



Smart Contract Audit Report

TrustBit Finance Smart Contract

2 May 2023



Table of Content

1 Executive Summary	2
Methodology	2
2 Findings Overview	6
2.1 Project info and Contract info	6
2.2 Summary	7
2.3 Key Findings	8
3 Detailed Description of Findings	9
3.1 PAUSER_ROLE privileged roles	9
3.2 DEFAULT_ADMIN_ROLE privileged role	10
4 Conclusion	11
5 Appendix	12
5.1 Basic Coding Assessment	12
5.2 Advanced Code Scrutiny	13
6 Disclaimer	14
References	15

1 EXECUTIVE SUMMARY

Numen Cyber Technology was engaged by TrustBit Finance to review smart contract implementation. The assessment was conducted in accordance with our systematic approach to evaluate potential security issues based upon customer requirement. The report provides detailed recommendations to resolve the issue and provide additional suggestions or recommendations for improvement.

The review identified two low-risk permissions issues.

The outcome of the assessment outlined in chapter 3 provides the system's owners a full description of the vulnerabilities identified, the associated risk rating for each vulnerability, and detailed recommendations that will resolve the underlying technical issue.

METHODOLOGY

To standardize the evaluation, we define the following terminology based on OWASP Risk Rating Methodology [10] which is the gold standard in risk assessment using the following risk models:

- Likelihood: represents how likely a particular vulnerability is to be uncovered and exploited in the wild.
- Impact: measures the technical loss and business damage of a successful attack.
- Severity: determine the overall criticality of the risk.

Likelihood and impact are categorized into three ratings: High, Medium and Low. Severity is determined by likelihood and impact and can be classified into four categories accordingly, Critical, High, Medium, Low shown in table 1.1.



Table 1.1: Overall Risk Severity

To evaluate the risk, we will be going through a list of items, and each would be labelled with a severity category. The audit was performed with a systematic approach guided by a comprehensive assessment list carefully designed to identify known and impactful security issues. If our tool or analysis does not identify any issue, the contract can be considered safe regarding the assessed item. For any discovered issue, we might further deploy contracts on our private test environment and run tests to confirm the findings. If necessary, we would additionally build a PoC to demonstrate the possibility of exploitation. The concrete list of check items is shown in Table 1.2.

- **Basic Coding Bugs:** We first statically analyze given smart contracts with our proprietary static code analyzer for known coding bugs, and then manually verify (reject or confirm) all the issues found by our tool.
- **Code and business security testing:** We further review business logics, examine system operations, and place DeFi-related aspects under scrutiny to uncover possible pitfalls and/or bugs.
- **Additional Recommendations:** We also provide additional suggestions regarding the coding and development of smart contracts from the perspective of proven programming practices.

Category	Assessment Item
-----------------	------------------------



Basic Coding Assessment	Apply Verification Control
	Authorization Access Control
	Forged Transfer Vulnerability
	Forged Transfer Notification
	Numeric Overflow
	Transaction Rollback Attack
	Transaction Block Stuffing Attack
	Soft fail Attack
	Hard fail Attack
	Abnormal Memo
	Abnormal Resource Consumption
	Secure Random Number
Advanced Source Code Scrutiny	Asset Security
	Cryptography Security
	Business Logic Review
	Source Code Functional Verification
	Account Authorization Control
	Sensitive Information Disclosure
	Circuit Breaker

	Blacklist Control
	System API Call Analysis
	Contract Deployment Consistency Check
Additional Recommendations	Semantic Consistency Checks
	Following Other Best Practices

Table 1.2: The Full List of Assessment Items

To better describe each issue we identified, we categorize the findings with Common Weakness Enumeration (CWE-699) [14], which is a community-developed list of software weakness types to better delineate and organize weaknesses around concepts frequently encountered in software development.

Numen Cyber



2 FINDINGS OVERVIEW

2.1 PROJECT INFO AND CONTRACT INFO

Project Information	Content
Project Name	TrustBit Finance
Audit Time	2023/4.28 - 2023/5.2
Language	Solidity
Chain	Binance
TrustBitFinance.sol	https://bscscan.com/address/0x3d66fC32829e154d270a2A0Ce4887D9FbC677DD5
Total Supply	20,000,000 TRS
Github	https://github.com/trustbit-finance/contracts

Table 2.1: PROJECT INFO AND CONTRACT INFO

Number

2.2 SUMMARY


Severity	Found	
Critical	0	
High	0	
Medium	0	
Low	2	
Informational	0	

Table 2.2: SUMMARY

Nummen Cyber



2.3 KEY FINDINGS

The review identified two low-risk permissions issues.

ID	Severity	Findings Title	Status	Confirm
NVE-001	Low	PAUSER_ROLE privileged role	Ignored	Confirmed
NVE-002	Low	DEFAULT_ADMIN_ROLE privileged role	Ignored	Confirmed

Table 2.3: Key Audit Findings

Nummen Cyber

3 DETAILED DESCRIPTION OF FINDINGS

3.1 PAUSER_ROLE PRIVILEGED ROLES

ID: NVE-001	Location: TrustBitFinance.sol
Severity: Low	Category: Authority Issues
Likelihood: Low	Impact: Medium

Description:

The TrustBitFinance contract is a Token contract whose main function is to issue tokens when the contract is deployed and to transfer the issued tokens to the contract deployer's address. the TrustBitFinance contract is deployed with the PAUSER_ROLE privileged role, which can be updated via the pause() and unpause() methods. paused variable. When the _paused variable is true, the overridden beforeTokenTransfer method will not continue to execute and the contract will not be able to perform transactions as the beforeTokenTransfer method is used in the _transfer, mint and _burn methods of the ERC20 contract. The contract PAUSER_ROLE privileged role can suspend and resume transactions, which may result in the project not functioning properly if the privileged role is maliciously controlled.

```
19     function pause() public onlyRole(PAUSER_ROLE) {
20         _pause();
21     }
22
23     function unpause() public onlyRole(PAUSER_ROLE) {
24         _unpause();
25     }
26
27     function _beforeTokenTransfer(address from, address to, uint256 amount)
28     internal
29     whenNotPaused
30     override
31     {
32         super._beforeTokenTransfer(from, to, amount);
33     }
```

Figure 1 PAUSER_ROLE PRIVILEGED ROLES

Recommendations:

Numen Cyber Labs recommends that privileged roles are carefully set with the _paused variable, and that privileged roles are managed using time locks and multiple signatures.

Result: PASS

Fix Result: Confirmed



3.2 DEFAULT_ADMIN_ROLE PRIVILEGED ROLE

ID: NVE-002

Location: TrustBitFinance.sol

Severity: Low

Category: Authority Issues

Likelihood: Low

Impact: Medium

Description:

The TrustBitFinance contract is deployed with the DEFAULT_ADMIN_ROLE privileged role set, which allows the PAUSER_ROLE privileged role to be set. The PAUSER_ROLE privileged PAUSER_ROLE privileged role can suspend and resume transactions, which can lead to the project not functioning properly if the DEFAULT_ADMIN_ROLE privileged role is maliciously controlled.

```
13     constructor() ERC20("TrustBit.Finance", "TRS") {
14         _grantRole(DEFAULT_ADMIN_ROLE, msg.sender);
15         _grantRole(PAUSER_ROLE, msg.sender);
16         _mint(msg.sender, 20000000 * 10 ** decimals());
17     }
18
19     function pause() public onlyRole(PAUSER_ROLE) {
20         _pause();
21     }
22
23     function unpause() public onlyRole(PAUSER_ROLE) {
24         _unpause();
25     }
26
27     function _beforeTokenTransfer(address from, address to, uint256 amount)
28     internal
29     whenNotPaused
30     override
31     {
32         super._beforeTokenTransfer(from, to, amount);
33     }
```

Figure 2 DEFAULT_ADMIN_ROLE PRIVILEGED ROLE

Recommendations:

Numen Cyber Labs recommends that the DEFAULT_ADMIN_ROLE privileged role be carefully set up with the PAUSER_ROLE privileged role, and that the privileged role be managed using time locks and multiple signatures.

Result: Pass

Fix Result: Confirmed



4 CONCLUSION

In this audit, we thoroughly analyzed TrustBit Finance smart contract implementation. The problems found are described and explained in detail in Section 3. The issues identified in the audit have been raised with the project manager and the two low risk privileged role issues will operate normally with the normal use of project staff. We therefore consider the audit result to be **Pass**. To improve this report, we greatly appreciate any constructive feedbacks or suggestions, on our methodology, audit findings, or potential gaps in scope/coverage.

Numen Cyber

5 APPENDIX

5.1 BASIC CODING ASSESSMENT

5.1.1 Apply Verification Control

- Description: The security of apply verification
- Result: Not found
- Severity: **Critical**

5.1.2 Authorization Access Control

- Description: Permission checks for external integral functions
- Result: Not found
- Severity: **Critical**

5.1.3 Forged Transfer Vulnerability

- Description: Assess whether there is a forged transfer notification vulnerability in the contract
- Result: Not found
- Severity: **Critical**

5.1.4 Transaction Rollback Attack

- Description: Assess whether there is transaction rollback attack vulnerability in the contract.
- Result: Not found
- Severity: **Critical**

5.1.5 Transaction Block Stuffing Attack

- Description: Assess whether there is transaction blocking attack vulnerability.
- Result: Not found
- Severity: **Critical**

5.1.6 soft fail Attack Assessment

- Description: Assess whether there is soft fail attack vulnerability.
- Result: Not found
- Severity: **Critical**

5.1.7 hard fail Attack Assessment

- Description: Examine for hard fail attack vulnerability
- Result: Not found
- Severity: **Critical**

5.1.8 Abnormal Memo Assessment

- Description: Assess whether there is abnormal memo vulnerability in the contract.
- Result: Not found
- Severity: **Critical**

5.1.9 Abnormal Resource Consumption

- Description: Examine whether abnormal resource consumption in contract processing.
- Result: Not found
- Severity: **Critical**

5.1.10 Random Number Security

- Description: Examine whether the code uses insecure random number.
- Result: Not found
- Severity: **Critical**

5.2 ADVANCED CODE SCRUTINY

5.2.1 Cryptography Security

- Description: Examine for weakness in cryptograph implementation.
- Results: Not Found
- Severity: **High**

5.2.2 Account Permission Control

- Description: Examine permission control issue in the contract
- Results: Not Found
- Severity: **Medium**

5.2.3 Malicious Code Behaviour

- Description: Examine whether sensitive behaviour present in the code
- Results: Not found
- Severity: **Medium**

5.2.4 Sensitive Information Disclosure

- Description: Examine whether sensitive information disclosure issue present in the code.
- Result: Not found
- Severity: **Medium**

5.2.5 System API

- Description: Examine whether system API application issue present in the code
- Results: Not found
- Severity: **Low**



6 DISCLAIMER

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to the Company in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes without Numen's prior written consent.

This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts Numen to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. Numen's position is that each company and individual are responsible for their own due diligence and continuous security. Numen's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

REFERENCES

[1] MITRE. CWE- 191: Integer Underflow (Wrap or Wraparound).

<https://cwe.mitre.org/data/definitions/191.html>.

[2] MITRE. CWE- 197: Numeric Truncation Error.

<https://cwe.mitre.org/data/definitions/197.html>.

[3] MITRE. CWE-400: Uncontrolled Resource Consumption.

<https://cwe.mitre.org/data/definitions/400.html>.

[4] MITRE. CWE-440: Expected Behavior Violation.

<https://cwe.mitre.org/data/definitions/440.html>.

[5] MITRE. CWE-684: Protection Mechanism Failure.

<https://cwe.mitre.org/data/definitions/693.html>.

[6] MITRE. CWE CATEGORY: 7PK - Security Features.

<https://cwe.mitre.org/data/definitions/254.html>.

[7] MITRE. CWE CATEGORY: Behavioral Problems.

<https://cwe.mitre.org/data/definitions/438.html>.

[8] MITRE. CWE CATEGORY: Numeric Errors.

<https://cwe.mitre.org/data/definitions/189.html>.

[9] MITRE. CWE CATEGORY: Resource Management Errors.

<https://cwe.mitre.org/data/definitions/399.html>.

[10] OWASP. Risk Rating Methodology.

https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology



Numen Cyber Technology Pte. Ltd.

11 North Buona Vista Drive, #04-09,
The Metropolis, Singapore 138589

Tel: 65-63555555

Fax: 65-63666666

Email: sales@numencyber.com

Web: <https://numencyber.com>