



Executive Summary

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

Type	Token Vesting	Documentation quality	Medium	<div><div></div></div>
Timeline	2025-03-24 through 2025-03-28	Test quality	Medium	<div><div></div></div>
Language	FunC	Total Findings	2	<div><div></div></div> Acknowledged: 2
Methods	Architecture Review, Unit Testing, Functional Testing, Computer-Aided Verification, Manual Review	High severity findings ⓘ	0	
Specification	README.md	Medium severity findings ⓘ	0	
Source Code	<ul style="list-style-type: none">https://gitlab.com/bemolabs/bemo/smart-contracts/vesting #446e919 	Low severity findings ⓘ	2	<div><div></div></div> Acknowledged: 2
Auditors	<ul style="list-style-type: none">Jonathan Mevs Auditing EngineerMichael Boyle Auditing Engineer	Undetermined severity findings ⓘ	0	
		Informational findings ⓘ	0	

Summary of Findings

Bemo is liquid staking protocol on the TON network. In this audit, we reviewed their token vesting contract. Vesting contracts are deployed for each user, where tokens can be claimed at a periodic rate throughout the vesting period, with a defined amount of tokens becoming immediately claimable after a cliff date. The Bemo admin has the ability to configure the vesting terms for each jetton beneficiary. The terms include total jetton tokens vested, the cliff date, the cliff unlock amount, the vesting period and distribution frequency. Once vesting has completed, the admin maintains the ability to send arbitrary service messages through this contract.

During the audit we identified two low severity issues. The issue regarding input validation could totally invalidate a deployed vesting contract, causing funds to be wasted on redeployment. The other issue suggests using timestamps with more bits of storage to future proof the contracts beyond 2038. Additionally, we note operational considerations and describe the key actors and their capabilities. The test suite is complete and all tests pass, covering "happy" and "unhappy" paths.

Fix-Review Update 2025-03-31:

After reviewing the client's response to our findings, it is clear that they have thoroughly explained their current approach. While some enhancements are not implemented immediately, the client has acknowledged the issues and outlined plans for future improvements.

ID	DESCRIPTION	SEVERITY	STATUS
BEMO-1	Missing Input Validation	<ul style="list-style-type: none">Low ⓘ	Acknowledged
BEMO-2	Using 32-Bit Unsigned Integer for Timestamp	<ul style="list-style-type: none">Low ⓘ	Acknowledged

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

Files Included

Repo: [https://gitlab.com/bemolabs/bemo/smart-contracts/vesting\(446e919239a3a56ab17b19a3ddea38bc96a1e7d3\)](https://gitlab.com/bemolabs/bemo/smart-contracts/vesting(446e919239a3a56ab17b19a3ddea38bc96a1e7d3)) Files: `contracts/vesting.fc`

Files Excluded

Repo: [https://gitlab.com/bemolabs/bemo/smart-contracts/vesting\(446e919239a3a56ab17b19a3ddea38bc96a1e7d3\)](https://gitlab.com/bemolabs/bemo/smart-contracts/vesting(446e919239a3a56ab17b19a3ddea38bc96a1e7d3)) Files: `contracts/imports/stdlib.fc`

Operational Considerations

- The Vesting admin is expected to deploy the user's Vesting contract with an accurate `admin_address` (themselves) and `claimer_address` (the vesting user). Further, the jetton wallet address where the `OP::JETTON_TRANSFER` message will be sent, needs to be accurately assigned.
- The Bemo Vesting Admin's Jetton Wallet address is expected to maintain enough jettons to honor what will be claimed by the claimer.

Key Actors And Their Capabilities

Admin

- The Vesting Admin first deploys the Vesting contract assigning their address and the claimer's address.
- Prior to the initialization step, the claimer can take no actions on this contract. During the initialization the admin specifies the following:
 - `jetton_balance` : the total amount of jettons vested to the claimer
 - `cliff_end_date`
 - `cliff_numerator` and `cliff_denominator` : specifying the portion of total jettons claimable after the cliff period ends
 - `vesting_period`
 - `distribution_frequency` : the frequency in which jettons are distributed beyond the cliff, through the end of the vesting period.
- Once vesting has completed, the admin maintains the ability to send arbitrary service messages through this contract.

Claimer

- Claiming Vesting Tokens: Authorized to invoke the CLAIM_JETTONS operation to unlock and transfer vested tokens.
- Access Control: The contract strictly verifies that only the designated claimer can claim tokens.

Jetton Wallet

- Holds the tokens and acts as the intermediary for token transfers. It is set during initialization and then used to execute jetton transfers upon claim requests.

Findings

BEMO-1 Missing Input Validation

• Low ⓘ Acknowledged

i Update

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

As far as bemo (admin) is concerned with deploying vesting contracts, issues with incorrect parameters during initialization won't arise

File(s) affected: contacts/vesting.fc

Description: The following locations in the code can benefit from further input validation. Invalid assignments during initialization could render the contract useless, causing funds to be wasted on redeployment.

- Ensure denominator > 0 to avoid division by zero.
- Ensure cliff_numerator < cliff_denominator to guarantee less than 100% unlock at the cliff.
- Verify that distribution_frequency > 0 .
- Verify that vesting_period ≥ distribution_frequency .
- Optionally, check if vesting_period is a multiple of distribution_frequency (i.e. vesting_period % distribution_frequency == 0).
- Confirm a non-zero token balance.
- Validate that the computed cliff unlock amount does not exceed the total token balance.
- Ensure that admin, claimer, and jetton wallet addresses are well-formed and non-empty.

Recommendation: Consider implementing the validation above.

BEMO-2 Using 32-Bit Unsigned Integer for Timestamp

• Low ⓘ Acknowledged

i Update

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

This contract won't be in use after 2038, so it won't face any overflow issues

File(s) affected: contacts/vesting.fc

Description: Due to using a 32-bit unsigned integer for timestamps, the timestamp will overflow at 03:14:07 UTC on January 19, 2038, preventing the vesting contract from correctly handling dates and rendering it unusable thereafter.

Recommendation: Consider using a larger integer to store timestamps.

Auditor Suggestions

S1 Code Improvements

Acknowledged

i Update

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

We decided not to fix this because the gas costs are little and it won't affect the contract's functionality

File(s) affected: `vesting.fc`

Description:

```
if (now() >= cliff_end_date + vesting_period) {  
    return jetton_balance - jettons_claimed;  
}
```

This check can be moved to the start of the `claimable_jettons()` function to save gas on other calculations, and immediately give the user the remainder of their vested tokens.

Recommendation: Consider moving that check to the start of the function.

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

- `535...47e ./contracts/vesting.fc`
- `ba1...e43 ./contracts/imports/stdlib.fc`

Test Suite Results

The test suite passes and covers "happy" and "unhappy" paths. As a further improvement on the test suite, we recommend including a test that verifies the functionality of processing a bounced jetton transfer message.

```
npm test
```

```
> vesting@0.0.1 test  
> jest --verbose
```

```
PASS  tests/Vesting.spec.ts
```

```
  Vesting
```

- ✓ should deploy (15 ms)
- ✓ should accept claim only from claimer (11 ms)
- ✓ should not process claim if there is nothing to claim (13 ms)
- ✓ should not process claim if there is not enough funds (25 ms)
- ✓ should claim entire jettons after 1 year (37 ms)
- ✓ should unlock cliff jettons after cliff period (40 ms)

- ✓ should accept service message only from admin (9ms)
- ✓ should not process service message if lockup period not finished (9 ms)
- ✓ should service message (34 ms)

Test Suites: 1 passed, 1 total
Tests: 9 passed, 9 total
Snapshots: 0 total
Time: 2.121 s, estimated 4 s
Ran all test suites.

Changelog

- 2025-03-28 - Initial report
- 2025-03-31 - 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 and may not be represented as such. No third party is entitled to rely on the report in any any way, including for the purpose of making any decisions to buy or sell a product, 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 or any open source or third-party software, code, libraries, materials, or information to, to, called by, referenced by or accessible through the report, its content, or any related 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 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.

