



# Smart Contract Security Audit Report

---

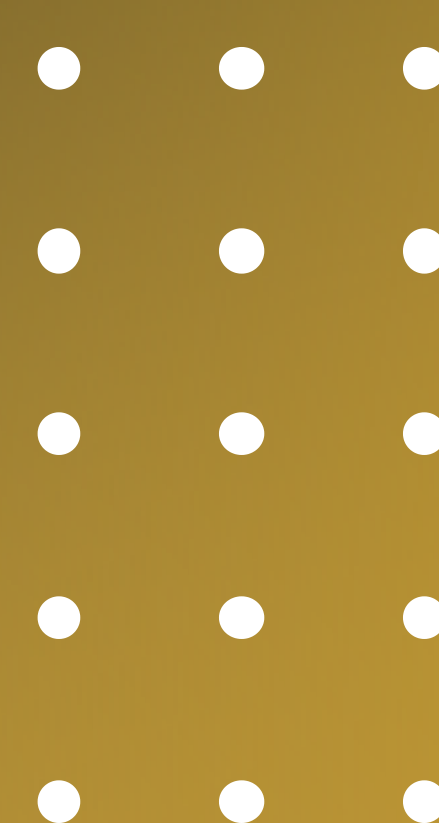
## Prometheus

July 2022

Security Status



[www.hacksafe.io](https://www.hacksafe.io)



# Audit Details



## Audited project

Prometeus



## Deployer address

0xE082e567E35D8930DafEC2FBaa96fd88418718De



## Client contacts

Prometeus team



## Blockchain

Binance Smart Chain



## Website

<https://prom.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 Prometheus to perform an audit of smart contracts:**

- <https://bscscan.com/address/0xaf53d56ff99f1322515e54fdde93ff8b3b7dafd5#code>

**The purpose of the audit was to achieve the**

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

The information in this report should be understood 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

# Contract Details

## Token contract details for 30.07.2022

Token Type	: BEP-20
Contract name	: PROMToken
Contract address	: 0xaF53d56ff99f1322515E54FdDE93FF8b3b7DAFd5
Compiler version	: v0.5.16+commit.9c3226ce
Total supply	: 20,000,000
Token Ticker	: PROM
Decimals	: 18
Token Holders	: 4,424
Top 100 token holder's	: 99.90%
Transactions count	: 94,317
Contract deployer address	: 0xE082e567E35D8930DafEC2FBaa96fd88418718De
Owner address	: 0xe082e567e35d8930dafec2fbaa96fd88418718de

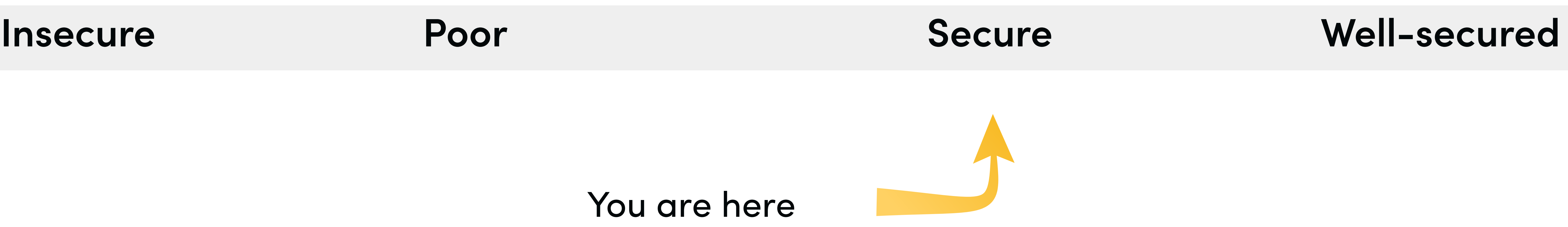


# Social profiles

Twitter Profile	: <a href="https://twitter.com/prom_io">https://twitter.com/prom_io</a>
Telegram Profile	: <a href="https://t.me/prom_io">https://t.me/prom_io</a>
LinkedIN Profile	: <a href="https://www.linkedin.com/company/prom-io/">https://www.linkedin.com/company/prom-io/</a>
Coinmarketcap profile	: <a href="https://coinmarketcap.com/currencies/prom/">https://coinmarketcap.com/currencies/prom/</a>
Coingecko profile	: <a href="https://www.coingecko.com/en/coins/prom/">https://www.coingecko.com/en/coins/prom/</a>

# 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.




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

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



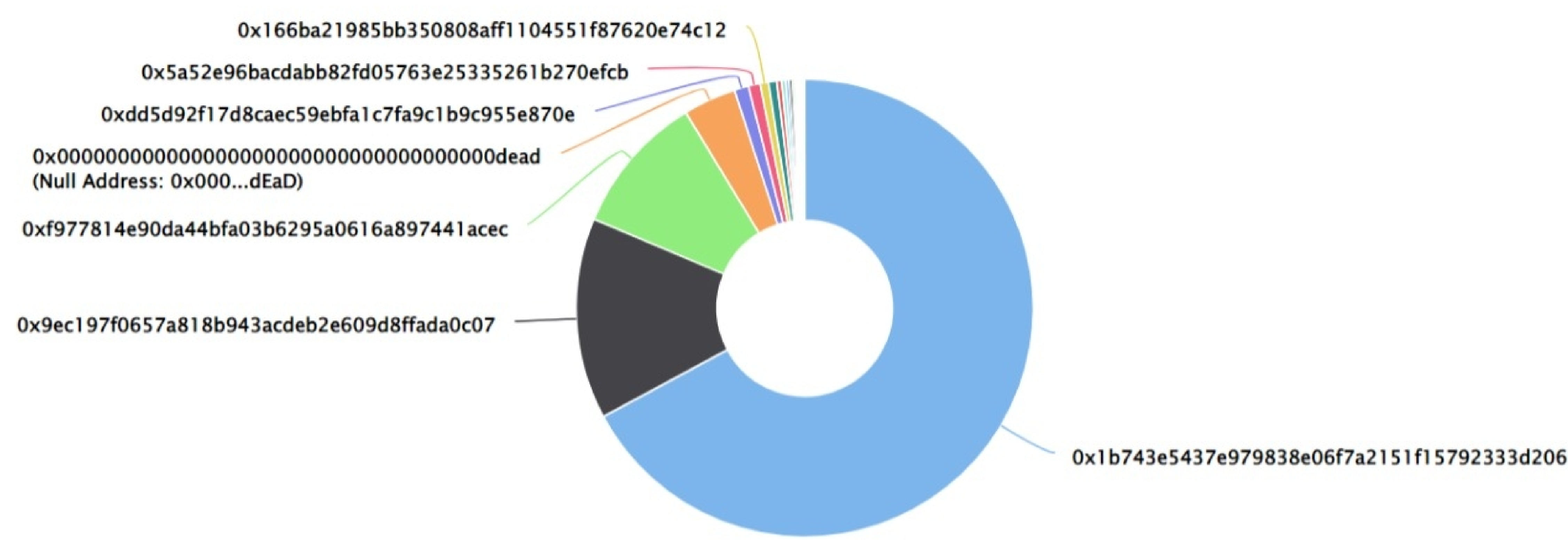
# Prometeus Token Distribution

 The top 100 holders collectively own 99.90% (19,979,549.50 Tokens) of Prometeus

 Token Total Supply: 20,000,000.00 Token | Total Token Holders: 4,430

Prometeus Top 100 Token Holders

Source: BscScan.com

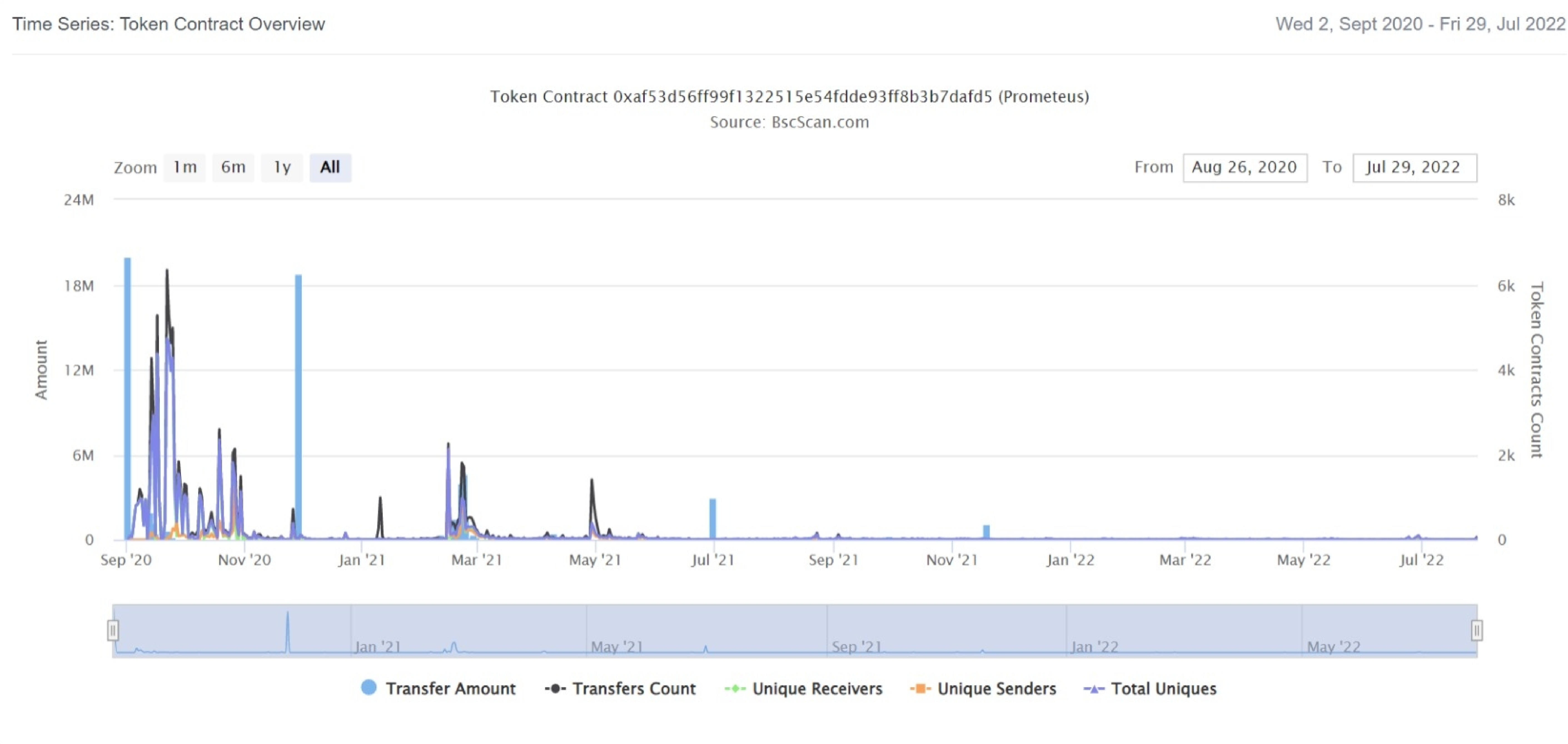


## Prometeus Top 20 Token Holders

Rank	Address	Quantity (Token)	Percentage
1	0x1b743e5437e979838e06f7a2151f15792333d206	13,429,284.556568437385356799	67.1464%
2	📁 0x9ec197f0657a818b943acdeb2e609d8ffada0c07	2,842,532.129225719316373063	14.2127%
3	0xf977814e90da44bfa03b6295a0616a897441acec	1,988,390.125319050005820766	9.9420%
4	Null Address: 0x000...dEaD	750,000	3.7500%
5	0xdd5d92f17d8caec59ebfa1c7fa9c1b9c955e870e	200,000	1.0000%
6	0x5a52e96bacdabb82fd05763e25335261b270efcb	166,215.399999999994179234	0.8311%
7	0x166ba21985bb350808aff1104551f87620e74c12	117,222.286897801691020145	0.5861%
8	Binance: Hot Wallet 6	115,881.887521016386325205	0.5794%
9	0xd33b153dd0318c897f8add0b7e13233de234efd9	68,969.898690091090946778	0.3448%
10	0x4510795377118e75b01f350f5ebb28168173ce9d	55,320.024429198797888553	0.2766%
11	0x944711e619ec67c1328e6dd098740c057bf9101b	50,000	0.2500%
12	0x084c7b0e1878832aeb2690780757bfad1f450043	43,593.848397380945709014	0.2180%
13	0xa63d7cb2e6c8d443f6ae73c90f7f1d04940d9c6a	34,170.37992443	0.1709%
14	Binance: Hot Wallet 7	24,013.87120916	0.1201%
15	0x797eee36a8da66d7ab19e9bc362eed0de2647190	14,569.73628012599404467	0.0728%
16	0x73b725c5a39824323c5aa312f92cbaea3cc6d91d	14,406.974586963894509293	0.0720%
17	0x97c4a8c46a746c722710420ed48bf048022ce6b8	10,625.296	0.0531%
18	0xa8177b35cea28114945efa85d2fa09f45e575ec1	10,043.89697	0.0502%
19	0xaf4b2a9c73d38126bb96739ac605e9d5d2348696	9,756.80899281442763	0.0488%
20	0xefaac74c4f253a874f65b7740a2ca30ee752d784	6,291.20056978	0.0315%

# Prometheus Token Distribution

## Prometheus Contract Overview





# Contract functions details

## + [Int] IBEP20

- [Ext] totalSupply
- [Ext] decimals
- [Ext] symbol
- [Ext] name
- [Ext] getOwner
- [Ext] balanceOf
- [Ext] transfer
- [Ext] allowance
- [Ext] approve
- [Ext] transferFrom

## + Context

- [Int] \_msgSender
- [Int] \_msgData

## + [Lib] SafeMath

- [Int] add
- [Int] sub
- [Int] sub
- [Int] mul
- [Int] div
- [Int] div
- [Int] mod
- [Int] mod

## + Ownable (Context)

- [Int] <constructor>
- [Pub] owner
- [Pub] renounceOwnership #
  - modifiers: onlyOwner
- [Pub] transferOwnership
  - modifiers: onlyOwner
- [Int] \_transferOwnership #

## + PROMToken (Context, IBEP20, Ownable)

- [Pub] <constructor> #
- [Ext] getOwner
- [Ext] decimals
- [Ext] symbol

# Contract functions details

- [Ext] name
- [Ext] totalSupply
- [Ext] balanceOf
- [Ext] transfer #
- [Ext] allowance
- [Ext] approve #
- [Ext] transferFrom #
- [Pub] increaseAllowance
- [Pub] decreaseAllowance
- [Int] \_transfer #
- [Int] \_approve #

(\$) = 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.	Passed
19.	Incorrect Naming State Variable	Passed
20.	Too old version	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

One low severity issue found.

### 1. Too old compiler version

- **Description**

The contract has been deployed using very much old solidity compiler version.

- **Recommendation**

The contract deployment should be done using the any of the newest solidity compiler versions.

# Centralization

## Owner privileges:

- Prometheus Contract:
  - Owner can transfer and renounce 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 as smart contract ownership has not been renounced. Following are Admin functions:

- Renounceownership
- Transferownership



# 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.