Tryon

Security Code Review

https://twitter.com/VidarTheAuditor - 22 March 2021

Overview

Project Summary

Project Name	Tryon
Description	
Platform	**Binance Smart Chain**, Solidity
Contracts	https://bscscan.com/address/0x4401b7de9c55622fd132057526d8d975ce241114#code

Executive Summary

Binance Smart Chain BEP20 compatible contract was provided.

We have run extensive static analysis of the codebase as well as standard security assessment utilising industry approved tools.

We have not found any critical vulnerabilities arising from a third party involvement.

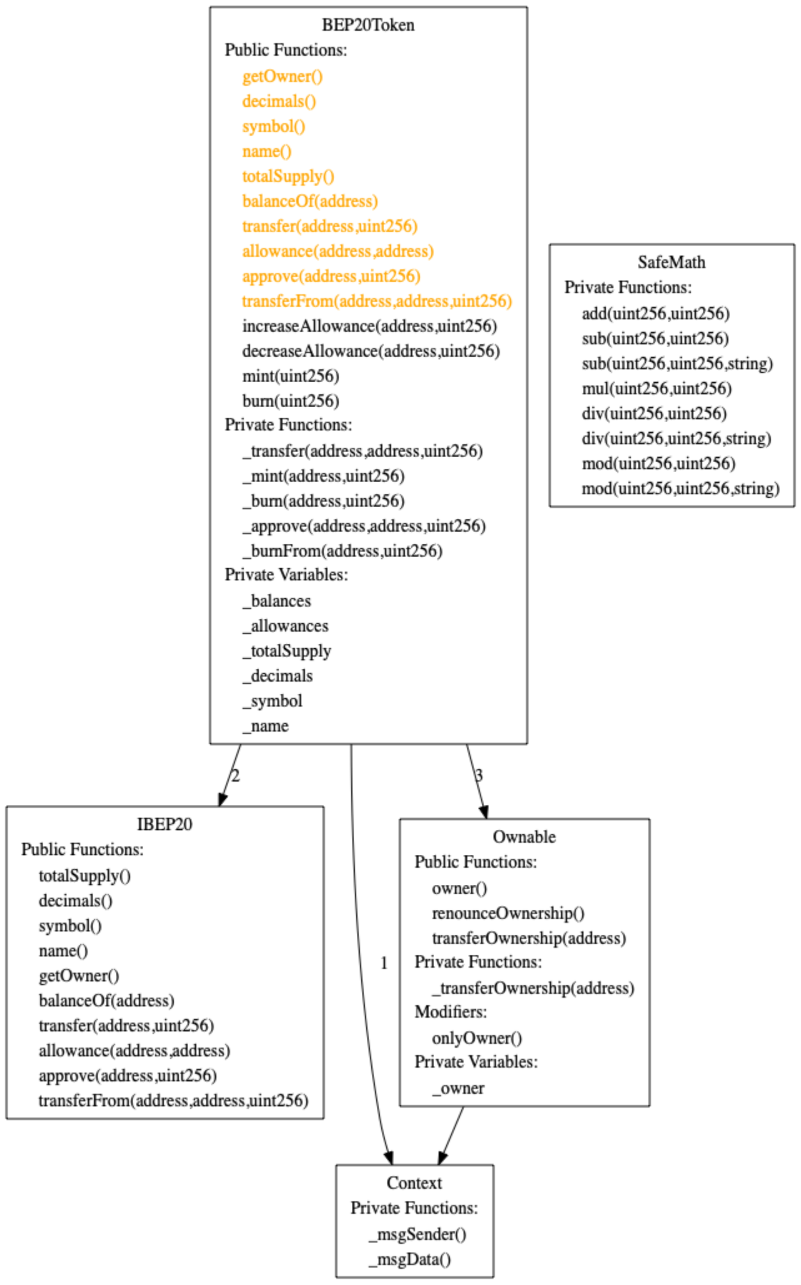
We have found a critical risk in the BEP20 contract that allows contract owner to mint unlimited number of tokens.

More information and recommendations can be found in the Deployment section.

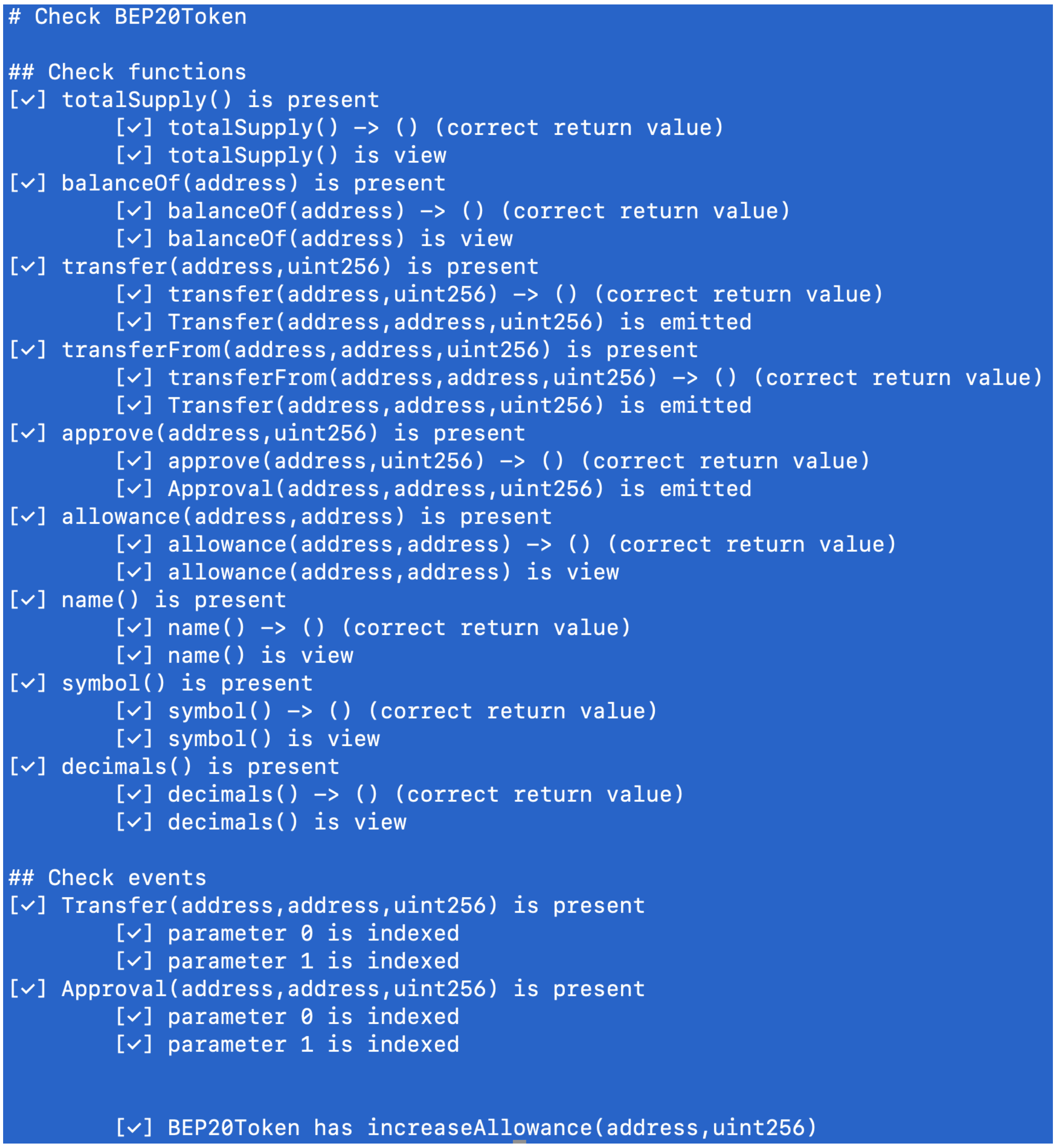
Disclaimer: The analysis did not include any tokenomics analysis (e.g. APY rates etc).

Architecture & Standards

Please find below the calling architecture of the reviewed contracts.



TRYON Token contract is fully BEP20 compatible.

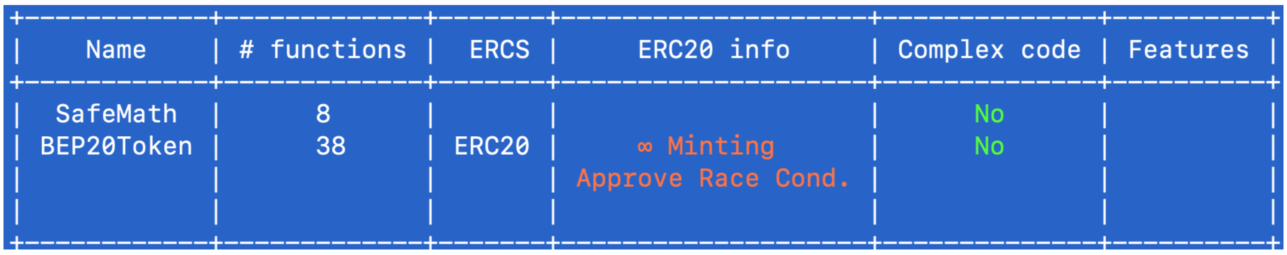


```
# Check BEP20Token

## Check functions
[✓] totalSupply() is present
        [✓] totalSupply() -> () (correct return value)
        [✓] totalSupply() is view
[✓] balanceOf(address) is present
        [✓] balanceOf(address) -> () (correct return value)
        [✓] balanceOf(address) is view
[✓] transfer(address,uint256) is present
        [✓] transfer(address,uint256) -> () (correct return value)
        [✓] Transfer(address,address,uint256) is emitted
[✓] transferFrom(address,address,uint256) is present
        [✓] transferFrom(address,address,uint256) -> () (correct return value)
        [✓] Transfer(address,address,uint256) is emitted
[✓] approve(address,uint256) is present
        [✓] approve(address,uint256) -> () (correct return value)
        [✓] Approval(address,address,uint256) is emitted
[✓] allowance(address,address) is present
        [✓] allowance(address,address) -> () (correct return value)
        [✓] allowance(address,address) is view
[✓] name() is present
        [✓] name() -> () (correct return value)
        [✓] name() is view
[✓] symbol() is present
        [✓] symbol() -> () (correct return value)
        [✓] symbol() is view
[✓] decimals() is present
        [✓] decimals() -> () (correct return value)
        [✓] decimals() is view

## Check events
[✓] Transfer(address,address,uint256) is present
        [✓] parameter 0 is indexed
        [✓] parameter 1 is indexed
[✓] Approval(address,address,uint256) is present
        [✓] parameter 0 is indexed
        [✓] parameter 1 is indexed


        [✓] BEP20Token has increaseAllowance(address,uint256)
```

Findings

Number of contracts: 5 (including inherited ones)

Use: SafeMath



```
+------------+-------------+-------+--------------------+--------------+-----------+
|    Name    | # functions | ERCS  |     ERC20 info     | Complex code | Features  |
+------------+-------------+-------+--------------------+--------------+-----------+
|  SafeMath  |      8      |       |                    |      No      |           |
| BEP20Token |      38     | ERC20 |     ∞ Minting      |      No      |           |
|            |             |       | Approve Race Cond. |              |           |
|            |             |       |                    |              |           |
+------------+-------------+-------+--------------------+--------------+-----------+
```

TRYON token has minting capabilities, it may posses significant risks as owner can create unlimited number of tokens.

See Deployment section for more details.

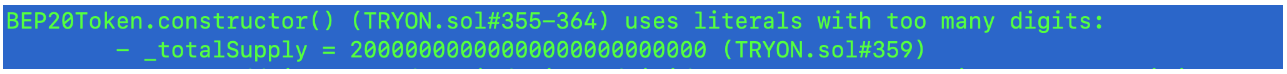
Static Analysis Findings

High issues: None

Medium issues: None

Low/Informational issues:

Too many digits:



```
BEP20Token.constructor() (TRYON.sol#355-364) uses literals with too many digits:
        - _totalSupply = 2000000000000000000000000 (TRYON.sol#359)
```

Literals with many digits are difficult to read and review.

Dynamic Tests

We have run fuzzing/property-based testing of Solidity smarts contracts. It was using sophisticated grammar-based fuzzing campaigns based on a contract ABI to falsify user-defined predicates or Solidity assertions.

There were also dynamic tests run on EVM byte code to detect common vulnerabilities including integer underflows, owner-overwrite-to-Ether-withdrawal, and others.

We have not found any significant risks using dynamic tests tools.

Deployment & Contract Ownership

The contracts are currently deployed on BSC Mainnet:

- https://bscscan.com/address/0x4401b7de9c55622fd132057526d8d975ce241114#code

Risks

- **The minting functionality exists in the BEP20 contract, that is a significant risk for any token holder.**
- As the pre-sale has not been done and there is no specific contract for this, it has to be validated that the liquidity is properly added to the based on the team information.

Recommendations

- If the minting functionality is needed for the system the ownership of the contract has to be protected. It can be done by:
  - Deploying a governance on top of the ownership of all contracts.
    - It could be a proper multi-sig controlled setup or full Compound like governance system - https://medium.com/compound-finance/compound-governance-5531f524cf68
  - Move ownership of the contract to the TimelockController contract
- If the minting functionality is not needed, the ownership of the contract should be renounced to 0x0 address.
- After the pre-sale the liquidity has to be added to the DEX pool (e.g. PancakeSwap)
- The LP tokens should be burnt or locked in the specialised contract.

Disclaimer

The information appearing in this report is for general purposes only and is not intended to provide any legal security guarantees to any individual or entity. As one review is not enough to provide 100% security against any attacks or bugs, it is advisable to conduct more reviews or/and audits.

The report does not provide personalised investment advice or recommendations, especially does not provide advice to conclude any transactions and it does not provide investment, financial, legal or tax advice.

We are not responsible or liable for any loss which results from the report.

The report should not be considered as an investment advice.