



Smart Contract Security Audit Report

Rally

May 2022

Security Status



www.hacksafe.io



Audit Details



Audited project

Rally



Deployer address

0x80D225fE717faEfD206Ac824Eb93D68fc326604c



Client contacts

Rally team



Blockchain

Ethereum



Website

<https://rally.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.

Background

HeckSafe was commissioned by Rally to perform an audit of smart contracts:

- <https://etherscan.io/address/0xf1f955016EcbCd7321c7266BccFB96c68ea5E49b#code>

Contract Details

Token contract details for 12.05.2022

Contract name	: RallyToken
Contract address	: 0xf1f955016EcbCd7321c7266BccFB96c68ea5E49b
Total supply	: 15,000,000,000
Token Ticker	: RLY
Decimals	: 18
Network	: Etherscan
Token Holders	: 9,303 addresses
Transactions count	: 238,312
Contract deployer address	0x80D225fE717faEfD206Ac824Eb93D68fc326604c

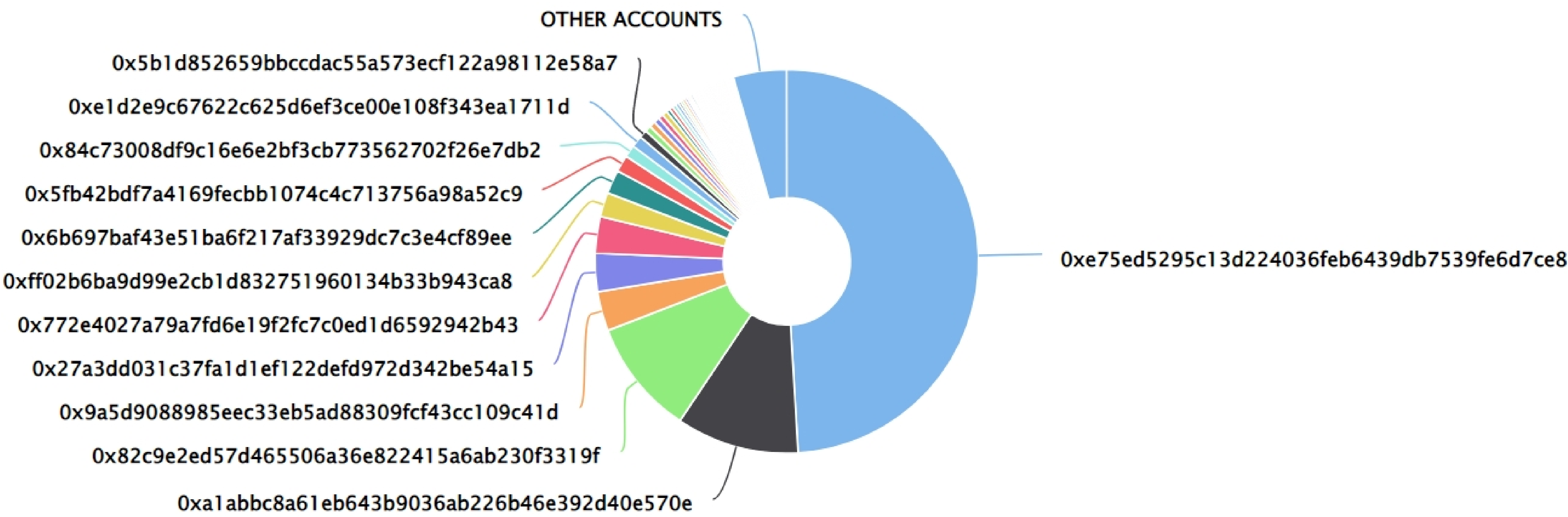
Rally Token Distribution

 The top 100 holders collectively own 95.51% (14,326,412,502.76 Tokens) of Rally











 Token Total Supply: 15,000,000,000.00 Token | Total Token Holders: 9,303

Rally Top 100 Token Holders

Source: Etherscan.io



Rally Top 20 Token Holders

Rank	Address	Quantity (Token)	Percentage
1	 0xe75ed5295c13d224036feb6439db7539fe6d7ce8	7,360,400,974.83845356	49.0693%
2	 0xa1abbc8a61eb643b9036ab226b46e392d40e570e	1,551,911,463	10.3461%
3	 0x82c9e2ed57d465506a36e822415a6ab230f3319f	1,457,802,873.7	9.7187%
4	0x9a5d9088985eec33eb5ad88309fcf43cc109c41d	500,000,000	3.3333%
5	0x27a3dd031c37fa1d1ef122defd972d342be54a15	484,146,718	3.2276%
6	 0x772e4027a79a7fd6e19f2fc7c0ed1d6592942b43	460,000,001	3.0667%
7	0xff02b6ba9d99e2cb1d832751960134b33b943ca8	304,722,873	2.0315%
8	0x6b697baf43e51ba6f217af33929dc7c3e4cf89ee	302,095,333	2.0140%
9	 0x5fb42bdf7a4169fecbb1074c4c713756a98a52c9	210,568,800	1.4038%
10	0x84c73008df9c16e6e2bf3cb773562702f26e7db2	154,274,153	1.0285%
11	 0xe1d2e9c67622c625d6ef3ce00e108f343ea1711d	145,924,796.101629115080059902	0.9728%
12	0x5b1d852659bbccdac55a573ecf122a98112e58a7	100,000,000	0.6667%
13	0xef159cec7408a107bc0d6f4209ad6f51bbdd4a78	77,631,699	0.5175%
14	 0xc93e3583c6dc8dfd59b974d704b5b79f02e210a	75,720,294.30154644	0.5048%
15	0x4dac00bfc49cff35319e801dca6b74ce5f9a8f23	71,864,894	0.4791%
16	 0x615657cd4a78c530f86e84c13d7504fc80a7496a	64,251,071	0.4283%
17	 Sablier v1.1	63,147,185.503003968680653946	0.4210%
18	0x4336435cd0c48874ba7c64faa629662d1c3e1026	47,727,270	0.3182%
19	 Uniswap V2: RLY	46,327,211.165623152087739484	0.3088%
20	0xf9f644f1bb10983d2b7fa80f1f1634d2b4f50f0a	45,578,103	0.3039%

Contract functions details

+ Context

- [Int] _msgsender
- [Int] _msgdata

+ [Int] IERC20

- [Ext] totalSupply
- [Ext] balanceOf
- [Ext] transfer#
- [Ext] allowance
- [Ext] approve#
- [Ext] transferFrom#

+ [Lib] SafeMath

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

+ [Lib] Address

- [Int] isContract
- [Int] sendValue
- [Int] functionCall
- [Int] functionCall
- [Int] functionCallWithValue
- [Int] functionCallWithValue
- [Pvt] _functionCallWithValue

+ [Int] IERC20Metadata (IERC20)

- [Ext] name
- [Ext] symbol
- [Ext] decimals

Contract functions details

+ERC20 (Context, IERC20, IERC20Metadata)

- [Pub]<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] _setupDecimals#
- [Int] _spendAllowance#

+ RallyToken (ERC20)

- [Pub] <constructor>#

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.	Low issue
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

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 issues 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 v0.6.2 the contract should contain the following line:

```
pragma solidity 0.6.2;
```

2. Scoping and Declarations.

Unused function.

- **Description**

The `sendValue` , `functionCall` , `functionCallWithValue`, `_functionCallWithValue` function does nothing.

- **Location:**

Line number: 313, 339, 349, 364, 374, 379.

- **Recommendation:**

We advise to remove unused code to practise clean code style and it will save some computational gas too.

Conclusion

Smart contract contains low severity issues!

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.