Audit of smart contracts ricetron

Revision 1 dated 01.15.2021

Contents

Brief information

Project: ricetron.com
Network: TRON
Compiler version: 0.5.12
Optimization: enabled
The audit date: 01.15.2021

Information

The contract code was reviewed and analyzed for vulnerabilities, logical errors, and the developers' exit scams possibility. This work was carried out concerning the project source code provided by the customer.

During the audit, no errors were found that could affect the security of the funds.

The detected problems full list can be found below.

General conclusion

As a result of the audit, no errors were found that affect the security of users' funds on the contract. No obvious signs of an exit scam were found.

Telescr.in guarantees the ricetron contracts security and performance.

Liability disclaimer

The telescr.in team within this audit framework is not responsible for the developers or third parties' actions on the platforms associated with this project (websites, mobile applications, and so on). The audit confirms and guarantees only the smart contract correct functioning in the revision provided by the project developers (check the revision).

Confirmed by digital signature

Aggregated data

The Contract analysis was performed using the following methods:

- Static analysis
  - o Checking the code for common errors leading to the most common vulnerabilities
- Dynamic analysis
  - o The Contract launching and carrying out the attacks various kinds to identify vulnerabilities
- Code Review

Received data

Recommendation	Type	Priority	Occurrence probability
Borderline states	Remark	Low	Low
Code quality	Notice	Low	Low

A. Errors

Not found

B. Remarks

1.   Borderline states

If the spenderAllowance value is equal to the maximum value for uint256, then the value of available tokens for transferFrom will not decrease.

C. Warnings

Not found

D. Notice

1. Code quality

It is better to give all percentages relative to one common quotient. So the percentage value for each calculation will be more readable.

Application. Error classification

Priority	
informational	This question is not directly related to functionality but may be important to understand.
low	This question has nothing to do with security, but it can affect some behavior in unexpected ways.
Average	The problem affects some functionality but does not result in an economically significant user funds loss.
high	This issue can result in the user funds loss.
Probability	
Low	It is unlikely that the system is in a state in which an error could occur or could be caused by any party.
Average	This problem may likely arise or be caused by some party.
high	It is highly likely that this problem could arise or could be exploited by some parties.

Application. Digital bytecode print

The audit was carried out for the code certain version on the compiler version 0.7.0 with the optimization enabled.

To check the contract bytecode for identity to the one that was analyzed during the audit, you must:
1. Get contract bytecode (in any block explorer)
2. [Get SHA1 from bytecode string](#)
3. Compare with reference in this report

Sha1 from bytecode:

        6aa7664b2c83d71f3db6018f55e1725d0d0d6cfb

Sha1 from bytecode (non-metadata):

        2cdd8b345c6abe0adbe8691f2a4846fb3a9298d1

Contract address:

        TPBRQ9AqBpKctTyJ3RmAcobW8cNAFXF4SB

[Check the digital print](#)

Application. Digital bytecode print

Application. Signature of the audit report

```
{
  "address": "0x505ade8cea4db608250e503a5e8d4cb436044d2e",
  "msg": "As a result of the audit, no errors were found that affect the security of users' funds on the contract. No obvious signs of an exit scam were found. Telescr.in guarantees the ricetron contracts security and performance. Sha1 from bytecode: 6aa7664b2c83d71f3db6018f55e1725d0d0d6cfb Sha1 from bytecode (non-metadata): 2cdd8b345c6abe0adbe8691f2a4846fb3a9298d1 Contract address:  TPBRQ9AqBpKctTyJ3RmAcobW8cNAFXF4SB",
  "sig": "0x605227cda92dc960e54515bc917506a69596c873b12531c261a813f0af81a1d4481f4b6f0cf615cd9b0f73a5e60c73d6bae52c5269b04feebc24aca687ecc2c21c",
  "version": "3"
}
```

[Check the signature]