



Security Assessment

**balance finance**

Oct 14th, 2021



# Table of Contents

## Summary

### Overview

[Project Summary.](#)

[Audit Summary.](#)

[Vulnerability Summary.](#)

[Audit Scope](#)

### Findings

[BFP-01 : Unlocked Compiler Version](#)

[BFP-02 : Lack of Error Message](#)

[BFP-03 : Visibility Specifiers Missing](#)

[BFP-04 : Missing Emit Events](#)

[BFP-05 : Use `require` Instead Of `if`](#)

[BFP-06 : Lack of Zero Address Validation](#)

[BFP-07 : Unused Return Value](#)

[BFP-08 : Initial Token Distribution](#)

[BFP-09 : Centralization Risk](#)

[BFP-10 : Too Many Digits](#)

[BFP-11 : Function Visibility Optimization](#)

[BFP-12 : Not Reset `newOwner`](#)

### Appendix

### Disclaimer

### About

# Summary

This report has been prepared for balance finance to discover issues and vulnerabilities in the source code of the balance finance project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Static Analysis and Manual Review techniques.

The auditing process pays special attention to the following considerations:

- Testing the smart contracts against both common and uncommon attack vectors.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

The security assessment resulted in findings that ranged from critical to informational. We recommend addressing these findings to ensure a high level of security standards and industry practices. We suggest recommendations that could better serve the project from the security perspective:

- Enhance general coding practices for better structures of source codes;
- Add enough unit tests to cover the possible use cases;
- Provide more comments per each function for readability, especially contracts that are verified in public;
- Provide more transparency on privileged activities once the protocol is live.

# Overview

## Project Summary

Project Name	balance finance
Platform	Ethereum
Language	Solidity
Codebase	<a href="https://etherscan.io/token/0xbccf7f0d846a051b8845bd216ee0ffb5131f9873">https://etherscan.io/token/0xbccf7f0d846a051b8845bd216ee0ffb5131f9873</a>
Commit	

## Audit Summary

Delivery Date	Oct 14, 2021
Audit Methodology	Static Analysis, Manual Review
Key Components	

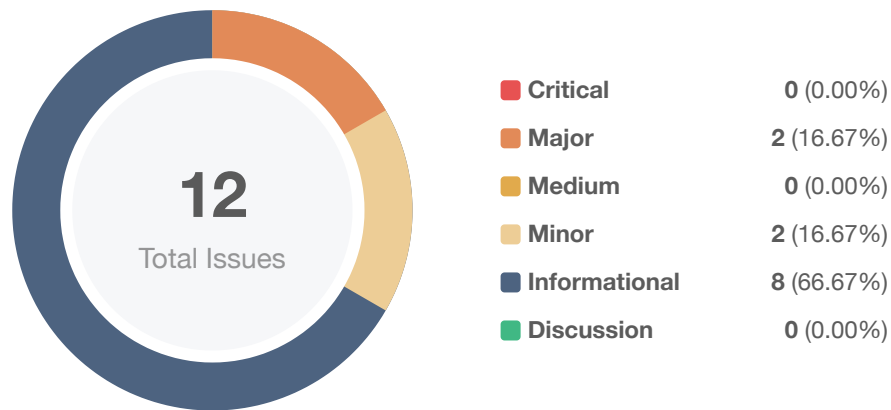
## Vulnerability Summary

Vulnerability Level	Total	⚠ Pending	⊗ Declined	ℹ Acknowledged	🔄 Partially Resolved	✅ Resolved
🔴 Critical	0	0	0	0	0	0
🟠 Major	2	0	0	2	0	0
🟡 Medium	0	0	0	0	0	0
🟠 Minor	2	0	0	2	0	0
🟡 Informational	8	0	0	8	0	0
🟢 Discussion	0	0	0	0	0	0

## Audit Scope

ID	File	SHA256 Checksum
BFP	balancefinance.sol	c9ac457da1614b9a64537f8ef688a4d6db6b53b05cacd292012419cb114a55d5

# Findings



ID	Title	Category	Severity	Status
BFP-01	Unlocked Compiler Version	Language Specific	● Informational	① Acknowledged
BFP-02	Lack of Error Message	Coding Style	● Informational	① Acknowledged
BFP-03	Visibility Specifiers Missing	Language Specific	● Informational	① Acknowledged
BFP-04	Missing Emit Events	Gas Optimization	● Informational	① Acknowledged
BFP-05	Use <code>require</code> Instead Of <code>if</code>	Gas Optimization	● Informational	① Acknowledged
BFP-06	Lack of Zero Address Validation	Volatile Code	● Minor	① Acknowledged
BFP-07	Unused Return Value	Coding Style	● Informational	① Acknowledged
<b>BFP-08</b>	Initial Token Distribution	<b>Centralization / Privilege</b>	● Major	① Acknowledged
<b>BFP-09</b>	Centralization Risk	<b>Centralization / Privilege</b>	● Major	① Acknowledged
BFP-10	Too Many Digits	Coding Style	● Informational	① Acknowledged
BFP-11	Function Visibility Optimization	Gas Optimization	● Informational	① Acknowledged
BFP-12	Not Reset <code>newOwner</code>	Logical Issue	● Minor	① Acknowledged

## BFP-01 | Unlocked Compiler Version

Category	Severity	Location	Status
Language Specific	● Informational	balancefinance.sol: 5	ⓘ Acknowledged

### Description

The contract contains unlocked compiler versions. An unlocked compiler version in the contract's source code permits the user to compile it at or above a particular version. This, in turn, leads to differences in the generated bytecode between compilations due to differing compiler version numbers. This can lead to ambiguity when debugging as compiler-specific bugs may occur in the codebase that would be difficult to identify over a span of multiple compiler versions rather than a specific one.

### Recommendation

It is general practice to alternatively lock the compiler at a specific version rather than allow a range of compiler versions to be utilized to avoid compiler-specific bugs and thus be able to detect emerging ones. We recommend locking the compiler at the lowest possible version that supports all the capabilities required by the codebase. This will ensure that the project utilizes a compiler version that has been in use for the longest time and as such is less likely to contain yet-undiscovered bugs.

### Alleviation

No alleviation.

## BFP-02 | Lack of Error Message

Category	Severity	Location	Status
Coding Style	● Informational	balancefinance.sol: 9, 15, 46, 54, 85	ⓘ Acknowledged

### Description

`require` can be used to check for conditions and throw an exception if the condition is not met, in which case the descriptive error message provided by the developer will appear and help to tracking error and debugging.

### Recommendation

We advise the client to add error messages.

### Alleviation

No alleviation.



## BFP-03 | Visibility Specifiers Missing

Category	Severity	Location	Status
Language Specific	● Informational	balancefinance.sol: 12~13, 40~41	ⓘ Acknowledged

### Description

The linked variable declaration does not have a visibility specifier explicitly set.

### Recommendation

Inconsistencies in the default visibility the Solidity compilers impose can cause issues in the functionality of the codebase. We advise that visibility specifier for the linked variable is explicitly set.

### Alleviation

No alleviation.

## BFP-04 | Missing Emit Events

Category	Severity	Location	Status
Gas Optimization	● Informational	balancefinance.sol: 14	ⓘ Acknowledged

### Description

Functions that affect the status of sensitive variables should be able to emit events as notifications to customers.

- `changeOwner()`
- `acceptOwnership()`

### Recommendation

We advise the client to add events for sensitive actions and emit them in the function as follows:

```
event OwnerChanged(address indexed previousOwner, address indexed newOwner);

function changeOwner(address payable _newOwner) public onlyOwner {
    require(_newOwner!=address(0));
    emit OwnerChanged(newOwner, _newOwner);
    newOwner = _newOwner;
}
```

### Alleviation

No alleviation.

## BFP-05 | Use `require` Instead Of `if`

Category	Severity	Location	Status
Gas Optimization	● Informational	balancefinance.sol: 19	ⓘ Acknowledged

### Description

`require` can be used to check for conditions and throw an exception if the condition is not met, in which case the descriptive error message provided by the developer will appear and help to tracking error and debugging.

### Recommendation

We advise that the `if` conditional is instead executed as a `require` statement thus removing the necessity of an `if` block.

### Alleviation

No alleviation.

## BFP-06 | Lack of Zero Address Validation

Category	Severity	Location	Status
Volatile Code	● Minor	balancefinance.sol: 45, 53, 62	ⓘ Acknowledged

### Description

The variables `_to` , `_from` and `_spender` should be verified as non-zero values to prevent being mistakenly assigned as `address(0)` in the `transfer()` , `transferFrom()` and `approve()` functions respectively.

### Recommendation

We advise the client to check that the addresses are not zero in `transfer()` , `transferFrom()` and `approve()` like as follows:

```
require(_to != address(0), "Token: _to is a zero address");
```

### Alleviation

No alleviation.

## BFP-07 | Unused Return Value

Category	Severity	Location	Status
Coding Style	● Informational	balancefinance.sol: 45, 53, 62, 68, 43	① Acknowledged

### Description

The return value `balance`, `success` and `remaining` are declared but never used in the function body.

### Recommendation

We advise the client to remove or comment out the function parameter.

### Alleviation

No alleviation.

## BFP-08 | Initial Token Distribution

Category	Severity	Location	Status
Centralization / Privilege	● Major	balancefinance.sol: 81	ⓘ Acknowledged

## Description

The 2000000000000000000000 of tokens are sent to the owner when deploying the contract. This could be a centralization risk as the owner can distribute tokens without obtaining the consensus of the community.

## Recommendation

We recommend the team be transparent regarding the initial token distribution process.

## Alleviation

No alleviation.

## BFP-09 | Centralization Risk

Category	Severity	Location	Status
Centralization / Privilege	● Major	balancefinance.sol: 86	ⓘ Acknowledged

### Description

To bridge the gap in trust between the administrators need to express a sincere attitude regarding the considerations of the administrator team's anonymity.

The `owner` has the responsibility to notify users about the following capabilities:

- set the newOwner through `changeOwner()`
- transfer ETH to itself automatically when someone or other contract sends ETH to it.

### Recommendation

We advise the client to carefully manage the privileged account's private keys to avoid any potential risks of being hacked. In general, we strongly recommend centralized privileges or roles in the protocol to be improved via a decentralized mechanism or via smart-contract-based accounts with enhanced security practices, e.g. Multisignature wallets.

Indicatively, here are some feasible suggestions that would also mitigate the potential risks at the different levels in terms of the short-term and long-term:

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;
- Assignment of privileged roles to multi-signature wallets to prevent a single point of failure due to the private key;
- Introduction of a DAO/governance/voting module to increase transparency and user involvement.

### Alleviation

No alleviation.

## BFP-10 | Too Many Digits

Category	Severity	Location	Status
Coding Style	● Informational	balancefinance.sol: 79	ⓘ Acknowledged

### Description

Literals with many digits are difficult to read and review.

### Recommendation

We advise the client to use the scientific notation to improve readability.

### Alleviation

No alleviation.



## BFP-11 | Function Visibility Optimization

Category	Severity	Location	Status
Gas Optimization	● Informational	balancefinance.sol: 14, 18, 43, 45, 53, 62, 68	① Acknowledged

### Description

`public` functions that are never called by the contract could be declared `external`. When the inputs are arrays, `external` functions are more efficient than `public` functions.

For example:

- `changeOwner()`
- `acceptOwnership()`
- `balanceOf()`
- `transfer()`
- `transferFrom()`
- `approve()`
- `allowance()`

### Recommendation

We advise that the functions' visibility specifiers are set to `external` and the array-based arguments change their data location from `memory` to `calldata`, optimizing the gas cost of the function.

### Alleviation

No alleviation.

## BFP-12 | Not Reset `newOwner`

Category	Severity	Location	Status
Logical Issue	● Minor	balancefinance.sol: 18	ⓘ Acknowledged

### Description

According to the current logic, the `newOwner` should be reset.

### Recommendation

We advise the client to set the `newOwner` to `address(0)`.

### Alleviation

No alleviation.

# Appendix

## Finding Categories

### Centralization / Privilege

Centralization / Privilege findings refer to either feature logic or implementation of components that act against the nature of decentralization, such as explicit ownership or specialized access roles in combination with a mechanism to relocate funds.

### Gas Optimization

Gas Optimization findings do not affect the functionality of the code but generate different, more optimal EVM opcodes resulting in a reduction on the total gas cost of a transaction.

### Logical Issue

Logical Issue findings detail a fault in the logic of the linked code, such as an incorrect notion on how `block.timestamp` works.

### Volatile Code

Volatile Code findings refer to segments of code that behave unexpectedly on certain edge cases that may result in a vulnerability.

### Language Specific

Language Specific findings are issues that would only arise within Solidity, i.e. incorrect usage of `private` or `delete`.

### Coding Style

Coding Style findings usually do not affect the generated byte-code but rather comment on how to make the codebase more legible and, as a result, easily maintainable.

## Checksum Calculation Method

The "Checksum" field in the "Audit Scope" section is calculated as the SHA-256 (Secure Hash Algorithm 2 with digest size of 256 bits) digest of the content of each file hosted in the listed source repository under the specified commit.

The result is hexadecimal encoded and is the same as the output of the Linux "sha256sum" command against the target file.

# Disclaimer

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to you (“Customer” or the “Company”) in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes, nor may copies be delivered to any other person other than the Company, without CertiK’s prior written consent in each instance.

This report is not, nor should be considered, an “endorsement” or “disapproval” of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any “product” or “asset” created by any team or project that contracts CertiK to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. CertiK’s position is that each company and individual are responsible for their own due diligence and continuous security. CertiK’s goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

The assessment services provided by CertiK is subject to dependencies and under continuing development. You agree that your access and/or use, including but not limited to any services, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third-parties.

ALL SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF ARE PROVIDED “AS IS” AND “AS

AVAILABLE” AND WITH ALL FAULTS AND DEFECTS WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, CERTIK HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, CERTIK SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM COURSE OF DEALING, USAGE, OR TRADE PRACTICE. WITHOUT LIMITING THE FOREGOING, CERTIK MAKES NO WARRANTY OF ANY KIND THAT THE SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET CUSTOMER’S OR ANY OTHER PERSON’S REQUIREMENTS, ACHIEVE ANY INTENDED RESULT, BE COMPATIBLE OR WORK WITH ANY SOFTWARE, SYSTEM, OR OTHER SERVICES, OR BE SECURE, ACCURATE, COMPLETE, FREE OF HARMFUL CODE, OR ERROR-FREE. WITHOUT LIMITATION TO THE FOREGOING, CERTIK PROVIDES NO WARRANTY OR UNDERTAKING, AND MAKES NO REPRESENTATION OF ANY KIND THAT THE SERVICE WILL MEET CUSTOMER’S REQUIREMENTS, ACHIEVE ANY INTENDED RESULTS, BE COMPATIBLE OR WORK WITH ANY OTHER SOFTWARE, APPLICATIONS, SYSTEMS OR SERVICES, OPERATE WITHOUT INTERRUPTION, MEET ANY PERFORMANCE OR RELIABILITY STANDARDS OR BE ERROR FREE OR THAT ANY ERRORS OR DEFECTS CAN OR WILL BE CORRECTED.

WITHOUT LIMITING THE FOREGOING, NEITHER CERTIK NOR ANY OF CERTIK’S AGENTS MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND, EXPRESS OR IMPLIED AS TO THE ACCURACY, RELIABILITY, OR CURRENCY OF ANY INFORMATION OR CONTENT PROVIDED THROUGH THE SERVICE. CERTIK WILL ASSUME NO LIABILITY OR RESPONSIBILITY FOR (I) ANY ERRORS, MISTAKES, OR INACCURACIES OF CONTENT AND MATERIALS OR FOR ANY LOSS OR DAMAGE OF ANY KIND INCURRED AS A RESULT OF THE USE OF ANY CONTENT, OR (II) ANY PERSONAL INJURY OR PROPERTY DAMAGE, OF ANY NATURE WHATSOEVER, RESULTING FROM CUSTOMER’S ACCESS TO OR USE OF THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS.

ALL THIRD-PARTY MATERIALS ARE PROVIDED “AS IS” AND ANY REPRESENTATION OR WARRANTY OF OR CONCERNING ANY THIRD-PARTY MATERIALS IS STRICTLY BETWEEN CUSTOMER AND THE THIRD-PARTY OWNER OR DISTRIBUTOR OF THE THIRD-PARTY MATERIALS.

THE SERVICES, ASSESSMENT REPORT, AND ANY OTHER MATERIALS HEREUNDER ARE SOLELY PROVIDED TO CUSTOMER AND MAY NOT BE RELIED ON BY ANY OTHER PERSON OR FOR ANY PURPOSE NOT SPECIFICALLY IDENTIFIED IN THIS AGREEMENT, NOR MAY COPIES BE DELIVERED TO, ANY OTHER PERSON WITHOUT CERTIK’S PRIOR WRITTEN CONSENT IN EACH INSTANCE.

NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING

MATERIALS AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST CERTIK WITH RESPECT TO SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS.

THE REPRESENTATIONS AND WARRANTIES OF CERTIK CONTAINED IN THIS AGREEMENT ARE SOLELY FOR THE BENEFIT OF CUSTOMER. ACCORDINGLY, NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH REPRESENTATIONS AND WARRANTIES AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST CERTIK WITH RESPECT TO SUCH REPRESENTATIONS OR WARRANTIES OR ANY MATTER SUBJECT TO OR RESULTING IN INDEMNIFICATION UNDER THIS AGREEMENT OR OTHERWISE.

FOR AVOIDANCE OF DOUBT, THE SERVICES, INCLUDING ANY ASSOCIATED ASSESSMENT REPORTS OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL, REGULATORY, OR OTHER ADVICE.

## About

Founded in 2017 by leading academics in the field of Computer Science from both Yale and Columbia University, CertiK is a leading blockchain security company that serves to verify the security and correctness of smart contracts and blockchain-based protocols. Through the utilization of our world-class technical expertise, alongside our proprietary, innovative tech, we're able to support the success of our clients with best-in-class security, all whilst realizing our overarching vision; provable trust for all throughout all facets of blockchain.

