Smart Contract Security Audit V1

Taste Tester Smart Contract

8/4/2022



business@saferico.com https://t.me/SFI_ANN

Table of Contents

Table of Contents

Background

Project Information

NFT Information

Executive Summary

File and Function Level Report

File in Scope:

Issues Checking Status

Severity Definitions Audit Findings

Automatic testing

Testing proves Inheritance graph Call graph

Unified Modeling Language (UML)

Functions signature Automatic general report

Conclusion

Disclaimer

Background

The purpose of the audit was to achieve the following:

- Ensure that the smart contract functions as intended.
- Identify potential security issues with the smart contract.

The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Project Information

- Platform: polygon
- Contract Address: 0x8109ccbC9d9d1beF822B4C64feFeBa91fF29f78E
- Code:

https://mumbai.polygonscan.com/address/0x8109ccbC9d9d1beF822B4C64feFeBa91fF29f78E#code

NFT Information

• Name: TT

• Total Supply: 10000

- Holders:
- Total transactions:

Contracts address deployed to test net (Polygon)

Taste Tester Smart contract on Polygon test net to test write functions by the auditor.

https://mumbai.polygonscan.com/address/0x99d3af4eb349037b8cd54267195ab1a752d4e1e3

Executive Summary

According to our assessment, the customer's solidity smart contract is **Well-Secured**. Because the team fix all low issues.

Well Secured	√
Secured	
Poor Secured	
Insecure	

Automated checks are with remix IDE. All issues were performed by the team, which included the analysis of code functionality, manual audit found during automated analysis were manually reviewed and applicable vulnerabilities are presented in the audit overview section. The general overview is presented in the Project Information section and all issues found are located in the audit overview section.

Team found 0 critical, 0 high, 0 medium, 1 low, 0 very low-level issues and 1 note in all solidity files of the contract

The files:

TT.sol

File and Function Level Report

File in Scope:

Contract Name	SHA 256	Contract Address
	hash	
	453d09f45876608e4e1a88b	0xcc1a754478bd5B95654B4AcB91Ae654Dd5
11.501	90a02a9057074b5b54c1f2ef	A6d587
	298e26764d0644cad	

• Contract: TT

• Inherit: ERC721Enumerable, Ownable

• Observation: All passed including security check

Test Report: passedScore: passed

• Conclusion: passed

Function	Test Result	Type / Return Type	Score
name	✓	Read / public	Passed
symbol	√	Read / public	Passed
baseExtension	√	Read / public	Passed
supportsInterface	√	Read / public	Passed
cost	√	Read / public	Passed
balanceOf	√	Read / public	Passed
Owner	✓	Read / public	Passed
tokenByIndex	√	Read / public	Passed
tokenOfOwnerByIndex	✓	Read / public	Passed
getApprovedForAll	√	Read / public	Passed
ownerOf	√	Read / public	Passed
getApproved	√	Read / public	Passed

baseURI	√	Read / public	Passed
tokenURI	√	Read / public	Passed
totalSupply	√	Read / public	Passed
MAX_SUPPLY	√	Read / public	Passed
approve	√	Write / public	Passed
safeTransferFrom	√	Write / public	Passed
safeTransferFrom	√	Write / public	Passed
setBaseURI	√	Write / public	Passed
mint	√	Write / payable	Passed
transferOwnership	✓	Write / public	Passed
setApprovalForAll	√	Write / public	Passed
transferFrom	√	Write / public	Passed
renounceOwnership	√	Write / public	Passed
withdraw	✓	Write / payable	Passed

Issues Checking Status

No.	Issue Description	Checking Status
1	Compiler warnings.	Passed
2	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Design Logic.	Passed
6	Timestamp dependence.	Passed with Notes
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Passed
10	Methods execution permissions.	Passed
11	Economy model. If application logic is based on an incorrect economic model, the application would not function correctly and participants would incur financial losses. This type of issue is most often found in bonus rewards systems, Staking and Farming contracts, Vault and Vesting contracts, etc.	Passed
12	The impact of the exchange rate on the logic.	Passed
13	Private user data leaks.	Passed
14	Malicious Event log.	Passed
15	Scoping and Declarations.	Passed
16	Uninitialized storage pointers.	Passed
17	Arithmetic accuracy.	Passed

Severity Definitions

Risk Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to tokens loss etc.
High	High-level vulnerabilities are difficult to exploit; however, they also have significant impact on smart contract execution, e.g. public access to crucial functions
Medium	Medium-level vulnerabilities are important to fix; however, they can't lead to tokens lose
Low	Low-level vulnerabilities are mostly related to outdated, unused etc. code snippets, that can't have significant impact on execution
Note	Lowest-level vulnerabilities, code style violations and info statements can't affect smart contract execution and can be ignored.

Audit Findings

Critical:

No critical severity vulnerabilities were found.

High:

No high severity vulnerabilities were found.

Medium:

No Medium severity vulnerabilities were found

Low:

#Multiple pragma statements

Line	Pragma
7	pragma solidity ^0.8.0;
77	pragma solidity ^0.8.1;
304	pragma solidity ^0.8.0;
332	pragma solidity ^0.8.0;
360	pragma solidity ^0.8.0;
391	pragma solidity ^0.8.0;
536	pragma solidity ^0.8.0;
567	pragma solidity ^0.8.0;
596	pragma solidity ^0.8.0;
623	pragma solidity ^0.8.0;
1072	pragma solidity ^0.8.0;
1237	pragma solidity ^0.8.0;
1313	pragma solidity ^0.8.0;

Description

There are multiple pragma statements in the code. Only the compiler version 0.8.7 will work with the code, but keeping only one pragma statement helps in maintaining readability of the code.

Remediation

Keep a single pragma statement.

Status: Closed. Fixed In version 2

Very Low:

No Very Low severity vulnerabilities were found.

Notes:

#Missing SPDX-License-Identifier:

Description

Warning: SPDX license identifier not provided in source file. Before publishing, consider adding a comment containing "SPDX-License-Identifier: <SPDX-License>" to each source file. Use "SPDX-License-Identifier: UNLICENSED" for non-open-source code. Please see https://spdx.org for more information .

Remediation

Add License Identifier

// SPDX-License-Identifier: UNLICENSE

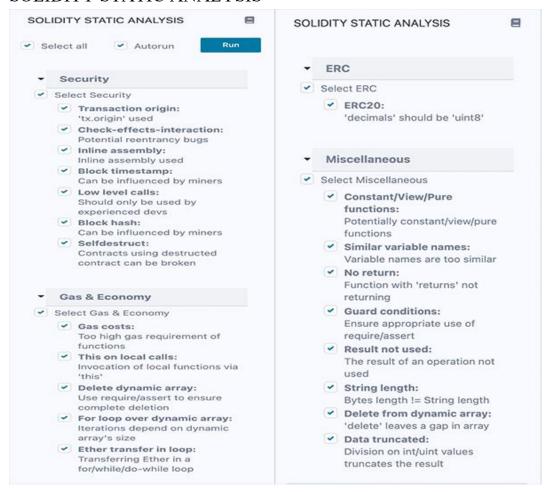
Status: Closed. Fixed In version 2

Automatic Testing

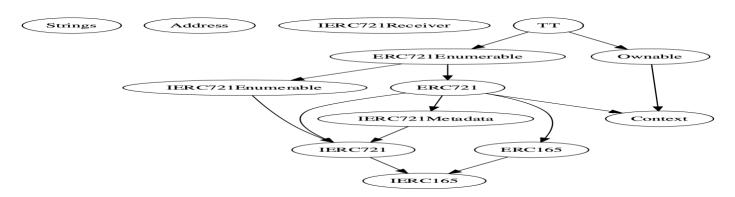
1- Check for security



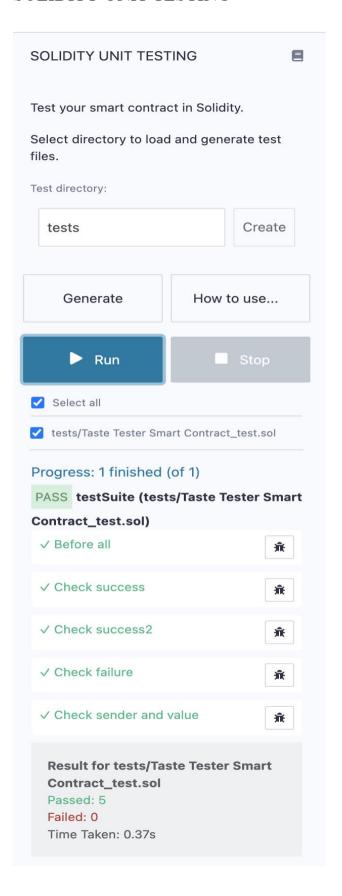
2- SOLIDITY STATIC ANALYSIS



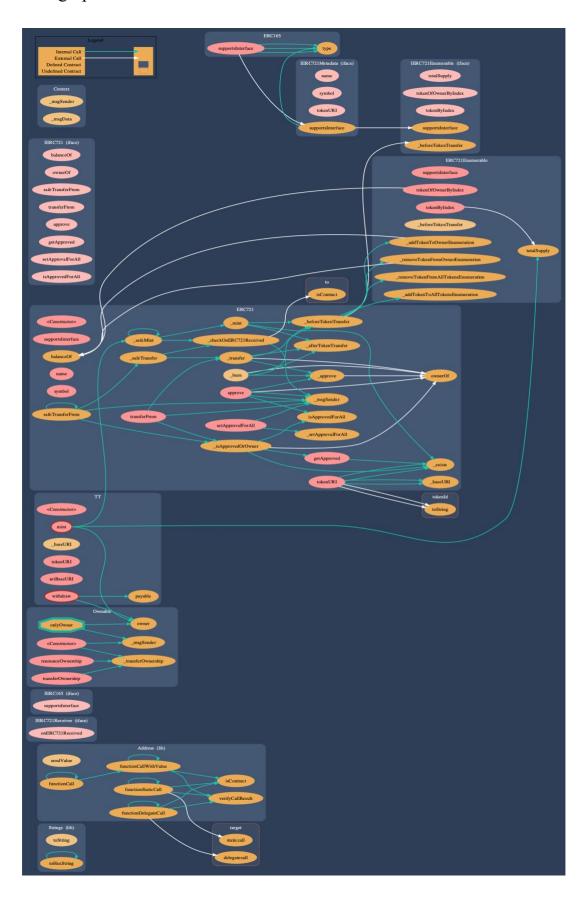
3- Inheritance graph



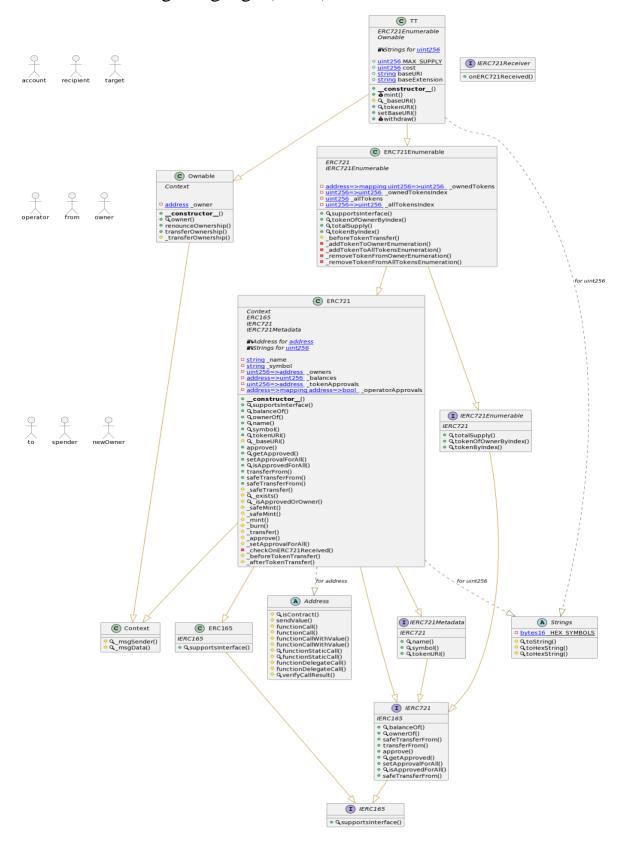
4- SOLIDITY UNIT TESTING



5- Call graph



Unified Modeling Language (UML)



Functions signature

```
Sighash | Function Signature
 _____
 16279055 => isContract(address)
 6900a3ae => toString(uint256)
 8fba8d5c => toHexString(uint256)
 63elcbea => toHexString(uint256,uint256)
 24a084df => sendValue(address,uint256)
 a0b5ffb0 => functionCall(address,bytes)
 241b5886 => functionCall(address,bytes,string)
 2a011594 => functionCallWithValue(address, bytes, uint256)
 d525ab8a => functionCallWithValue(address, bytes, uint256, string)
 c21d36f3 => functionStaticCall(address,bytes)
 dbc40fb9 => functionStaticCall(address,bytes,string)
 ee33b7e2 => functionDelegateCall(address,bytes)
 57387df0 => functionDelegateCall(address,bytes,string)
 946b5793 => verifyCallResult(bool, bytes, string)
 150b7a02 => onERC721Received(address,address,uint256,bytes)
 01ffc9a7 => supportsInterface(bytes4)
 70a08231 => balanceOf(address)
 6352211e \Rightarrow ownerOf(uint256)
 42842e0e => safeTransferFrom(address,address,uint256)
 23b872dd => transferFrom(address,address,uint256)
 095ea7b3 => approve(address, uint256)
 081812fc => getApproved(uint256)
 a22cb465 => setApprovalForAll(address,bool)
 e985e9c5 => isApprovedForAll(address,address)
 b88d4fde => safeTransferFrom(address,address,uint256,bytes)
 18160ddd => totalSupply()
 2f745c59 => tokenOfOwnerByIndex(address,uint256)
 4f6ccce7 => tokenByIndex(uint256)
 06fdde03 => name()
 95d89b41 => symbol()
c87b56dd => tokenURI(uint256)

119df25f => msgSender()

8b49d47e => msgData()

743976a0 => baseURI()

24b6b8c0 => exists(uint256)

4cdc9549 => isApprovedOrOwner(address, uint256)

8a4f832b => safeMint(address, uint256)

6a4f832b => mint(address, uint256)

9b1f9e74 => burn(uint256)

30e0789e => transfer(address, uint256)

8c4e3f32 => approve(address, uint256)

8c4e3f32 => setApprovalForAll(address, address, uint256, bytes)

1fd01de1 => checkOnERC721Received(address, address, uint256)

8f811a1c => afterTokenTransfer(address, address, uint256)

69025b5f => addTokenToOwnerEnumeration(address, uint256)

68df0d53 => removeTokenFromOwnerEnumeration(uint256)

4cbb4a0a => removeTokenFromOwnerEnumeration(uint256)
 c87b56dd => tokenURI(uint256)
 4cbb4a0a => removeTokenFromAllTokensEnumeration(uint256)
```

```
8da5cb5b => owner()
715018a6 => renounceOwnership()
f2fde38b => transferOwnership(address)
d29d44ee => _transferOwnership(address)
40c10f19 => mint(address, uint256)
55f804b3 => setBaseURI(string)
3ccfd60b => withdraw()
```

Automatic general report

```
Files Description Table
| File Name | SHA-1 Hash |
|-----|
| /Users/macbook/Desktop/smart contracts/Taste Tester Smart Contract.sol |
3a6bab74eb95c2a2e6271272932672bef4ef2f6b |
Contracts Description Table
| Contract |
              Type Bases
| **Function Name** | **Visibility** | **Mutability** |
**Modifiers** |
| **Strings** | Library | ||| | | | | | | | | |
| L | toString | Internal 🖺 | | |
| L | toHexString | Internal A | | | |
| L | toHexString | Internal A | | |
| L | functionCall | Internal A | O | |
| L | functionStaticCall | Internal 🖺 | | |
| | verifyCallResult | Internal | | | |
| **IERC721Receiver** | Interface | |||
| L | onERC721Received | External | | NO | |
| **IERC165** | Interface | |||
| L | supportsInterface | External | | | NO | |
| **ERC165** | Implementation | IERC165 |||
| L | supportsInterface | Public | | NO |
| **IERC721** | Interface | IERC165 |||
| L | balanceOf | External | | | NO | | | L | ownerOf | External | | NO | |
| L | transferFrom | External | | NO | |
| L | approve | External [ | O | NO[ ]
| L | getApproved | External | | NO | |
| L | isApprovedForAll | External | | NO | |
```

```
| **IERC721Enumerable** | Interface | IERC721 ||| | |
| L | totalSupply | External | | NO | |
| L | tokenOfOwnerByIndex | External | | | NO | |
| L | tokenByIndex | External | | NO | |
| **IERC721Metadata** | Interface | IERC721 |||
| L | name | External | | NO | |
| L | symbol | External | | | NO
| L | tokenURI | External | | NO | |
| **Context** | Implementation | |||
| L | msgSender | Internal 🖺 | | |
| L | msgData | Internal 🖺 | | |
| **ERC721** | Implementation_| Context, ERC165, IERC721, IERC721Metadata |||
 L | <Constructor> | Public | | | NO |
| L | supportsInterface | Public | |
| L | balanceOf | Public | | NO | |
 L | ownerOf | Public | | NO | |
 L | name | Public | | NO | |
 L | symbol | Public | | | NO
 L | tokenURI | Public | | NO | |
 | L | getApproved | Public | | NO | |
 L | setApprovalForAll | Public | | O | NO | |
 L | isApprovedForAll | Public | | NO | |
 L | transferFrom | Public | | NO | |
 L | safeTransfer | Internal 🖺 | 🗓 | |
 _ | _exists | Internal 🖺 |
                       L | isApprovedOrOwner | Internal 🖺 |
 L | safeMint | Internal 🖺 | 🔘 | |
 L | mint | Internal 🖺 | 🔘 | |
 | approve | Internal | | | | |
| L | setApprovalForAll | Internal A | O
 L | checkOnERC721Received | Private 🖺 | 🔘 | |
 | L | afterTokenTransfer | Internal 🖺 | 🔘 | |
| **ERC721Enumerable** | Implementation | ERC721, IERC721Enumerable |||
 | supportsInterface | Public | | NO
| L | tokenOfOwnerByIndex | Public | | NO | |
| L | totalSupply | Public | | NO | |
 | L | beforeTokenTransfer | Internal 🖺 | 🔘 | |
| L | _addTokenToOwnerEnumeration | Private 🖺 | 🔘
removeTokenFromOwnerEnumeration | Private 🖺 | 🔘 | |
```

```
| L | removeTokenFromAllTokensEnumeration | Private 🖺 | 🔘 | | | | | |
| **Ownable** | Implementation | Context |||
| L | <Constructor> | Public | | | NO | |
| L | owner | Public | | NO | |
| L | renounceOwnership | Public | | onlyOwner | L | transferOwnership | Public | onlyOwner |
| **TT** | Implementation | ERC721Enumerable, Ownable |||
| L | <Constructor> | Public | | | | ERC721 |
| L | mint | Public | | III | NO | |
| L | baseURI | Internal 🖺 | | |
| L | withdraw | Public | | I I | onlyOwner |
Legend
| Symbol | Meaning | |
|---|---|---|
| Function can modify state | | Function is payable |
```

Conclusion

The contracts are written systematically. Team found no critical issues. So, it is good to go for production.

Since possible test cases can be unlimited and developer level documentation (code flow diagram with function level description) not provided, for such an extensive smart contract protocol, we provide no such guarantee of future outcomes. We have used all the latest static tools and manual observations to cover maximum possible test cases to scan Everything.

Security state of the reviewed contract is "Well-secured".

- ✓ No volatile code.
- ✓ Not many high severity issues were found.

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as of the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against the team on the basis of what it says or doesn't say, or how team produced it, and it is important for you to conduct your own independent investigations before making any decisions. team go into more detail on this in the below disclaimer below – please make sure to read it in full.

By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice. No one shall have any right to rely on the report or its contents, and Saferico and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (Saferico s) owe no duty of care towards you or any other person, nor does Saferico make any warranty or representation to any person on the accuracy or completeness of the report. The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and Saferico hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, Saferico hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Saferico, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report. The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.