QuillAudits

Audit Report
September, 2021

For

PADD.FINANCE

Contents

Scope of the Audit

The scope of this audit was to analyze and document the PADD Token smart contract codebase for quality, security, and correctness.

Checked Vulnerabilities

We have scanned the smart contract for commonly known and more specific vulnerabilities. Here are some of the commonly known vulnerabilities that we considered:

- Re-entrancy
- Timestamp Dependence
- Gas Limit and Loops
- DoS with Block Gas Limit
- Transaction-Ordering Dependence
- Use of tx.origin
- Exception disorder
- Gasless send
- Balance equality
- Byte array
- Transfer forwards all gas
- BEP20 API violation
- Malicious libraries
- Compiler version not fixed
- Redundant fallback function
- Send instead of transfer
- Style guide violation
- Unchecked external call
- Unchecked math
- Unsafe type inference
- Implicit visibility level

Techniques and Methods

Throughout the audit of smart contract, care was taken to ensure:

- The overall quality of code.
- Use of best practices.
- Code documentation and comments match logic and expected behaviour.
- Token distribution and calculations are as per the intended behaviour mentioned in the whitepaper.
- Implementation of BEP-20 token standards.
- Efficient use of gas.
- Code is safe from re-entrancy and other vulnerabilities.

The following techniques, methods and tools were used to review all the smart contracts.

Structural Analysis
In this step, we have analysed the design patterns and structure of smart contracts. A thorough check was done to ensure the smart contract is structured in a way that will not result in future problems.

Static Analysis
Static analysis of smart contracts was done to identify contract vulnerabilities. In this step, a series of automated tools are used to test the security of smart contracts.

Code Review / Manual Analysis
Manual analysis or review of code was done to identify new vulnerabilities or verify the vulnerabilities found during the static analysis. Contracts were completely manually analysed, their logic was checked and compared with the one described in the whitepaper. Besides, the results of the automated analysis were manually verified.

Gas Consumption
In this step, we have checked the behaviour of smart contracts in production. Checks were done to know how much gas gets consumed and the possibilities of optimization of code to reduce gas consumption.

Tools and Platforms used for Audit
Mythril, Slither, SmartCheck, Surya, Solhint.

Issue Categories

Every issue in this report has been assigned to a severity level. There are four levels of severity, and each of them has been explained below.

Risk-level	Description
High	A high severity issue or vulnerability means that your smart contract can be exploited. Issues on this level are critical to the smart contract's performance or functionality, and we recommend these issues be fixed before moving to a live environment.
Medium	The issues marked as medium severity usually arise because of errors and deficiencies in the smart contract code. Issues on this level could potentially bring problems, and they should still be fixed.
Low	Low-level severity issues can cause minor impact and or are just warnings that can remain unfixed for now. It would be better to fix these issues at some point in the future.
Informational	These are four severity issues that indicate an improvement request, a general question, a cosmetic or documentation error, or a request for information. There is low-to-no impact.

Number of issues per severity

Type	High	Medium	Low	Informational
Open	0	0	0	0
Acknowledged	0	0	0	0
Closed	1	0	0	0

Introduction

During the period of **Sept 20, 2021 to Sept 21, 2021** - QuillAudits Team performed a security audit for PADD smart contracts.

The code for the audit was taken from the following official link:

V	Date	Transaction hash	Note
Version 1	21-09-21	https://explorer.kcc.io/en/address/0x1e2fbb76c8daf5a0a8f91388bac09511f3d7ac62	Version 1
Version 2	21-09-21	https://github.com/Padd-Finance/PADD-COIN/blob/master/contracts/StandardToken.sol 6a197ae09d00eaf49198003c3f91f225d383ee37	Version 1.1

Issues Found

High severity issues

1. Unused Internal Function

Description

_burnFrom() function in the contract is never used. As this is an unused internal function we can reduce the deployment cost. Also as per the logic 5% of the tokens will be burned, but as burn function is internal then it will not be possible to burn the tokens using it from external calls

4. 5% of the revenue generated from the platform will be used for token burn

Line: 588

```
  /**
   *  Destroys `amount` tokens from `account`.`amount` is then deducted
   * from the caller's allowance.
   *
   * See {_burn} and {_approve}.
   */
  function _burnFrom(address account, uint256 amount) internal {
    _burn(account, amount);
    _approve(account, _msgSender(), _allowances[account][_msgSender()].sub(amount,
"BEP20: burn amount exceeds allowance"));
  }
}
```

Remediation

Either make this function external or remove if not needed

Status: Fixed

Medium severity issues

No issues were found.

Low level severity issues

No issues were found.

Informational

No issues were found.

Functional Tests

Function Names	Testing results
owner	PASS
onlyOwner	PASS
transfer	PASS
approve	PASS
transferFrom	PASS
allowance	PASS
_mint	PASS
_burnFrom	CLOSED
renounceOwnership	PASS
transferOwnership	PASS
increaseAllowance	PASS
decreaseAllowance	PASS

Closing Summary

All the issues mentioned in the report are resolved and the contract is working fine for multiple test cases and token security prospective.

Disclaimer

Quillhash audit is not a security warranty, investment advice, or endorsement of the PADD platform. This audit does not provide a security or correctness guarantee of the audited smart contracts. The statements made in this document should not be interpreted as investment or legal advice, nor should its authors be held accountable for decisions made based on them. Securing smart contracts is a multistep process. One audit cannot be considered enough. We recommend that the PADD Team put in place a bug bounty program to encourage further analysis of the smart contract by other third parties.

Audit Report September, 2021

For

PADD.FINANCE

QuillAudits

Canada, India, Singapore, United Kingdom

audits.quillhash.com

audits@quillhash.com