



Executive Summary

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

Type	Token	Documentation quality	Medium
Timeline	2025-04-17 through 2025-04-18	Test quality	High
Language	Solidity	Total Findings	3 Fixed: 2 Acknowledged: 1
Methods	Architecture Review, Unit Testing, Functional Testing, Computer-Aided Verification, Manual Review	High severity findings ⓘ	0
Specification	None	Medium severity findings ⓘ	1 Acknowledged: 1
Source Code	<ul style="list-style-type: none">https://github.com/hemilabs/hemi-token #8e3988b 	Low severity findings ⓘ	2 Fixed: 2
Auditors	<ul style="list-style-type: none">Hytham Farah Auditing EngineerLeonardo Passos Senior Research Engineer	Undetermined severity findings ⓘ	0
		Informational findings ⓘ	0

Summary of Findings

The Hemi token (HEMI) is a sophisticated ERC20 implementation designed with a controlled inflation mechanism and cross-layer interoperability. It inherits from OpenZeppelin's ERC20, ERC20Permit, ERC20Votes, and Ownable contracts, providing standard token functionality alongside governance capabilities and gasless approvals. The token launches with an initial supply of 10 billion tokens and features a configurable annual inflation rate that can only be reduced, never increased.

What makes Hemi particularly distinctive is its automated bridging mechanism. When new tokens are minted through the periodic emission process (which can only occur once every 30 days), they are automatically transferred to a Layer 2 solution via a configurable L2 tunnel.

Overall, the code is well written and sound. The most significant issue found involves the emission mechanism (**HEMI-1**), which is susceptible to compounding effects due to its reliance on total supply calculations. The rest are issues to do with renounceable ownership and recommendations for additional validation on key protocol parameters.

FIX-REVIEW UPDATE: In the latest updates, vulnerabilities **HEMI-2** and **HEMI-3** have been successfully fixed, along with both suggestions S1 and S2, indicating proactive remediation of identified issues. **HEMI-1** remains acknowledged, as the client asserts that compounding is an intentional business design decision.

ID	DESCRIPTION	SEVERITY	STATUS
HEMI-1	Emissions Vary with Minting Frequency Due to Compounding	● Medium ⓘ	Acknowledged
HEMI-2	Incorrect Ownership Renouncement May Block Crucial Functions	● Low ⓘ	Fixed
HEMI-3	Zero Inflation Rate at Deployment Time Becomes Immutable	● Low ⓘ	Fixed

Assessment Breakdown

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

i 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

The bridging mechanism is outside of scope. The audited contract deals only with token creation, configuration and inflation. Tokens sent to the bridge are assumed to operate as intended in this scope.

Files Included

Repo: [https://github.com/hemilabs/hemi-token\(8e3988b6ad1372ce0306161bd6418bb8615dc645\)](https://github.com/hemilabs/hemi-token(8e3988b6ad1372ce0306161bd6418bb8615dc645))

Files:

- src/Hemi.sol

Operational Considerations

- The emissions mechanism relies on the manual invocation of `mintEmissions()` and assumes that calls are made only after `MINTAGE_PERIOD` (30 days) has passed since `lastEmission`. Failure to call the function on schedule will not halt operations but may delay emissions and result in time-proportional adjustments to future minting amounts.
- Emissions are bridged to an L2 network using a configured `IL2Tunnel` interface. This bridging assumes that the `l2Tunnel`, `remoteToken`, and `l2Destination` are correctly configured and compatible with the bridging mechanism. Misconfiguration or changes to the L2 tunnel logic may cause emission transfers to fail or revert.
- Emission minting to the L2 is executed via `bridgeERC20To()` with a static `l2TunnelMinGasLimit` of 400,000 gas. If this gas limit is insufficient due to L2 cost changes or implementation differences, emissions may be delayed or blocked.
- The system assumes one-time setup of emission parameters (`setupEmissions()`) prior to enabling emissions. Once `enableEmissions()` is called, the emission configuration becomes immutable, and incorrect setup cannot be amended on-chain.
- The design permits a one-directional inflation cut through `updateInflationRate()`, assuming that protocol governance will exercise this option prior to calling `disableInflationCut()`, after which further reductions become permanently inaccessible.
- Initial supply is minted once at deployment to `_initialMintReceiver` (10 billion tokens), with all subsequent supply growth governed exclusively by the inflation function and emission logic. No additional minting pathways exist outside of emissions.

Key Actors And Their Capabilities

Owner

Responsibilities

The `owner` holds limited administrative rights focused on emission configuration and inflation rate management. They are responsible for:

- Setting up the L2 tunnel configuration needed for bridging emissions.
- Enabling the start of emissions (one-time initialization).
- Reducing the token’s annual inflation rate, as long as inflation cuts are permitted.
- Disabling further inflation rate cuts, making the inflation policy permanent.

Trust Assumptions

- Will set up L2 tunnel parameters correctly before emissions begin.
 - Will not delay or block emissions arbitrarily after setup.
 - Will lower inflation rates judiciously if allowed — they **cannot** increase it.
 - Will not disable inflation cuts maliciously unless the inflation rate is already acceptable.
-
- **Important Boundary:**
 - The `owner` cannot mint tokens arbitrarily — emissions can only occur based on a deterministic formula tied to the elapsed time and total supply. This is a **trust-minimized mechanism**.
 - Once emissions are enabled, the `owner` cannot update the tunnel configuration or restart the emission timer. This ensures **immutability and predictability** post-initialization.

Exclusive Functions

1. Hemi:

1. `enableEmissions()` :
Starts the emission process by setting `lastEmission` and optionally minting an initial emission. Can only be called once and only after tunnel setup.
2. `updateInflationRate()` :
Reduces the `annualInflationRate` . The function enforces a one-way change — **only decreases** are allowed. This right is further gated by the `allowInflationCut` flag.
3. `disableInflationCut()` :
Permanently revokes the ability to reduce the inflation rate in the future, locking in the current value.
4. `setupEmissions()` :
Configures the bridging layer by setting `l2Tunnel` , `l2Destination` , and `remoteToken` . This can be updated multiple times but only before emissions are enabled.

InitialMintReceiver

Responsibilities

Receives the full initial supply of ten billion HEMI tokens at deployment. These tokens are not bridged and are immediately transferable. The contract grants this address economic power over the token’s early distribution, liquidity bootstrapping, or strategic allocation.

Trust Assumptions

- Will not dump tokens or use them in a way that undermines the protocol’s credibility or price stability.
- Will ensure tokens are allocated as stated.

Exclusive Functions

- None — this actor does not have any on-chain permissions, but their economic influence is derived from the deployment constructor.

Findings

HEMI-1

Emissions Vary with Minting Frequency Due to Compounding

• **Medium** ⓘ **Acknowledged**

Update

Marked as "Acknowledged" by the client.
The client provided the following explanation:

It is a business requirement to calculate inflation on current total supply hence the compounding effect is not an issue.

File(s) affected: `src/Hemi.sol`

Description: Emissions are calculated using `totalSupply` , which includes previously minted tokens. This introduces compounding behavior—each emission increases the base for future ones. The more frequently `mintEmissions()` is called, the more total tokens are minted over time, even if the total duration and inflation rate remain the same.

This means token inflation will depend on the frequency of the `mintEmissions()` and not only on the `anualInflationRate` variable, where call timing can affect long-term token supply.

Exploit Scenario:

Assume:

- Initial supply: 1,000 tokens
- Annual inflation: 20%
- Total duration: 2 years

Case 1: One emission after 2 years

`1000 * (1 + 0.20 * 2) = 1400 tokens`

Case 2: One emission per year (2 total)

`1000 * (1.2 ** 2) = 1440 tokens`

Case 3: One emission every 6 months (4 total)

`1000 * (1.1 ** 4) = 1464.1 tokens`

A user (or system) calling `mintEmissions()` more frequently than expected can extract extra inflation without any increase in elapsed time or inflation rate.

Recommendation: Ensure that the annual inflation rate is consistent and does not depend on the frequency of calls to `mintEmissions()`.

HEMI-2

Incorrect Ownership Renouncement May Block Crucial Functions

• **Low** ⓘ **Fixed**



Update

Marked as "Fixed" by the client.
Addressed in: `f267afb850451fa1ebeede4632af296b822bfa0d` .

File(s) affected: `src/Hemi.sol`

Description: If the `Hemi` token contract `owner` incorrectly renounces his ownership, functions with the `onlyOwner` modifier won't be able to be executed, blocking crutial functions (e.g., `enableEmissions`). This stems from the fact that ownership is performed as a single step.

Recommendation: Consider overriding the `renounceOwnership()` function to disable it. Alternatively, make ownership transfer a two-step process (See S2).

HEMI-3 Zero Inflation Rate at Deployment Time Becomes Immutable

• **Low** ⓘ **Fixed**



Update

Marked as "Fixed" by the client.
Addressed in: `f267afb850451fa1ebeede4632af296b822bfa0d` .

File(s) affected: `src/Hemi.sol`

Description: The construtor of the `Hemi` token contract does not enforce that the inflation rate parameter `_annualInflationRate` is greater than zero. If zero is passed as an argument, the owner won't be able to later adjust it to a different value, as inflation only decreases with time or stays constant.

Recommendation: Unless it is for a valid use case, validate that the initial value of the inflation rate is greater than zero.

Auditor Suggestions

S1 Unlocked Pragma

Fixed



Update

Marked as "Fixed" by the client.
Addressed in: `f267afb850451fa1ebeede4632af296b822bfa0d` .

File(s) affected: `src/Hemi.sol`

Related Issue(s): [SWC-103](#)

Description: Every Solidity file specifies in the header a version number of the format `pragma solidity (^)0.8.*`. The caret (`^`) before the version number implies an unlocked pragma, meaning that the compiler will use the specified version *and above*, hence the term "unlocked".

Recommendation: For consistency and to prevent unexpected behavior in the future, we recommend to remove the caret to lock the file onto a specific Solidity version.

S2 Critical Role Transfer Not Following Two-Step Pattern

Fixed



Update

Marked as "Fixed" by the client.
Addressed in: `f267afb850451fa1ebeede4632af296b822bfa0d`.

File(s) affected: `src/Hemi.sol`

Description: The owner of the contracts can call `transferOwnership()` to transfer the ownership to a new address. If an uncontrollable address is accidentally provided as the new owner address then the contract will no longer have an active owner, and functions with the `onlyOwner` modifier can no longer be executed.

Recommendation: Consider using OpenZeppelin's `Ownable2Step` contract to adopt a two-step ownership pattern in which the new owner must accept their position before the transfer is complete.

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

Overall solid test suite covering all the main functionality of the audited contract.

```
Ran 12 tests for test/Hemi.t.sol:HemiTest
[PASS] testCalculateEmission() (gas: 109211)
[PASS] testDisableAllowInflationCut() (gas: 16034)
[PASS] testEmission() (gas: 383136)
[PASS] testEnableEmissions() (gas: 100093)
[PASS] testFirstEmissionAmount() (gas: 363343)
[PASS] testInflationRateReduced() (gas: 23979)
[PASS] testRevertIfEmissionNotSetup() (gas: 18322)
[PASS] testRevertIfEmissionsAlreadyEnabled() (gas: 100789)
[PASS] testRevertIfInvalidAnnualInflationRate() (gas: 19487)
[PASS] testRevertIfMintagePeriodNotElapsed() (gas: 100920)
[PASS] testRevertIfNullAddressInSetup() (gas: 31059)
[PASS] testSetupEmissions() (gas: 79348)
Suite result: ok. 12 passed; 0 failed; 0 skipped; finished in 3.06s (4.46s CPU time)
```


Code Coverage

Overall coverage is solid achieving 90% in most metrics. We recommend improving branch coverage to at least 90%.

UPDATE: Coverage is slightly lower after the fix review, as more code was added but no new tests were added.

File	% Lines	% Statements	% Branches	% Funcs
src/Hemi.sol	89.83% (53/59)	89.47% (51/57)	63.64% (7/11)	90.00% (9/10)

Changelog

- 2025-04-18 - Initial report
- 2025-04-23 - 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.

