May 2021

Security Audit

TronCore

TAzg2Z3T94DTvpjBAvVepAUN12y2KyUaNM

www.grox.solutions

CRITICAL ISSUES (critical, high severity): **0**

Bugs and vulnerabilities that enable theft of funds, lock access to funds without possibility to restore it, or lead to any other loss of funds to be transferred to any party; high priority unacceptable bugs for deployment at mainnet; critical warnings for owners, customers or investors.

ERRORS, BUGS AND WARNINGS (medium, low severity): **0**

Bugs that can trigger a contract failure, with further recovery only possible through manual modification of the contract state or contract replacement altogether; Lack of necessary security precautions; other warnings for owners and users.

OPTIMIZATION POSSIBILITIES (very low severity): **1**

Possibilities to decrease cost of transactions and data storage of Smart-Contracts.

NOTES AND RECOMMENDATIONS (very low severity): **2**

Tips and tricks, all other issues and recommendations, as well as errors that do not affect the functionality of the Smart-Contract.

Conclusion:

In the TronCore Smart-Contract were found no vulnerabilities and no backdoors. The code was manually reviewed for all commonly known and more specific vulnerabilities.

So TronCore Smart-Contract is safe for use in the main network.

https://tronscan.org/#/contract/TAzg2Z3T94DTvpjBAvVepAUN12y2KyUaNM/code

AUDIT RESULT:

Optimization possibilities

1. Recording statistical parameters in the blockchain (very low severity):

List of statistical parameters that also increase the cost of transactions and increase the amount of data stored in the blockchain:

```
uint24 public totalUsers;
uint256 public totalInvested;
uint256 public totalWithdrawn;
uint256 public totalDeposits;
uint256 refback;
uint24[5] levels;
```

Recommendation: use events and log this information instead of writing it to the blockchain.

Note: this comment doesn't affect the main functionality of the smart-contract.

Notes

2. Loops over dynamic variables (very low severity):

In the `withdraw, getUserDividends`, `getUserAvailable`, `getUserTotalDeposits`, `getUserDividendsWithdrawn, getUserTotalRefback` functions loops unrestrictedly grow as the number of deposits increases. If one creates a large number of parallel deposits from a single wallet, it can lead to an excessive increase of the transaction cost.

Note: maximum amount of deposits from single account - 60.

3. Closing the last payment (very low severity).

If the last user who leaves the project has a payout greater than the smart-contract balance, he will receive the entire available balance, but it will be recorded that the entire payout was closed.

Note: this comment is not critical, since after the smart contract balance is empty, it is unlikely that the contract will be used again. So it makes sense for last user to get at least something.

Independent description of the smart-contract functionality:

The TronCore contract provides the opportunity to invest any amount in TRX (from 100 TRX) in the contract and get a 200% return on investment, if the contract balance has enough funds for payment.

Dividends are paid from deposits of users (Ponzi scheme).

It is allowed to participate in the project only from usual wallet (not smart-contract nor externally owner address).

You can create a Deposit by calling the “invest” function and attaching the required amount of TRX to the transaction (from 100 TRX inclusive).

Each subsequent Deposit is kept separately in the contract, in order to maintain the payment amount for each Deposit.

The daily percentage for user dividends starts from 1% and depends on the following factors:

- Contract Bonus: every 1 000 000 TRX on the balance of smart-contract first +0.25% until 3% (12 millions), then +0.20% until 6% (up to 27 millions), then +0.15% until 9% (up to 47 millions), then +0.10% until 12% maximum (up to 77 millions). Once reached contract bonus cannot decrease.

- Hold Bonus: every 24 hours of non-withdrawal of dividends from the smart-contract +0.10% first week (until 0.7%), then +0.13% second week (until 1.61%), then +0.15% third week (until 2.66%), then +0.17% last week (until 3.85% maximum). If user creates new deposits the percent keeps growing.

Total maximum user daily percent is 16.85%.

All dividends are calculated at the moment of request and available for withdrawal at any time.

Withdrawal is performed by calling the “withdraw” function from the address the Deposit was made.

Contract owners fee: part of the invested funds is sent to two addresses:

TJLuhnrACJCnYnENrMtRz9hdwPgTPaDuKy - 8.8% (marketing address)
TEVK9mjkC1KsvCXtGrJvsrBGDtYndKs9yE - 2.2% (project address)

There is five-level referral program: in the “invest” function, one can specify the address of the referrer.

As a result, the referrer (upline) will get possibility to withdraw share of the investor's Deposit according to the following table (the line depends on Ref’s deposits: sum of deposits of referrals):

Referrer level	1	2	3	4	5
Ref’s deps < 5M TRX	3	1	1	0	0
Ref’s deps > 5M TRX	4	2	1	0.5	0
Ref’s deps > 15M TRX	5	2	1	0.5	0.5

Also there is a revenue refbonus: after each first withdrawal of dividends - direct referrer gets percent of that dividends according with table (for 1 level).

Requirements for the referrer: you can not specify your own wallet as a referrer, as well as a wallet that does not have at least one contribution in the smart contract. If wrong referrer is provided, no referrer is set.

The referrer is specified once at the time of the first deposit and is assigned to the user without the possibility of changing. From each subsequent Deposit, the referrer will get his percents.

Any user that has at least one contribution in the project can specify his own ‘refBackPercent’ - share of the referral bonus that will be returned to his direct referral (only 1 referral level).

To set refBackPercent user must call ‘setRefBackPercent’ function with percent parameter with 2 decimals (means 1% = 100, 100% = 10000).

The smart-contract has limits of total Invested value per 24 hours (since start date):

First 5 days - 1M TRX, next 5 days - 2M TRX, next 5 days - 4M, next 5 days - 5M, next 5 days - 10M and then 25M every day further.

The contract contains statistical functions that do not require sending transactions:

1. `getContractBalance` – balance of the smart-contract.
2. `getContractBalanceRate` – current daily percent of the project.
3. `getCurrentDayAvailable` – current available daily limit to invest.
4. `isActive` – whether the user has active deposits.
5. `getUserCheckpoint` – UNIX time of last action of user.
6. `getUserAmountOfDeposits` - the number of user deposits.
7. `getUserTotalDeposits` – the sum of each deposits of the user.
8. `getUserPercentRate` - the current percentage for the user. (% + 2 decimals)
9. `getUserHoldBonus` - the current hold bonus for the user. (% + 2 decimals)
10. `getUserAvailableBalanceForWithdrawal` - total available amount to withdraw.
11. `getUserTotalWithdrawn` - total withdrawn amount by user.
12. `getUserDividends` – available dividends to withdraw.
13. `getUserDividendsWithdrawn` – withdrawn dividends by user.
14. `getUserDividendsSum` – total dividends (available + withdrawn).
15. `getUserReferralBonus` – available RefBonus to withdraw.

https://tronscan.org/#/contract/TAzg2Z3T94DTvpjBAvVepAUN12y2KyUaNM/code

The contract contains statistical functions that do not require sending transactions:

16. `getUserRefBonusWithdrawn` – withdrawn RefBonus by user.
17. `getUserTotalRefBonus` – total RefBonus (available + withdrawn).
18. `getUserRevenueBonus` – available RevenueBonus to withdraw.
19. `getUserRevenueBonusWithdrawn` – withdrawn RevenueBonus by user.
20. `getUserTotalRevenueBonus` – total RevenueBonus (available + withdrawn).
21. `getUserRefback` – available Refback to withdraw.
22. `getUserRefbackWithdrawn` – withdrawn Refback by user.
23. `getUserTotalRefback` - total Refback (available + withdrawn).
24. `getUserReferrer` - user's referrer (upline).
25. `getUserRefbackPercent` - refBackPercent set by user.
26. `getUserReferrerRefBackPercent` - refBackPercent of user's referrer.
27. `getUserDownlineCount` – amount of referrals on each level.
28. `getUserTotalRefInvested` – total invested amount by all referrals.
29. `getUserReferralPercent` – referral percent for user per each level.

May 2021

Disclaimer:

This audit is not a call to participate in the project and applies only to the Smart-Contract code at the specified address.

Do not forget that you are doing all financial actions at your own risk, especially if you deal with high-risk projects.

Warning:

Beware of fake audits.
All official info available on 3 resources only:

Website: **www.grox.solutions**
Telegram: **www.t.me/groxsolutions (@groxsolutions)**
YouTube: **www.youtube.com/c/groxsolutions**

If you have any questions or are interested in developing/auditing of Smart-Contracts, please contact us and we will consult you.

Telegram: **www.t.me/gafagilm (@gafagilm)**
E-mail: **info@grox.solutions**