HackSafe

Smart Contract Security Audit Report

One Rare Token

September 2022

Security Status

★★★★☆

www.hacksafe.io

Audit Details

Audited project
One Rare Token

Deployer address
0x6C9a2aF2f6C8f808AE6aE89A5B3C80f2414480aa

Client contacts
One Rare Token Team

Blockchain
Polygon

Website
https://onerare.io/

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at 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 us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the below disclaimer below – please make sure to read it in full.

DISCLAIMER: 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 TechRate and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (HackSafe) owe no duty of care towards you or any other person, nor does HackSafe 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 HackSafe 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, HackSafe hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against HackSafe, 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.

Procedure

Step 1 - In-Depth Manual Review

Manual line-by-line code reviews to ensure the logic behind each function is sound and safe from various attack vectors. This is the most important and lengthy portion of the audit process (as automated tools often cannot find the nuances that lead to exploits such as flash loan attacks).

Step 2 - Automated Testing

Simulation of a variety of interactions with your Smart Contract on a test blockchain leveraging a combination of automated test tools and manual testing to determine if any security vulnerabilities exist.

Step 3 – Leadership Review

The engineers assigned to the audit will schedule meetings with our leadership team to review the contracts, any comments or findings, and ask questions to further apply adversarial thinking to discuss less common attack vectors.

Step 4 - Resolution of Issues

Consulting with the team to provide our recommendations to ensure the code's security and optimize its gas efficiency, if possible. We assist project team's in resolving any outstanding issues or implementing our recommendations.

Step 5 - Published Audit Report

Boiling down results and findings into an easy-to-read report tailored to the project. Our audit reports highlight resolved issues and any risks that exist to the project or its users, along with any remaining suggested remediation measures. Diagrams are included at the end of each report to help users understand the interactions which occur within the project.

Background

HackSafe was commissioned by One Rare Token to perform an audit of smart contracts:

- https://polygonscan.com/address/0xff2382bd52efacef02cc895bcbfc4618608aa56f#code

The purpose of the audit was to achieve the

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

The information in this report should be 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.

Contract Details

Token contract details for 01.09.2022

Token Type	: ERC20
Contract name	: ORareToken
Contract address	: 0xFF2382Bd52efaceF02Cc895bcBFc4618608AA56F
Compiler version	: v0.8.9+commit.e5eed63a
Total supply	: 100,000,000
Token Ticker	: ORARE
Decimals	: 18
Token Holders	: 2,089
Transactions count	: 26,315
Contract deployer address	: 0x6C9a2aF2f6C8f808AE6aE89A5B3C80f2414480aa
Owner address	: 0xb59f908ffb2765cc308eaccdecb94c5980549549

Social profiles

Twitter profile	: https://twitter.com/onerarenft
Telegram profile	: https://t.me/+pO88PtVrRQc3NWE9
LinkedIn profile	: https://www.linkedin.com/company/onerare/
Facebook profile	: https://www.facebook.com/onerarenft

Audit Summary

According to the standard audit assessment, Customer`s solidity smart contracts are “Secure”. This token contract does contain owner control, which do not make it fully decentralized as owner does have control over smart contract.

Insecure	Poor secured	Secure	Well-secured

You are here 

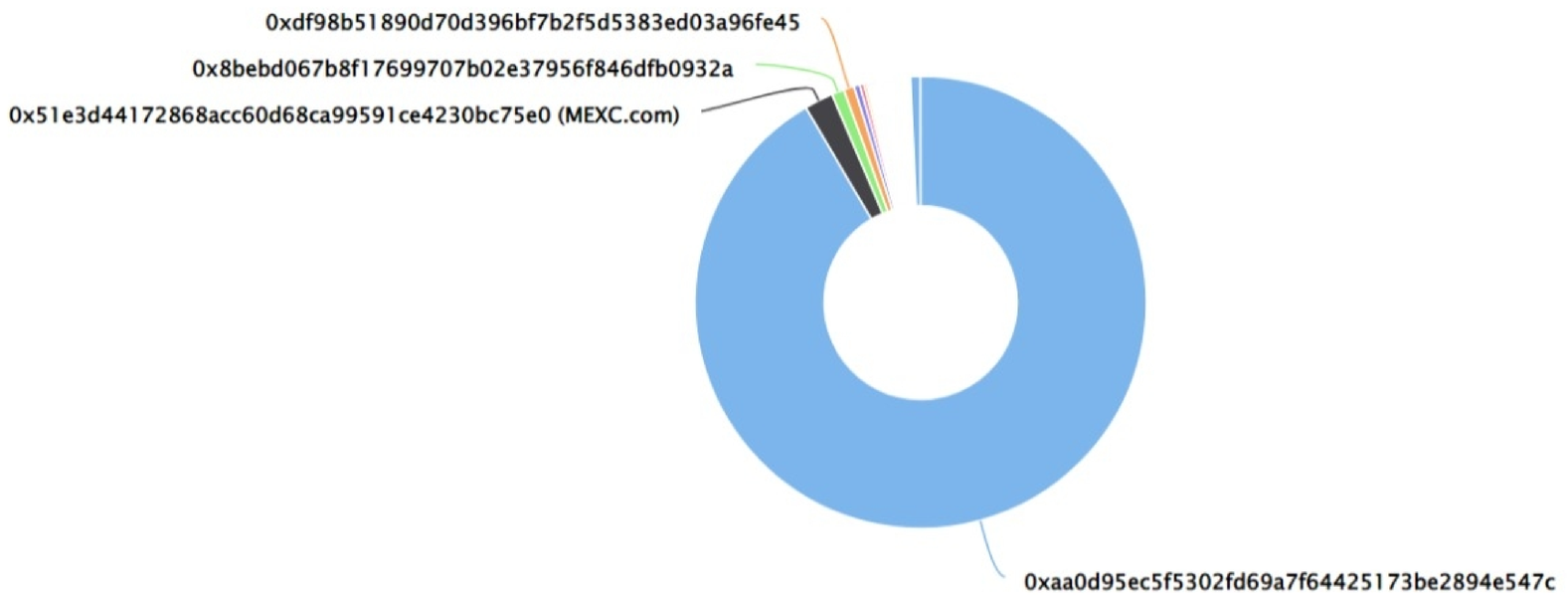
We used various tools like Slither, Mythril and Remix IDE. At the same time this finding is based on critical analysis of the manual audit. All issues found during automated analysis were manually reviewed and applicable vulnerabilities are presented in the issues checking status.

We found 0 critical, 0 high, 1 medium and 0 low and some very low-level issues. These issues are not critical ones.

One Rare Token Token Distribution

The top 100 holders collectively own 99.28% (99,284,254.99 Tokens) of One Rare Token

Token Total Supply: 100,000,000.00 Token | Total Token Holders: 2,089



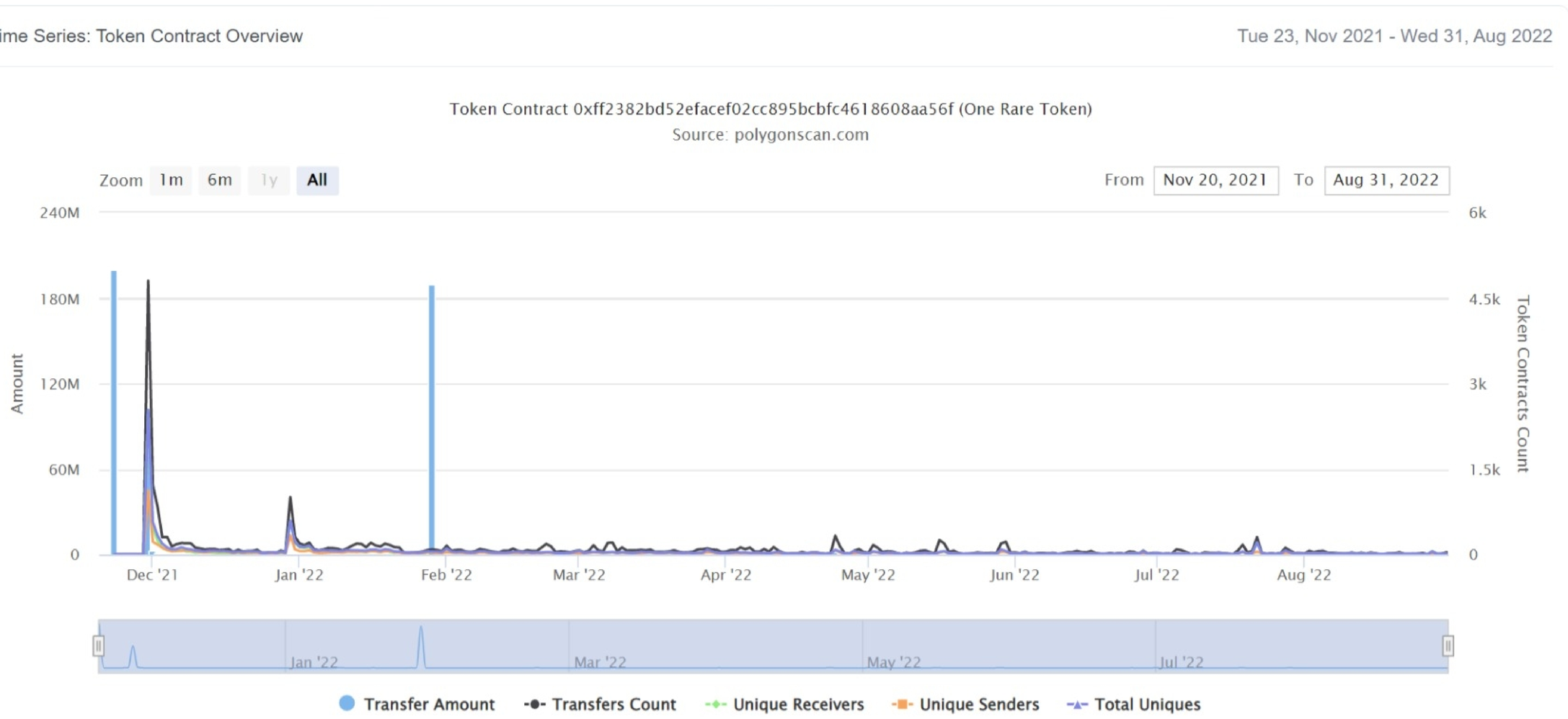
One Rare Token Top 20 Token Holders

(A total of 99,284,254.99 tokens held by the top 100 accounts from the total supply of 100,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0xaa0d95ec5f5302fd69a7f64425173be2894e547c	91,506,334.224869789093026392	91.5063%
2	MEXC.com	2,065,638.033957966607180023	2.0656%
3	0x8bebd067b8f17699707b02e37956f846dfb0932a	912,647.338242161500529552	0.9126%
4	0xdf98b51890d70d396bf7b2f5d5383ed03a96fe45	750,254.026741434118591901	0.7503%
5	0xb1e5ec3b7919868fc7836276a69016bd47e82ff5	410,366.694983472316347504	0.4104%
6	0x7df7c16657d0b983fde0487576717536f93db5c8	294,976.7506930018745399	0.2950%
7	0x80fcc1b8c390d054f5e64811977d3d5246ffe162	200,000	0.2000%
8	0xe155776328c199d4163dabfe9706945094d6bf58	165,252.08662643709806468	0.1653%
9	0x5ce23e8f2bcbf9cf2bf68842adf740fdf6c0a54d	160,005.924346022360311705	0.1600%
10	0x5ba7614b0b5e901a762af4c03c6a33d85d7b35dd	159,940.284424184118268551	0.1599%
11	0x9d7117a07fca9f22911d379a9fd5118a5fa4f448	146,884.292572662733664411	0.1469%
12	0x2467ebaf6860532384639836ca40706cd8f2cd17	129,840.115739478120531926	0.1298%
13	0x1dc3bcc07b93c73c476d7e1056b64c8bd947184a	114,955.438523042720085328	0.1150%
14	0x8aa07ce2a1d6f404be2e107bdceafcef8382d17f	105,634.407325618716672059	0.1056%
15	0x719855c7540ca4f4e71a4f70c34725000c9a79fd	100,160.75333847735275327	0.1002%
16	0x03cdc580e7de9d805483773cd73f00e9c6b2d0a0	86,321.9039317323062364	0.0863%
17	0x110b31a4f689e82db300e61457e32c817ad3f15b	80,740.126916957425042454	0.0807%
18	0xa48774c17e9e1ef63abd2ce5e76c1e62469909ed	80,637.017319987076570214	0.0806%
19	0x8c3b60a48219db57da76b34f65c3bc69bd91110a	80,476.9	0.0805%
20	0xfc3a2aa51544d1cacc55e270c0fe418fe66e55ac	75,125.34019602299702228	0.0751%

One Rare Token Token Distribution

One Rare Token Contract Overview



Contract functions details

ORareToken.sol
+ ORareToken (ERC20Permit, Ownable)
    -<constructor>
    -[Ext] recoverToken #
      -modifiers: onlyOwner

SafeERC20.sol
+[Lib] SafeERC20
    -[Int] safeTransfer
    -[Int] safeTransferFrom
    -[Int] safeApprove
    -[Int] safeIncreaseAllowance
    -[Int] safeDecreaseAllowance
    -[Pvt] _callOptionalReturn

ERC20.sol
+ ERC20 (Context, IERC20, IERC20Metadata)
    -<constructor>
    -[Pub] name
    -[Pub] symbol
    -[Pub] decimals
    -[Pub] totalSupply
    -[Pub] balanceOf
    -[Pub] transfer #
    -[Pub] allowance
    -[Pub] approve #
    -[Pub] transferFrom #
    -[Pub] increaseAllowance #
    -[Pub] decreaseAllowance #
    -[Int] _transfer #
    -[Int] _mint #
    -[Int] _burn #
    -[Int] _approve #
    -[Int] _beforeTokenTransfer #
    -[Int] _afterTokenTransfer #

IERC20.sol
+ [Int] IERC20
    -[Ext] totalSupply
    -[Ext] balanceOf

Contract functions details

-[Ext] transfer
-[Ext] allowance
-[Ext] approve
-[Ext] transferFrom

ERC20Permit.sol
+ ERC20Permit (ERC20, IERC2612Permit)
    -< constructor>
    -[Pub] permit
    -[Pub] nonces

Ownable.sol
+ Ownable (Context)
    -<constructor>
    -[Pub] owner
    -[Pub] initOwner #
    -[Pub] transferOwnership #
        -modifiers: onlyOwner

Address.sol
+[Lib] Address
    -[Int] isContract
    -[Int] sendValue
    -[Int] functionCall
    -[Int] functionCall
    -[Int] functionCallWithValue
    -[Int] functionCallWithValue
    -[Int] functionStaticCall
    -[Int] functionStaticCall
    -[Int] functionDelegateCall
    -[Int] functionDelegateCall
    -[Int] verifyCallResult

IERC20Metadata.sol
-[Int] IERC20Metadata (IERC20)
    -[Ext] name
    -[Ext] symbol
    -[Ext] decimals

Contract functions details

Context.sol
+ Context
    -[Int] _msgSender
    -[Int] _msgData

Counters.sol
+ [Lib] Counters
    -[Int] current
    -[Int] increment
    -[Int] decrement
    -[Int] reset

IERC2612Permit.sol
+[Int] IERC2612Permit
    -[Ext] permit
    -[Ext] nonces

($) = payable function
= non-constant function

Issues Checking Status

No.	Title	Status
1.	Unlocked Compiler Version	Passed
2.	Missing Input Validation	Passed
3.	Race conditions and Reentrancy. Cross-function race conditions.	Passed
4.	Possible delays in data delivery	Passed
5.	Oracle calls.	Passed
6.	Timestamp dependence.	Passed
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 of the contract.	Passed
12.	Private use data leaks.	Passed
13.	Malicious Event log.	Passed
14.	Scoping and Declarations.	Passed
15.	Uninitialized storage pointers.	Passed
16.	Arithmetic accuracy.	Passed
17.	Design Logic.	Passed
18.	Safe Open Zeppelin contracts implementation and usage.	Medium issue
19.	Incorrect Naming State Variable	Passed
20.	Too old version	Passed

Severity Definitions

Risk Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to assets loss or data manipulations.
High	High-level vulnerabilities are difficult to exploit; however, they also have a 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 assets loss or data manipulations.
Low	Low-level vulnerabilities are mostly related to outdated, unused, etc. code snippets that can't have a significant impact on execution.

Security Issues

Critical Severity Issues

No critical severity issue found.

High Severity Issues

No high severity issue found.

Medium Severity Issues

One medium severity issues found.

1. Safe Open Zeppelin contracts implementation and usage.

- **Description**

  ORareToken.sol, ERC20Permit.sol, Ownable.sol contracts have imported openzeppelin contracts direct from github repository of openzeppelin.

- **Recommendation**

  We advise you to not direct import from any github repository as any changes in that may affect your smart contract too.

Low Severity Issues

No low severity issue found.

Centralization

Owner Privileges :

- One Rare Token Contract:
    - Owner can transfer ownership.
    - Owner can recover token.

This smart contract has some functions which can be executed by the Admin (Owner) only. If the admin wallet private key would be compromised, then it would create trouble as smart contract ownership has not been renounced. Following are Admin functions functions:

- Transferownership
- Recovertoken

Conclusion

Smart contract contains medium severity issues! The further transfer and operations with the fund raised are not related to this particular contract.

HackSafe note: Please check the disclaimer above and note, the audit makes no statements or warranties on business model, investment attractiveness or code sustainability. The report is provided for the only contract mentioned in the report and does not include any other potential contracts deployed by Owner.