HackSafe

Smart Contract Security Audit Report

ZelaaNFT

July 2022

Security Status

★ ★ ★ ★ ⯪

www.hacksafe.io

Audit Details

Audited project
ZelaaNFT

Deployer address
0xf03D2798F4735e5382130e67959a51e682C8c8A1

Client contacts
ZelaaNFT team

Blockchain
Polygon

Website
https://www.zelaanft.com/

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 ZelaaNFT to perform an audit of smart contracts:

- https://polygonscan.com/address/0x173bead008df7ba34d0d04e3d6dd01a6db345c52#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 15.07.2022

Token Type	: ERC20
Contract name	: NFT
Contract address	: 0x173bEAd008df7Ba34d0d04e3D6dd01a6Db345C52
Compiler version	: v0.8.15+commit.e14f2714
Total supply	: 8,630,501
Token Ticker	: NFTz
Decimals	: 18
Token Holders	: 727
Top 100 token holder's dominance	: 91.21%
Transactions count	: 1,220
Contract deployer address	: 0xf03D2798F4735e5382130e67959a51e682C8c8A1
Owner address	: 0x79C87785531C79dd4624aE6b1FAE0029FdcaDe90

Social profiles

Twitter Profile	: https://www.twitter.com/zelaanft/
Telegram Profile	: https://t.me/ZelaaNFT
Coinmarketcap profile	: https://coinmarketcap.com/currencies/zelaanft/
Coingecko profile	: https://www.coingecko.com/en/coins/zelaanft/

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 

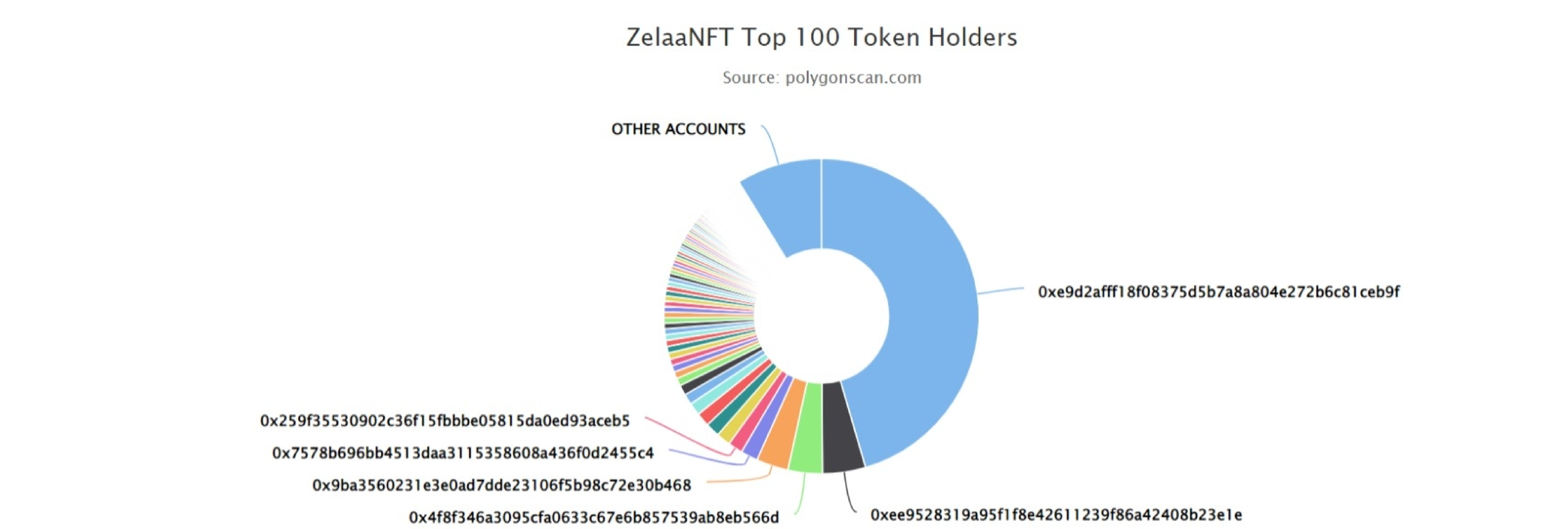
We used various tools like Slither, Mythril and Remix IDE. At the same time this finding is based on critical analysis of the manual audit. All issues found during automated analysis were manually reviewed and applicable vulnerabilities are presented in the issues checking status.

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

ZeelaNFT Token Distribution

The top 100 holders collectively own 91.21% (7,871,590.46 Tokens) of ZelaaNFT

Token Total Supply: 8,630,501.91 Token | Total Token Holders: 727



(A total of 7,871,590.46 tokens held by the top 100 accounts from the total supply of 8,630,501.91 token)

ZeelaNFT Top 20 Token Holders

Rank	Address	Quantity (Token)	Percentage
1	0xe9d2afff18f08375d5b7a8a804e272b6c81ceb9f	3,927,071.276537897897357299	45.5022%
2	0xee9528319a95f1f8e42611239f86a42408b23e1e	379,955	4.4025%
3	0x4f8f346a3095cfa0633c67e6b857539ab8eb566d	305,000	3.5340%
4	0x9ba3560231e3e0ad7dde23106f5b98c72e30b468	283,987.740757	3.2905%
5	0x7578b696bb4513daa3115358608a436f0d2455c4	151,000	1.7496%
6	0x259f35530902c36f15fbbbe05815da0ed93aceb5	130,769.15	1.5152%
7	0x3e8072c629a3aa09b79939d0fdf00cdaf5777e04	125,001.28761365	1.4484%
8	0xf836b6a6366ae436d02227118b2fb26a239e8e60	124,968.884975	1.4480%
9	0xcff0c24a354313519ec1e89a2e80390327261560	122,193.5952588	1.4158%
10	0x65ffc3d3e2fc29e00f22e330f351bd065f8e6f8c	111,639.80328225	1.2935%
11	0xf2a236e93821cd8b4781c1bd33dda37a380d3a6a	95,000	1.1007%
12	0x84d684ce1944fcd9c8ee4cf2db8914b26d051666	90,255	1.0458%
13	0x68a2dd926d2cb989d1befe3a0b8b94673a96d4da	67,271.9737	0.7795%
14	0x3233b921e1afabe6501d0191cceead2415d572fb	60,157.021210475	0.6970%
15	0x52c6a200a0649c595db278c918f2747fdc7369d1	59,052.7500844895	0.6842%
16	0x0398e1e805f057709fc2f753105319ddf1eabe2a	57,584.02565	0.6672%
17	0xe0d674decc698c195ec27691aad29cc856b064d3	57,500.0000000305	0.6662%
18	0x324cbf581e2285e2f0fbe757778708f3005d1815	56,565	0.6554%
19	0x19db1d4285bae5122dcedf271e988a0a360056b2	53,610.5716	0.6212%
20	0xbcbd199aa71f2494404abd8bc426bb5ceea62e68	50,837.7799692635	0.5890%

ZeelaNFT Token Distribution

ZeelaNFT Contract Overview

Contract functions details

+ [Int] IERC20
    -[Ext] totalSupply
    -[Ext] balanceOf
    -[Ext] transfer
    -[Ext] allowance
    -[Ext] approve
    -[Ext] transferFrom

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

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

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

+ NFT (Context, IERC20, IERC20Metadata, Ownable)
    -<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#

Contract functions details

-[Int] _burn #
-[Int] _approve #
-[Int] _beforeTokenTransfer #

($) = payable function
= non-constant function

Issues Checking Status

No.	Title	Status
1.	Unlocked Compiler Version	Low issue
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.	Passed
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

No medium severity issues found.

Low Severity Issues

One low severity issue found.

1. Unlocked Compiler Version.

- **Description**

  The contract utilizes an unlocked compiler version. An unlocked compiler version in the contract's source code permits the user to compile it at or above a particular version. This, in turn, leads to differences in the generated bytecode between compilations due to differing compiler version numbers. This can lead to ambiguity when debugging as compiler-specific bugs may occur in the codebase that would be difficult to identify over a span of multiple compiler versions rather than a specific one.

- **Recommendation**

  It is advisable that the compiler version is alternatively locked at the lowest version possible so that the contract can be compiled. For example, for version ^0.8.15 the contract should contain the following line:

  pragma solidity 0.8.15;

Centralization

Owner Privileges :

- ZeelaNFT Contract:
    - Owner can remove and transfer ownership.

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 but smart contract ownership has been renounced. Following are Admin functions:

- Transferownership
- Renounceownership

Conclusion

Smart contract contains low 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.