DuckChain Token Audit Report

Mon Jan 06 2025

contact@bitslab.xyz https://twitter.com/tonbit_

TonBit

DuckChain Token Audit Report

1 Executive Summary

1.1 Project Information

Description	It is a jetton token deployed on Ton, and the address is EQDWXjnVWheFemaAaFn-Cp4nDehvGllrXOZ8wqHm8sDEwn_c.
Type	Token
Auditors	TonBit
Timeline	Fri Jan 03 2025 - Fri Jan 03 2025
Languages	FunC
Platform	Ton
Methods	Architecture Review, Unit Testing, Manual Review
Source Code	https://github.com/movebit/ClientProjects
Commits	916280e92d21f7f977a2b3d355ab797f9785bd2e

1.2 Files in Scope

The following are the SHA1 hashes of the original reviewed files.

ID	File	SHA-1 Hash
PAR	DuckChain-Token/imports/params.fc	a59dfa4d790f6321439d0b08b37fc1d321f673fc
STD	DuckChain-Token/imports/stdlib.fc	5f1912d9dcde12b3c2dcb220cccc334af3fd5030
OCO	DuckChain-Token/imports/op-codes.fc	e445a071e96ccec60392012720cd05a586958089
DPA	DuckChain-Token/imports/discovery-params.fc	d829753856156b5ad1369a1aad2e8e282195eac7
UTI	DuckChain-Token/imports/utils.fc	7d54ceaeb7c32f925f3bb29fade4a52ab724842a
CON	DuckChain-Token/imports/constants.fc	04de1fdf8749c43cedcbb156df3c166bc2232ec5
JUT	DuckChain-Token/imports/jetton-utils.fc	97901223b30620b82c42ff1a14cb7ad0a09e98f7
JMI	DuckChain-Token/jetton-minter.fc	b3f238d76e6a399d7f334b588522ab78c44634cb

1.3 Issue Statistic

Item	Count	Fixed	Acknowledged
Total	1	1	0
Informational	0	0	0
Minor	0	0	0
Medium	0	0	0
Major	1	1	0
Critical	0	0	0

1.4 TonBit Audit Breakdown

TonBit aims to assess repositories for security-related issues, code quality, and compliance with specifications and best practices. Possible issues our team looked for included (but are not limited to):

- Transaction-ordering dependence
- Timestamp dependence
- Integer overflow/underflow by bit operations
- Number of rounding errors
- Denial of service / logical oversights
- Access control
- Centralization of power
- Business logic contradicting the specification
- Code clones, functionality duplication
- Gas usage
- Arbitrary token minting
- Unchecked CALL Return Values

1.5 Methodology

The security team adopted the **"Testing and Automated Analysis"**, **"Code Review"** strategy to perform a complete security test on the code in a way that is closest to the real attack. The main entrance and scope of security testing are stated in the conventions in the "Audit Objective", which can expand to contexts beyond the scope according to the actual testing needs. The main types of this security audit include:

(1) Testing and Automated Analysis

Items to check: state consistency / failure rollback / unit testing / value overflows / parameter verification / unhandled errors / boundary checking / coding specifications.

(2) Code Review

The code scope is illustrated in section 1.2.

(3) Audit Process

- Carry out relevant security tests on the testnet or the mainnet;
- If there are any questions during the audit process, communicate with the code owner in time. The code owners should actively cooperate (this might include providing the latest stable source code, relevant deployment scripts or methods, transaction signature scripts, exchange docking schemes, etc.);
- The necessary information during the audit process will be well documented for both the audit team and the code owner in a timely manner.

2 Summary

This report has been commissioned by DuckChain to identify any potential issues and vulnerabilities in the source code of the DuckChain Token smart contract, as well as any contract dependencies that were not part of an officially recognized library. In this audit, we have utilized various techniques, including manual code review and static analysis, to identify potential vulnerabilities and security issues.

During the audit, we identified 1 issues of varying severity, listed below.

ID	Title	Severity	Status
JMI-1	Centralization Risk	Major	Fixed

3 Participant Process

Here are the relevant actors with their respective abilities within the DuckChain Token Smart Contract :

Admin

- The Admin can mint any amount of tokens to an address through sending `op::mint()` .
- The Admin can set a new admin through sending op code `3` .

User

- The User can burn thier owned tokens through sending `op::burn_notification()` .

4 Findings

JMI-1 Centralization Risk

Severity: Major

Status: Fixed

Code Location:

DuckChain-Token/jetton-minter.fc#70

Descriptions:

Centralization risk was identified in the smart contract:

The admin can mint any amount of token to an address through sending the `op::mint()` .The administrator's arbitrary minting of tokens may lead to a crisis of trust, token depreciation, security vulnerabilities and legal risks.

Suggestion:

It is recommended that measures be taken to reduce the risk of centralization, such as a multi-signature mechanism.

Resolution:

The client has revoked the admin privilege, and the tx is https://tonviewer.com/transaction/e4acc3b9e3fbf05438265ba093b70fb0bd922b06c15e324dc5e54e

Appendix 1

Issue Level

- **Informational** issues are often recommendations to improve the style of the code or to optimize code that does not affect the overall functionality.
- **Minor** issues are general suggestions relevant to best practices and readability. They don't post any direct risk. Developers are encouraged to fix them.
- **Medium** issues are non-exploitable problems and not security vulnerabilities. They should be fixed unless there is a specific reason not to.
- **Major** issues are security vulnerabilities. They put a portion of users' sensitive information at risk, and often are not directly exploitable. All major issues should be fixed.
- **Critical** issues are directly exploitable security vulnerabilities. They put users' sensitive information at risk. All critical issues should be fixed.

Issue Status

- **Fixed:** The issue has been resolved.
- **Partially Fixed:** The issue has been partially resolved.
- **Acknowledged:** The issue has been acknowledged by the code owner, and the code owner confirms it's as designed, and decides to keep it.

Appendix 2

Disclaimer

This report is based on the scope of materials and documents provided, with a limited review at the time provided. Results may not be complete and do not include all vulnerabilities. The review and this report are provided on an as-is, where-is, and as-available basis. You agree that your access and/or use, including but not limited to any associated services, products, protocols, platforms, content, and materials, will be at your own risk. A report does not imply an endorsement of any particular project or team, nor does it guarantee its security. These reports should not be relied upon in any way by any third party, including for the purpose of making any decision to buy or sell products, services, or any other assets. TO THE FULLEST EXTENT PERMITTED BY LAW, WE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, IN CONNECTION WITH THIS REPORT, ITS CONTENT, RELATED SERVICES AND PRODUCTS, AND YOUR USE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NOT INFRINGEMENT.