Solidified experts performed an unbiased and isolated audit of the code below. The debrief took place on October 12, 2020.

The team provided replies to the issues reported on October 18, 2020. These are reflected in this report.

Audited Files

The following contracts were covered during the audit:

```
contracts
├── auction
│   └── HermezAuctionProtocol.sol
├── hermez
│   ├── Hermez.sol
│   ├── interfaces
│   │   ├── AuctionInterface.sol
│   │   ├── VerifierRollupInterface.sol
│   │   ├── VerifierWithdrawInterface.sol
│   │   └── WithdrawalDelayerInterface.sol
│   └── lib
│       ├── HermezHelpers.sol
│       └── InstantWithdrawManager.sol
│
├── math
│   └── SafeMathUint128.sol
│
└── withdrawalDelayer
    └── WithdrawalDelayer.sol
```

The contracts were supplied in the repositories:

https://github.com/hermeznetwork/contracts

SOLIDIFIED

Audit Report for Hermez on October 18, 2020

Notes

The audit was based on commit `60e03e981f1ce607c27d405952bfc98de376f0c5`.

Intended Behavior

The smart contracts implement the on-chain components of the Hermez protocol which include the following components:

- Support for forging batches of transactions by submitted to L1 and L2 by verifying a proof submitted by a coordinator
- An auctioning protocol used to select the coordinator (forger)
- An emergency mechanism implemented through a withdrawal delay contract

SOLIDIFIED

Audit Report for Hermez on October 18, 2020

Executive Summary

Solidified found that the Hermez protocol contracts contain 3 issues and 5 informational notes.

We recommend all issues are amended, while the notes are up to the team's discretion, as they refer to best practices or optimizations.

Issues found:

Issue #	Description	Severity	Status
1	Potential Shortcomings with exchange rates	Minor	Acknowledged
2	Donation Address can be 0, leading to burned tokens	Minor	Acknowledged
3	Tokens charging token transfer fee would cause balances accounting discrepancy	Minor	Acknowledged
4	Malleable signatures accepted	Note	Acknowledged
5	Endpoint Information for Coordinators is not Enforced	Note	Acknowledged
6	Roles in WithdrawlDelayer can be set to address 0	Note	Acknowledged
7	Saving constants with the hash result is cheaper than saving the hashing operation	Note	Acknowledged
8	Exit timestamps are only recorded for the last exist	Note	Acknowledged

SOLIDIFIED

Audit Report for Hermez on October 18, 2020

Issues Found

Critical Issues

No critical issues were found.

Major Issues

No major issues were found.

Minor Issues

1. Potential Shortcomings with exchange rates

It's unclear how the governance process will work, but being the entity responsible for setting token exchange values, it could lead to potential issues.

For example, recently registered tokens will have values set to 0, allowing for instant withdrawals.

Another possible ramification to this issue is that deposited tokens can exceed the max amount and become locked due to variance in the token price.

It may also react poorly to the steep changes in price and a short period, making withdrawals happen later or earlier than they actually should due to which bucket the USD amount falls into.

Recommendation
There's no clear and easy fix for this issue and we recommend the Hermez team to take the points presented here and ponder how and if it should be addressed.

Team Response
"This mechanism is intended to minimize the losses in case of catastrophic failure. The idea is to limit (but not remove) the loss that the system can have.
We plan to remove this mechanism at some point when we get confidence that the system works ok."

2. Donation Address can be 0, leading to burned tokens

In `HermezAuctionProtocol.sol`, the `setDonationAddress()` function does not verify that the donation address is not set to 0. This may lead to inconsistent distribution ratios with additional tokens being burned. If no donation is the desired behavior, it is better to set the donation ratio to 0%.

Recommendation
Check the parameter for address 0.

3. Tokens charging token transfer fee would cause balances accounting discrepancy

There are some tokens that charge a transfer fee (for example DGX, https://digix.zendesk.com/hc/en-us/articles/360039741872-What-are-the-fees-associated-with-DGX-). This will result in accounting discrepancy:

- Hermez.sol will actually receive fewer tokens than indicated by the amount. But the amount "moved" to L2 will be the original deposit amount (higher).
  This will eventually cause the withdrawal (exit) problems when the Hermez contract's token balance will not be sufficient for an exit.
- WithdrawalDelayer.sol will also receive fewer tokens than indicated by the amount. Since WithdrawalDelayer.sol does not allow specifying a withdrawal amount, it might happen that some users will not be able to withdraw the tokens at all due to the insufficient WithdrawalDelayer.sol contract's token balance.

Recommendation
There is no clear solution to this problem. We advise Hermez to make users aware of this issue and recommend not to move such tokens into L2

Team Response
"We will follow the recommendation to advise people not to use such a token."

Notes

4. Malleable signatures accepted

The `_checkSig()` in `HermezHelpers.sol`, the built-in `ecrecover()` function is used. This function still allows malleable signatures for backward compatibility reasons. Signatures that have an `s` value larger than `0x7FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF5D576E7357A4501DDFE92F46681B20A0` are usually rejected for Ethereum address post EIP-2.

Recommendation
Consider rejecting signatures with `s` values in the upper ranges, even though it may not be a security issue in this case.

5. Endpoint Information for Coordinators is not Enforced

The `setCoordinator()` function in the auction contract allows the URL parameter to be left unset. During the bidding process, the coordinator's address is checked for a zero value. However, no checks are performed that ensure the endpoint information is not left empty.

Recommendation
Check for a valid endpoint string.

6. Roles in WithdrawlDelayer can be set to address 0

The addresses of the DAO, white hack group, and Hermez keeper can be set to 0. This may be intentional in some cases, to allow for certain roles to be retired eventually. However, since this process is not reversible, it is recommended to use this facility with caution.

Recommendation
Verify if address 0 is intentional for each role.

7. Saving constants with the hash result is cheaper than saving the hashing operation

There are several places where hashing is used to identify operations. Pre-calculating these hashes and save constants could be used to save gas.

Recommendation
Consider re-calculating the hashes.

8. Deposit timestamps are only recorded for the last deposit

The `deposit()` function in `WithdrawalDelayer.sol` updates `depositTimestamp` to a new value every time the deposit is made.
If a user makes another `deposit()` before the previously deposited funds are withdrawn, the user would not be able to withdraw the previously deposited tokens until the `_withdrawalDelay` has passed after the second `deposit()`.

Recommendation
For simplicity and efficiency, it may be necessary to operate this way (overwriting the `depositTimestamp`), but users should be made aware that consecutive exits may result in withdrawal of the latest deposit.

Team Response
“We will make the users aware of that. We expect delayed withdrawals to be very rare.”

Audit Report for Hermez on October 18, 2020

Disclaimer

Solidified audit is not a security warranty, investment advice, or an endorsement of Iden3 or its products. This audit does not provide a security or correctness guarantee of the audited smart contract. Securing smart contracts is a multistep process, therefore, running a bug bounty program as a complement to this audit is strongly recommended.

The individual audit reports are anonymized and combined during a debrief process, in order to provide an unbiased delivery and protect the auditors of Solidified platform from legal and financial liability.

Solidified Technologies Inc.