Tornado - Verifier.sol Review

Review Resources:
A github repo containing protocol documentation and smart contracts was provided.

Project Auditor:

- Ahiara Ikechukwu Marvellous

Table of Contents

Review Summary

Tornado Cash

Tornado Cash is a non-custodial Ethereum and ERC20 privacy solution based on zkSNARKs. It improves transaction privacy by breaking the on-chain link between the recipient and destination addresses. It uses a smart contract that accepts ETH deposits that can be withdrawn by a different address. Whenever ETH is withdrawn by the new address, there is no way to link the withdrawal to the deposit, ensuring complete privacy.

The main branch of the tornado repo was reviewed, Verifier.sol was covered.

Scope

Code
Commit

The commit reviewed was `1ef6a263ac6a0e476d063fcb269a9df65a1bd56a`. The review covered the repository at the specific commit and focused on the contracts directory.

The review is a code review of smart contracts to identify potential vulnerabilities in the code.

Code Evaluation Matrix

Category	Mark	Description
Access Control	None	Access control wasn't applied in the contract.
Mathematics	Good	Solidity 0.7.0 is used, which doesn't provide overflow and underflow protection but no overflow or underflow vulnerabilities were found. No low-level bitwise operation was used.
Libraries	Good	No external library was used. An internal library, Proof, was used.
Complexity	Good	Zero-knowledge maths and cryptography were used in calculations. Inline assembly was also used. No proxy contracts or delegatecalls used.
Documentation	Poor	Comments don't exist for most functions and variables.
Monitoring	None	No events for core functions that modify state variables.
Testing	Good	All tests were passing and test coverage was expansive.

Findings Explanation

Findings are broken down into sections by their respective impact:

- Critical, High, Medium, Low impact
    - These are findings that range from attacks that may cause loss of funds, impact control/ownership of the contracts, or cause any unintended consequences/actions that are outside the scope of the requirements.
- Gas Savings
    - Findings that can improve the gas efficiency of the contracts
- Informational
    - Findings including recommendations and best practices

Informational Findings

1. **Solc version 0.7.0 not recommended for deployment.**

Recommendation

Use solc version 0.8.0-latest for overflow and underflow check, compiler bug fixes and contract bytecode optimizations

Final Remarks

Having reviewed the contracts, no critical vulnerabilities were found in verifier.sol. Inline assembly in combination with zero-knowledge maths was used in the contract and accurate/appropriate checks were implemented for various low-level calls.