



Executive Summary

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

Type	ERC20	Documentation quality	High	<div><div></div></div>
Timeline	2025-06-09 through 2025-06-09	Test quality	High	<div><div></div></div>
Language	Solidity	Total Findings	2	<div><div></div><div>Fixed: 2</div></div>
Methods	Architecture Review, Unit Testing, Functional Testing, Computer-Aided Verification, Manual Review	High severity findings ⓘ	0	
Specification	None	Medium severity findings ⓘ	0	
Source Code	<ul style="list-style-type: none">openledger #7eb2e0c79e1358d94ed563fd2eb03444d2955f1e 	Low severity findings ⓘ	0	
Auditors	<ul style="list-style-type: none">Hytham Farah Auditing EngineerHamed Mohammadi Auditing Engineer	Undetermined severity findings ⓘ	0	
		Informational findings ⓘ	2	<div><div></div><div>Fixed: 2</div></div>

Summary of Findings

The Open contract is a straightforward ERC20 token implementation built on OpenZeppelin's ERC20 and ERC20Burnable contracts. It creates a token and includes burning functionality that allows token holders to destroy their tokens, reducing the total supply. The contract is constructed with a fixed supply of 1 billion tokens (with 18 decimals) that are minted to the address specified during deployment. The OpenLedger team intends for this to be the custom gas token of an L2 chain, with Layer0 integration to allow the token for cross-chain bridging; however, both are out of scope for the current audit.

Despite minor documentation erros regarding the visibility of the `NAME` and `SYMBOL` constants (**OPEN-1**) and the listed security contact (**OPEN-2**), the implementation follows standard ERC20 practices with well-documented functions and includes a security contact for vulnerability reporting. The contract's simplicity and comprehensive test coverage (100% across all metrics) demonstrate a robust, focused design that handles core token functionality.

UPDATE: All issues were fully resolved following the recommendations.

ID	DESCRIPTION	SEVERITY	STATUS
OPEN-1	Documentation Inconsistency in Name and Symbol Constants	<ul style="list-style-type: none">Informational ⓘ	Fixed
OPEN-2	Placeholder Security Contact in Natspec	<ul style="list-style-type: none">Informational ⓘ	Fixed

Assessment Breakdown

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

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

The entire scope consists of only the `Open.sol` contract.

Files Included

contracts

└─ `Open.sol`

Operational Considerations

- The total token supply is fixed at deployment to 1 billion `OPEN`, minted directly to a specified `recipient` address. There are no additional minting functions beyond the constructor, implying a non-inflationary token once deployed.
- The token inherits from `ERC20Burnable`, enabling holders to destroy their own tokens voluntarily. This burn capability is irreversible and reduces the circulating supply permanently.
- The contract uses OpenZeppelin's `ERC20` and `ERC20Burnable` implementations from version `0.8.27`, assuming these libraries are unmodified and behave as per the published standard. If any local changes have been made to these libraries, they are out of scope and must be reviewed separately.
- Token name and symbol are defined as internal constants (`NAME`, `SYMBOL`) and passed to the `ERC20` constructor, which exposes them via the standard getter functions. This assumes no need for future changes to these identifiers.
- The deployment environment must provide a valid `recipient` address to the constructor. Misconfiguration here could result in tokens being minted to an inaccessible or incorrect address, which is not reversible.

Key Actors And Their Capabilities

Recipient (Deployer-Assigned)

Responsibility

- Receives the full initial supply of 1 billion `OPEN` tokens upon deployment.

- Effectively controls the distribution and early utility of the token.

Trust Assumptions

- Will manage the initial supply responsibly.

Exclusive Functions

1. `constructor()` :
 - Mints 1,000,000,000 `OPEN` tokens to the specified `recipient` .
 - This is a one-time action occurring during deployment.

Findings

OPEN-1

Documentation Inconsistency in Name and Symbol Constants

• Informational ⓘ Fixed



Update

Marked as "Fixed" by the client.
Addressed in: `f8af7c0252139cec286b5642ace5048704f23d50` .
The client provided the following explanation:

Changed Name and Symbol natspec comments

File(s) affected: `Open`

Description: The documentation for the `NAME` and `SYMBOL` constants contains misleading information. The comments state that "Field is declared public: `getter name()/symbol()` is created when compiled," which incorrectly suggests that these constants are declared as `public` state variables that automatically generate getter functions. However, these constants are actually declared as `internal` , and the getter functions (`name()` and `symbol()`) are inherited from the OpenZeppelin ERC20 implementation rather than being auto-generated.

Recommendation: Update the documentation to accurately reflect the implementation by modifying the comments for both constants.

OPEN-2 Placeholder Security Contact in Natspec

• Informational ⓘ Fixed



Update

Marked as "Fixed" by the client.
Addressed in: `749abe4a9dacf62209f1ee1ec944274b26c72aac` .
The client provided the following explanation:

Changed Security contact

File(s) affected: `Open`

Description: The smart contract `Open.sol` includes a placeholder email (`security@Open.example.com`) in the NatSpec comments. The `.example.com` domain is reserved for documentation and not intended for production use.

Recommendation: Replace `security@Open.example.com` with a valid, actively monitored email address before deployment.

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).

Test Suite Results

12 total tests, each of them passing, ensuring transfer, burn, and deployment work as expected.

Open

Deployment

✓ Should set the right name **and** symbol

✓ Should assign the total supply of **tokens** to the owner

✓ Should have correct total supply

Transactions

✓ Should transfer **tokens** between accounts

✓ Should fail **if** sender doesn't have enough **tokens**

✓ Should update allowances on approval

TransferFrom

✓ Should transfer **tokens** using transferFrom

✓ Should fail **if** trying to transferFrom more than allowed

Burning

✓ Should **burn tokens and** reduce total supply

✓ Should allow users to **burn** their own **tokens**

✓ Should fail **if** trying to **burn** more **tokens** than owned

✓ Should emit Transfer event on **burn**

12 passing (722ms)

Code Coverage

Outstanding code coverage with 100 percent coverage across all metrics.

File	% Stmts	% Branch	% Funcs	% Lines
contracts/	100	100	100	100
Open.sol	100	100	100	100
All files	100	100	100	100

Changelog

- 2025-06-08 - Initial report
- 2025-06-10 - 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.

