HackSafe

Smart Contract Security Audit Report

Metal

June 2022

Security Status

★★★★½

www.hacksafe.io

Audit Details

Audited project
Metal

Deployer address
0xE7184Ea8880D850874D3072b80A273dce10b910f

Client contacts
Metal team

Blockchain
Ethereum

Website
https://metalpay.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 Metal token to perform an audit of smart contracts:

- https://etherscan.io/address/0xF43089366899D83a9f26A773D59ec7eCF30355e#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 21.06.2022

Token Type	: ERC20
Contract name	: MetalToken
Contract address	: 0xF43089366899D83a9f26A773D59ec7eCF30355e
Compiler version	: v0.4.11+commit.68ef5810
Total supply	: 66,588,888
Token Ticker	: MTL
Decimals	: 8
Token Holders	: 12,701
Top 100 token holder's dominance	: 96.15%
Transactions count	: 345,201
Contract deployer address	: 0xE7184Ea8880D850874D3072b80A273dce10b910f
Owner address	: No owner

Social profiles

Twitter Profile	: https://twitter.com/metalpaysme
Facebook Profile	: https://www.facebook.com/metalpaysme/
Telegram Profile	: https://t.me/metalpay
Coinmarketcap profile	: https://coinmarketcap.com/currencies/metal/
Coingecko profile	: https://www.coingecko.com/en/coins/metal

Audit Summary

According to the standard audit assessment, Customer`s solidity smart contracts are “Secure”. This token contract does not contain owner control, which do make it fully decentralized as owner does not 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 2 low and some very low-level issues. These issues are not critical ones.

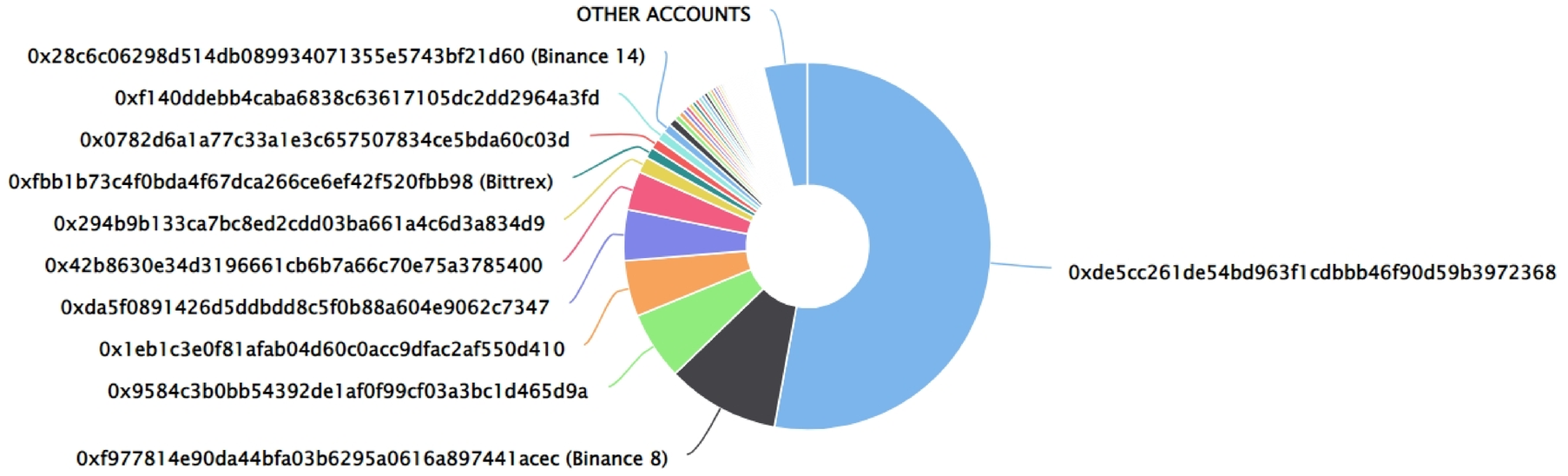
Metal Distribution

The top 100 holders collectively own 96.15% (64,025,448.50 Tokens) of Metal

Token Total Supply: 66,588,888.00 Token | Total Token Holders: 12,701

Metal Top 100 Token Holders

Source: Etherscan.io



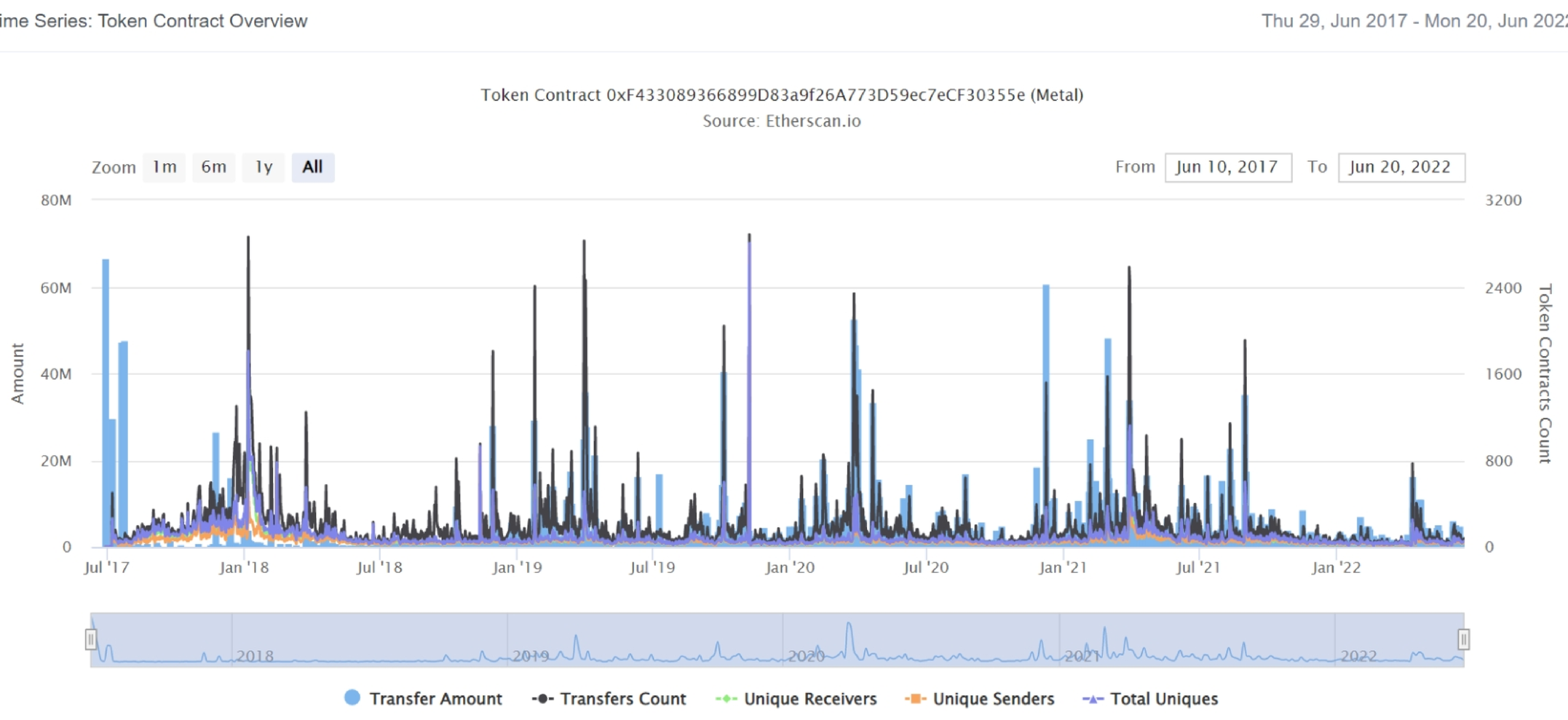
Metal Top 20 Token Holders

(A total of 64,025,448.50 tokens held by the top 100 accounts from the total supply of 66,588,888.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0xde5cc261de54bd963f1cdbbb46f90d59b3972368	35,223,551.36872814	52.8970%
2	Binance 8	6,600,000	9.9116%
3	0x9584c3b0bb54392de1af0f99cf03a3bc1d465d9a	4,000,000	6.0070%
4	0x1eb1c3e0f81afab04d60c0acc9dfac2af550d410	3,283,244.86470517	4.9306%
5	0xda5f0891426d5ddbdd8c5f0b88a604e9062c7347	2,983,288.9989318	4.4802%
6	0x42b8630e34d3196661cb6b7a66c70e75a3785400	2,277,925.96189533	3.4209%
7	0x294b9b133ca7bc8ed2cdd03ba661a4c6d3a834d9	914,942.675223	1.3740%
8	Bittrex	635,168.4823617	0.9539%
9	0x0782d6a1a77c33a1e3c657507834ce5bda60c03d	600,000.00248	0.9011%
10	0xf140ddebb4caba6838c63617105dc2dd2964a3fd	569,532.51323927	0.8553%
11	Binance 14	521,933.79455602	0.7838%
12	0x80fc4c7ac05acb0c2b05866f43ce929a356ed49e	449,242.65173858	0.6747%
13	FTX Exchange	311,622.16031097	0.4680%
14	0x8329d783f96e48d50f0f1f5a3c7f83b73a5c0218	290,351.12922273	0.4360%
15	0x0e0d02eac04df6961a4d7fc4077f84265e41b51b	241,511	0.3627%
16	0x1bed1a65e2b564a42012575f8729e33c32b5f061	231,134.9219	0.3471%
17	0x047e78b584923c8f57587d3d8e7e8a02ca01cfe2	230,637.11690572	0.3464%
18	0x971972218f0e3dc0d55c9a159e44a24372d02850	221,926.05472351	0.3333%
19	0x77d226140e01f9d33a14365e37409c66495e47fe	219,167.63840979	0.3291%
20	Binance 15	215,760.63740466	0.3240%

Metal Distribution

Metal Contract Overview



Contract functions details

+ [Lib] SafeMath
    -[Int] div
    -[Int] sub
    -[Int] add
    -[Int] max64
    -[Int] min64
    -[Int] max256
    -[Int] min256
    -[Int] assert

+ ERC20Basic
    - balanceOf
    - transfer

+ ERC20 (ERC20Basic)
    -allowance
    -transferFrom
    -approve

+BasicToken (ERC20Basic)
    - transfer #
      modifiers: onlyPayloadSize
    - balanceOf

+ StandardToken (BasicToken, ERC20)
    transferFrom
      modifiers: onlyPayloadSize
    approve
    allowance

+ MetalToken (StandardToken)
    MetalToken

($) = 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.	Compiler version too old	Low issue

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

Two 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 ^0.4.11 the contract should contain the following line:

  pragma solidity 0.4.11;

2. Too old compiler version

- **Description**

  Compiler version is old.

- **Recommendation**

  It is advisable that the compiler version is new rather than older one which provide more security and transparency in code.

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.