Telescr.in

Audit of smart contracts CyberTron

Revision 1 dated 01.15.2021

Telescr.in

Contents

Brief information

Project: cybertron.expert
Network: TRON
Compiler version: 0.5.4
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 affect the safety of funds and the performance of the contract.

The detected problems full list can be found below.

General conclusion

As result of the audit, no errors were found that could affect the security of user funds. No obvious signs of an exit scam were found. However, an investor should pay attention to deposit and withdrawal operations, in which part of the actions goes through manual execution.

Telescr.in guarantees the safety and performance of the CyberTron contract.

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.

Confirmed by digital signature

Aggregated data

The Contract analysis was performed using the following methods:

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

Received data

Recommendation	Type	Priority	Probability of occurrence
Overflow Probability	Warning	Average	Low
Unknown token	Warning	Average	High
Manual execution	Warning	High	High
Manual execution [2]	Warning	High	High

A. Errors

Not found.

B. Remarks

Not found.

C. Warnings

1. Overflow Probability

The SafeMath library is not used for all calculations in the contract. Even though the overflow probability during calculations in this method is minuscule, the general recommendation is to use SafeMath for such calculations.

2. Unknown token

10% of the withdrawal amount goes to the purchase of FRAG tokens, the source code of which is closed and has not been audited.

3. Manual execution

The owners of the contract state that 1% out of 2 goes to keep the contract working, and another 1% goes to the liquidity of the deflationary token and the right to receive a lifetime income as a percentage of your investment from the commission on trading in DEX. However, in the logic of the contract, only the collection of these percentages in the deposit () function for a separate address is implemented. The rest of these actions with these 2% occurs outside the algorithm of this contract, manually by the project administrators. Their performance is not guaranteed for this audit.

4. Manual execution [2]

In the withdraw () function during the withdrawal process - 10% of the amount goes to the "participant's personal insurance", while the contract only transfers this amount to a separate address, further actions with this amount are done manually by the project administrators and their implementation is not guaranteed within this audit.

D. Notice

Not found.

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 0.5.12 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:

ee39efc54175f6d5cbf6dc13bd40c1ca7e9311aa

Sha1 from bytecode (non-metadata):

3323d84e79c82f7bc30f73bb7bcc806f65e206a6

Contract address:

TAf4XKjKR3pvvtdZAgsYMtEG7uCF5tvCKv

Check the digital print

Application. Signature of the audit report


```
{
  "address": "0x505ade8cea4db608250e503a5e8d4cb436044d2e",
  "msg": "As result of the audit, no errors were found that could affect the security of user funds. No obvious signs of an exit scam were found. However, an investor should pay attention to deposit and withdrawal operations, in which part of the actions goes through manual execution. Telescr.in guarantees the safety and performance of the CyberTron contract. Sha1 from bytecode: ee39efc54175f6d5cbf6dc13bd40c1ca7e9311aa Sha1 from bytecode (non-metadata): 3323d84e79c82f7bc30f73bb7bcc806f65e206a6 Contract address:  TAf4XKjKR3pvvtdZAgsYMtEG7uCF5tvCKv",
  "sig": "0xcce9c80d80954cd0711042b435a87ca443dcb0ee9e705e9d63b110c34addcb6141838587d429c99de1ceea7c53c19b296c22beaac27d20d423f55e03f5a671641c",
  "version": "3"
}
```


Check the signature