Quantstamp

Trust Wallet Paymaster

Executive Summary

This audit report was prepared by Quantstamp, the leader in blockchain security.

Type	ERC4337 Paymaster
Timeline	2025-03-07 through 2025-03-19
Language	Solidity
Methods	Architecture Review, Unit Testing, Functional Testing, Computer-Aided Verification, Manual Review
Specification	None
Source Code	• https://github.com/trustwallet/paymaster #422cbae
Auditors	• Nikita Belenkov Senior Auditing Engineer • Hytham Farah Auditing Engineer • Ruben Koch Senior Auditing Engineer

Documentation quality	Undetermined
Test quality	Undetermined
Total Findings	6 Fixed: 2 Acknowledged: 4
High severity findings	0
Medium severity findings	0
Low severity findings	2 Fixed: 2
Undetermined severity findings	0
Informational findings	4 Acknowledged: 4

Summary of Findings

The TrustWallet Paymaster consists of two separate implementations for ERC-4337 EntryPoint v0.6 and v0.7, each providing standard token-based gas sponsorship functionality with signature verification and treasury management.

Both versions include a `PostOpGasEstimator` contract for simulating `postOp()` call behaviour. The main concern is that postOp reverts from failed token transfers could harm the paymaster's reputation under ERC-7562, particularly vulnerable to front-running attacks. Other minor issues include uninitialized treasury address, validation inconsistencies, and gas estimation inaccuracies in the PostOpGasEstimator.

Fix-Review-Update: All issues have either been reasonably acknowledged or properly fixed. The test suite has been also adapted to thoroughly test the `postOp()`-flow in the v0.6 implementation. This is a preliminary version of the final report, as the client has requested some feedback from the EF regarding nonce-behaviour in EntryPointv0.6. Once that is in, TW-PAY-5 will be revisited.

ID	DESCRIPTION	SEVERITY	STATUS
TW-PAY-1	Reverting Postop Could Harm Paymaster Reputation	Low	Fixed
TW-PAY-2	Paymaster Can Accidentally Cause The Whole Bundle To Revert Due to Bug in EntryPointV0.6	Low	Fixed
TW-PAY-3	Potential Validation Inconsistencies Leading to Paymaster Ban	Informational	Acknowledged
TW-PAY-4	Underestimation of Post-Op Gas in PostOpGasEstimator	Informational	Acknowledged
TW-PAY-5	Unclear Nonce Behavior in v0.6	Informational	Acknowledged
TW-PAY-6	No Association Between On-Chain Gas Costs and the User's ERC20-Payment	Informational	Acknowledged

Assessment Breakdown

Quantstamp's objective was to evaluate the repository for security-related issues, code quality, and adherence to specification and best practices.

> **Disclaimer**
> Only features that are contained within the repositories at the commit hashes specified on the front page of the report are within the scope of the audit and fix review. All features added in future revisions of the code are excluded from consideration in this report.

Possible issues we looked for included (but are not limited to):

- Transaction-ordering dependence
- Timestamp dependence
- Mishandled exceptions and call stack limits
- Unsafe external calls
- Integer overflow / underflow
- Number rounding errors
- Reentrancy and cross-function vulnerabilities
- Denial of service / logical oversights
- Access control
- Centralization of power
- Business logic contradicting the specification
- Code clones, functionality duplication
- Gas usage
- Arbitrary token minting

Methodology

1. Code review that includes the following
    1. Review of the specifications, sources, and instructions provided to Quantstamp to make sure we understand the size, scope, and functionality of the smart contract.
    2. Manual review of code, which is the process of reading source code line-by-line in an attempt to identify potential vulnerabilities.
    3. Comparison to specification, which is the process of checking whether the code does what the specifications, sources, and instructions provided to Quantstamp describe.
2. Testing and automated analysis that includes the following:
    1. Test coverage analysis, which is the process of determining whether the test cases are actually covering the code and how much code is exercised when we run those test cases.
    2. Symbolic execution, which is analyzing a program to determine what inputs cause each part of a program to execute.
3. Best practices review, which is a review of the smart contracts to improve efficiency, effectiveness, clarity, maintainability, security, and control based on the established industry and academic practices, recommendations, and research.
4. Specific, itemized, and actionable recommendations to help you take steps to secure your smart contracts.

Scope

We examined the main Paymaster contracts for a v0.6 and v0.7 EntryPoint.

Files Included

```
├── src/
    ├── v0.6/
    │   ├── Paymaster.sol
    │   ├── PostOpGasEstimator.sol
    │   └── interfaces/
    │       └── IPaymaster.sol
    │
    └── v0.7/
        ├── Paymaster.sol
        ├── PostOpGasEstimator.sol
        └── interfaces/
            └── IPaymaster.sol
```

Files Excluded

All other files in the repo

Operational Considerations

1. The paymaster system relies on ERC-4337's EntryPoint contract for validating and executing UserOperations. The EntryPoint's bundler network must remain functional to ensure sponsored transactions can be included in blocks, as centralization or bundler unavailability could prevent users from executing operations.
2. The token transfer mechanism in `_postOp()` assumes that token contracts follow standard ERC-20 behavior and don't have transfer fees or unusual behavior like rebasing.
3. The on-chain conditions remain unchanged for the UserOperation's `sender` to keep the approval state stable right before the invocation of the `postOp()` function between the off-chain simulation from the signer-API and the actual on-chain invocation as part of the bundle. Changes in the blockchain state between the signer simulating the UserOperation and its on-chain execution could cause the token transfer to revert on-chain. However, this primarily affects the paymaster's funds and reputation rather than user funds.
4. The paymaster's deposit balance in the EntryPoint must remain sufficient to cover the gas costs of sponsored operations.

Key Actors And Their Capabilities

Owner

Responsibility

- Manages critical paymaster parameters, including treasury address and signer address.
- Controls the verification process for sponsored transactions through signer management.
- Ensures proper configuration of token collection through treasury management.

Trust Assumptions

- Will not maliciously update the signer to one that signs harmful operations (though even malicious payment details + signatures from the signer API would still require the `UserOperation.sender` to submit a passing UserOperation validation, meaning that the user does directly confirm the paymaster's payment details).
- Will maintain sufficient paymaster deposit balance for user operations.
- Will not set malicious treasury that could disrupt token transfers and harm user operations.

Exclusive Functions

Paymaster (v0.6 and v0.7):

- `updateTreasury()`: Updates the address where collected tokens are sent.
- `updateSigner()`: Updates the address authorized to sign paymaster validations.

Signer

Responsibility

- External service that validates and signs UserOperations for gas sponsorship.
- Controls which operations get sponsored by the paymaster through signature verification.

Trust Assumptions

- No significant trust assumptions as the signer is controlled by TrustWallet and any malicious behavior would primarily harm their own service and reputation.

Exclusive Functions

No direct contract functions, but their signature is required for:

- Validating paymaster operations through `_validatePaymasterUserOp()`.
- Authorizing gas sponsorship for user operations.

User

Responsibility

- Assure to only sign UserOperations with paymaster-signer data that has a proper and fair paymaster reimbursement in ERC20 tokens. Technically, the signer data encodes an ERC20 token transfer that is not directly related to the on-chain gas consumed, so users need to check that the requested payment is appropriate.

Definitions

- **High severity** – High-severity issues usually put a large number of users' sensitive information at risk, or are reasonably likely to lead to catastrophic impact for client's reputation or serious financial implications for client and users.
- **Medium severity** – Medium-severity issues tend to put a subset of users' sensitive information at risk, would be detrimental for the client's reputation if exploited, or are reasonably likely to lead to moderate financial impact.

- **Low severity** – The risk is relatively small and could not be exploited on a recurring basis, or is a risk that the client has indicated is low impact in view of the client's business circumstances.
- **Informational** – The issue does not post an immediate risk, but is relevant to security best practices or Defence in Depth.
- **Undetermined** – The impact of the issue is uncertain.
- **Fixed** – Adjusted program implementation, requirements or constraints to eliminate the risk.
- **Mitigated** – Implemented actions to minimize the impact or likelihood of the risk.
- **Acknowledged** – The issue remains in the code but is a result of an intentional business or design decision. As such, it is supposed to be addressed outside the programmatic means, such as: 1) comments, documentation, README, FAQ; 2) business processes; 3) analyses showing that the issue shall have no negative consequences in practice (e.g., gas analysis, deployment settings).

Appendix

File Signatures

The following are the SHA-256 hashes of the reviewed files. A file with a different SHA-256 hash has been modified, intentionally or otherwise, after the security review. You are cautioned that a different SHA-256 hash could be (but is not necessarily) an indication of a changed condition or potential vulnerability that was not within the scope of the review.

Files

- `e60...635 ./src/v0.6/Paymaster.sol`
- `ffd...707 ./src/v0.6/PostOpGasEstimator.sol`
- `e91...8a8 ./src/v0.7/Paymaster.sol`
- `6a5...454 ./src/v0.7/PostOpGasEstimator.sol`

Tests

- `cfd...024 ./test/Setup.sol`
- `bc0...dd2 ./test/v0.6/Paymaster.t.sol`
- `f3d...1a6 ./test/v0.7/Paymaster.t.sol`

Test Suite Results

11 total tests all passing.

Update Fix-Review: Two tests have been added thoroughly testing the `postOp()` flow to match expected behaviour in the v0.6 implementation.

Code Coverage

Could not compile coverage report due to stack too deep errors.

Changelog

- 2025-03-19 - Initial report
- 2025-04-13 - Final report

About Quantstamp

Quantstamp is a global leader in blockchain security. Founded in 2017, Quantstamp's mission is to securely onboard the next billion users to Web3 through its best-in-class Web3 security products and services.

Quantstamp's team consists of cybersecurity experts hailing from globally recognized organizations including Microsoft, AWS, BMW, Meta, and the Ethereum Foundation. Quantstamp engineers hold PhDs or advanced computer science degrees, with decades of combined experience in formal verification, static analysis, blockchain audits, penetration testing, and original leading-edge research.

To date, Quantstamp has performed more than 500 audits and secured over $200 billion in digital asset risk from hackers. Quantstamp has worked with a diverse range of customers, including startups, category leaders and financial institutions. Brands that Quantstamp has worked with include Ethereum 2.0, Binance, Visa, PayPal, Polygon, Avalanche, Curve, Solana, Compound, Lido, MakerDAO, Arbitrum, OpenSea and the World Economic Forum.

Quantstamp's collaborations and partnerships showcase our commitment to world-class research, development and security. We're honored to work with some of the top names in the industry and proud to secure the future of web3.

Notable Collaborations & Customers:

- Blockchains: Ethereum 2.0, Near, Flow, Avalanche, Solana, Cardano, Binance Smart Chain, Hedera Hashgraph, Tezos
- DeFi: Curve, Compound, Maker, Lido, Polygon, Arbitrum, SushiSwap
- NFT: OpenSea, Parallel, Dapper Labs, Decentraland, Sandbox, Axie Infinity, Illuvium, NBA Top Shot, Zora
- Academic institutions: National University of Singapore, MIT

Timeliness of content

The content contained in the report is current as of the date appearing on the report and is subject to change without notice, unless indicated otherwise by Quantstamp; however, Quantstamp does not guarantee or warrant the accuracy, timeliness, or completeness of any report you access using the internet or other means, and assumes no obligation to update any information following publication or other making available of the report to you by Quantstamp.

Notice of confidentiality

This report, including the content, data, and underlying methodologies, are subject to the confidentiality and feedback provisions in your agreement with Quantstamp. These materials are not to be disclosed, extracted, copied, or distributed except to the extent expressly authorized by Quantstamp.

Links to other websites

You may, through hypertext or other computer links, gain access to web sites operated by persons other than Quantstamp. Such hyperlinks are provided for your reference and convenience only, and are the exclusive responsibility of such web sites' owners. You agree that Quantstamp are not responsible for the content or operation of such web sites, and that Quantstamp shall have no liability to you or any other person or entity for the use of third-party web sites. Except as described below, a hyperlink from this web site to another web site does not imply or mean that Quantstamp endorses the content on that web site or the operator or operations of that site. You are solely responsible for determining the extent to which you may use any content at any other web sites to which you link from the report. Quantstamp assumes no responsibility for the use of third-party software on any website and shall have no liability whatsoever to any person or entity for the accuracy or completeness of any output generated by such software.

Disclaimer

The review and this report are provided on an as-is, where-is, and as-available basis. To the fullest extent permitted by law, Quantstamp disclaims all warranties, expressed implied, in connection with this report, its content, and the related services and products and your use thereof, including, without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement. You agree that access and/or use of the report and other results of the review, including but not limited to any associated services, products, protocols, platforms, content, and materials, will be at your sole risk. FOR AVOIDANCE OF DOUBT, THE REPORT, ITS CONTENT, ACCESS, AND/OR USAGE THEREOF, INCLUDING ANY ASSOCIATED SERVICES OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, INVESTMENT, TAX, LEGAL, REGULATORY, OR OTHER ADVICE. This report is based on the scope of materials and documentation provided for a limited review at the time provided. You acknowledge that Blockchain technology remains under development and is subject to unknown risks and flaws and, as such, the report may not be complete or inclusive of all vulnerabilities. The review is limited to the materials identified in the report and does not extend to the compiler layer, or any other areas beyond the programming language, or programming aspects that could present security risks. The report does not indicate the endorsement by Quantstamp of any particular project or team, nor guarantee its security, and may not be represented as such. No third party is entitled to rely on the report in any way, including for the purpose of making any decisions to buy or sell a product, service or any other asset. Quantstamp does not warrant, endorse, guarantee, or assume responsibility for any product or service advertised or offered by a third party, or any open source or third-party software, code, libraries, materials, or information to, called by, referenced by or accessible through the report, its content, or any related services and products, any hyperlinked websites, or any other websites or mobile applications, and we will not be a party to or in any way be responsible for monitoring any transaction between you and any third party. As with the purchase or use of a product or service through any medium or in any environment, you should use your best judgment and exercise caution where appropriate.