CERTIK

Security Assessment

DeFi Basket

Nov 9th, 2021

Table of Contents

Summary

This report has been prepared for DeFi Basket to discover issues and vulnerabilities in the source code of the DeFi Basket 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	DeFi Basket
Description	https://defibasket.org/
Platform	Ethereum
Language	Solidity
Codebase	https://github.com/defibasket/defibasket-contracts
Commit	67fd9a879a1b381a2788db9a8266e70ca5184e71

Audit Summary

Delivery Date	Nov 09, 2021
Audit Methodology	Static Analysis, Manual Review
Key Components	

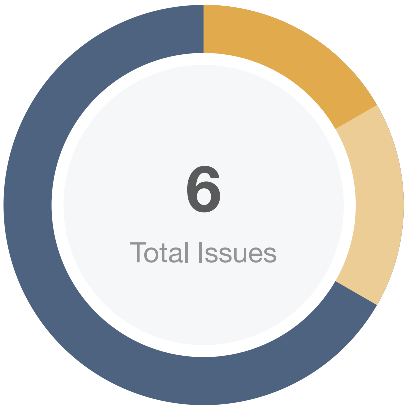
Vulnerability Summary

Vulnerability Level	Total	Pending	Declined	Acknowledged	Partially Resolved	Resolved
● Critical	0	0	0	0	0	0
● Major	0	0	0	0	0	0
● Medium	1	0	0	1	0	0
● Minor	1	0	0	1	0	0
● Informational	4	0	0	2	0	2
● Discussion	0	0	0	0	0	0

Audit Scope

ID	File	SHA256 Checksum
AVD	bridges/trusted/AaveV2DepositBridge/AaveV2DepositBridge.sol	721842b3d1060a0234ec0769eff1574d52833b7648fa1665469632176de66353
ADB	bridges/trusted/AutofarmDepositBridge/AutofarmDepositBridge.sol	30982c2106ae8edf54258aae3fe163c94b7ea13e973fdc8ebed9e806895f6eae
QLB	bridges/trusted/QuickswapLiquidityBridge/QuickswapLiquidityBridge.sol	8a50f1b905636f28bf3f7308d7ae82ad67caa565530c121b9e0a052c4bdea56c
QSB	bridges/trusted/QuickswapSwapBridge/QuickswapSwapBridge.sol	88eebb33a0482268951b245e805f89e10dfd7d00795bb84df5b33325418c0ba6
WMW	bridges/trusted/WMaticWrapBridge/WMaticWrapBridge.sol	0456d1b1fdf77203d56b01529f172f32abc562f9176a61fb45549819670361e4
DFB	DeFiBasket.sol	c378b6a411facac418a23fe48e8f1426992d3e113e2754fcf7e2fb0089591f5c
WAL	Wallet.sol	6f8ad74701000001085f392152c9a24e5251d5a0e216655493fa016b9bd0fdcd

Findings



🟥 **Critical**	**0** (0.00%)
🟧 **Major**	**0** (0.00%)
🟨 **Medium**	**1** (16.67%)
▨ **Minor**	**1** (16.67%)
🟦 **Informational**	**4** (66.67%)
🟩 **Discussion**	**0** (0.00%)

ID	Title	Category	Severity	Status
DeFi Basket-01	Financial Models	Logical Issue	● Informational	ⓘ Acknowledged
BRI-01	Third Party Dependencies	Volatile Code	● Informational	ⓘ Acknowledged
BRI-02	Logic issue of `bridges`	Volatile Code	● Minor	ⓘ Acknowledged
IPL-01	Fee Distribution	Logical Issue	● Informational	✓ Resolved
IPL-03	Potential fake bridge address	Logical Issue	● Medium	ⓘ Acknowledged
WAL-01	Boolean Equality	Coding Style	● Informational	✓ Resolved

DeFi Basket-01 | Financial Models

Category	Severity	Location	Status
Logical Issue	● Informational	Global	ⓘ Acknowledged

Description

This protocol is used to absorb user deposits, including `ETH` and other `ERC20` tokens, and to obtain income by investing these funds in a third-party Defi agreement. But after reviewing the code, this protocol has the following logical issues:

- When the user deposit by function "depositPortfolio", this protocol does not record the user's amount, so this protocol cannot verify the amount when the user withdraws.
- There are no rewards for users who deposit in this protocol.
- When the user withdraws by function "withdrawPortfolio", there may be not enough ERC20 tokens or ETH, so the user cannot withdraw his deposit.

Recommendation

Financial models of blockchain protocols need to be resilient to attacks. They need to pass simulations and verifications to guarantee the security of the overall protocol.

The financial model of this protocol is not in the scope of this audit.

Alleviation

[DeFi Basket Team]:

- Each portfolio has its own Wallet, which is a separate smart contract with a separate address. Amounts available to withdraw are based on how much balance this address has for each asset.
- Yes, there is no logic on the contract for rewards to users that deposit.
- It is possible that the withdrawal transaction is generated with parameters that will make the transaction fail. We ensure the transaction will succeed by keeping track of the allocation of that specific portfolio. This is all done off-chain using python scripts and MongoDB.

BRI-01 | Third Party Dependencies

Category	Severity	Location	Status
Volatile Code	● Informational	bridges	ⓘ Acknowledged

Description

In the branch of `bridges`, there are a lot of third-party protocols:

- `AaveV2DepositBridge.sol`, which uses `Aave2LendingPool` and `AaveIncentivesController`;
- `AutofarmDepositBridge.sol`, which uses `AutoFarm` protocol;
- `QuickswapSwapBridge,sol`and `QuickswapLiquidityBridge`, which use `QuickSwap`.

The contract is serving as the underlying entity to interact with the above third-party protocols. The scope of the audit treats 3rd party entities as black boxes and assumes their functional correctness. However, in the real world, 3rd parties can be compromised and this may lead to lost or stolen assets. In addition, upgrades of 3rd parties can possibly create severe impacts, such as increasing fees of 3rd parties, migrating to new LP pools, etc.

Recommendation

We understand that the business logic of `DeFi Basket` requires interaction with `Aave2`, `AutoFarm`, etc. We encourage the team to constantly monitor the statuses of 3rd parties to mitigate the side effects when unexpected activities are observed.

Alleviation

[DeFi Basket Team]: The team will constantly:

1. Monitor the status of third-party protocols by constantly checking their updates and vulnerabilities;
2. Continuously monitoring smart contracts using tools such as Forta.

BRI-02 | Logic issue of `bridges`

Category	Severity	Location	Status
Volatile Code	● Minor	bridges	ⓘ Acknowledged

Description

All of the bridge contracts owners will invest users' deposits in the third-party Defi protocol to obtain rewards. Taking the `AaveV2DepositBridge` contract as an example, there are the following two issues that need to discuss:

1. In our opinion, the principle of a bridge contract comes from the user's wallet, but we did not find this fund flow. Especially, how can guarantee that the investment principal in the bridge contract can be returned to the corresponding wallet address when the user withdraws funds from the wallet?

2. How to transfer the investment income of the current contract?

Alleviation

[DeFi Basket Team]:

1. Funds are always tracked off-chain, and such tracking is required to generate the transactions which will withdraw funds from users' wallets. Soon this functionality will be built on The Graph so it is more decentralized and easily accessible to other users.

2. Remembering that wallet balances are tracked off-chain and taking AaveV2DepositBridge as an example, the withdraw flow is as follows:

- 2.1. User calls `withdrawPortfolio` from the front-end (only the NFT owner of the corresponding portfolio can call it);
- 2.2. `withdrawPortfolio` calls the method `_writeToWallet`, which makes the Wallet of the corresponding `nftId` make a delegate call to a bridge, in this case, the `withdraw` function from AaveV2DepositBridge;
- 2.3. Since the call is delegated, the amount to be withdrawn as well as the corresponding rewards are transferred to the user's Wallet;
- 2.4. `withdrawPortfolio` now calls the method `_withdrawFromWallet`, which transfers the assets from the Wallet to the corresponding NFT owner.

IPL-01 | Fee Distribution

Category	Severity	Location	Status
Logical Issue	● Informational	DeFiBasket.sol: 246, 248, 258	✓ Resolved

Description

Every time the function `_depositToWallet` is called, the `owner()` will get the fee reward.

Recommendation

This is the business logic of the DeFi Basket protocol, however, users should be aware of the fee distribution.

Alleviation

[DeFi Basket Team]:

The fee of 0.1% is shown from the user's interface, and it is one of our priorities to always be transparent in terms of fees and pricing.

The documentation (https://docs.defibasket.org/users/fees) and every possible action the user takes that will have a Defi Basket fee charged has a reminder for the user of this fee.

IPL-03 | Potential fake bridge address

Category	Severity	Location	Status
Logical Issue	● Medium	DeFiBasket.sol: 108, 134	ⓘ Acknowledged

Description

Functions `createPortfolio` and `depositPortfolio` are `external`, and their parameters `address[] calldata bridgeAddresses,` will be passed to function `_writeToWallet`. And then `Wallet.useBridges` will call the specified contract with `bridgeAddresses` and `bridgeEncodedCalls`. How to verify the input parameters are effective? The current `modifier checkBridgeCalls` cannot prevent malicious requests.

```
function useBridges(
  address[] calldata bridgeAddresses,
  bytes[] calldata bridgeEncodedCalls
) external override defibasketOnly {
  bool isSuccess;
  bytes memory result;

  for (uint16 i = 0; i < bridgeAddresses.length; i++) {
    (isSuccess, result) = bridgeAddresses[i].delegatecall(bridgeEncodedCalls[i]);
    ......
}
```

Recommendation

We recommend adding a new contract to maintain the white list of the bridge contracts and getting the bridge address from the white list array.

Alleviation

[DeFi Basket Team]:

We understand that by implementing a whitelist, DeFi Basket would no longer be a permissionless protocol, and it would become more centralized than we would like it to be.

Furthermore, the cases where it actually protects our users are scarce and belong in the following categories:

1. DeFi Basket UI has been compromised;
2. A malicious UI is impersonating DeFi Basket (whitelisting should be the least of our problems here);

In our opinion, these cases are not worth the cost of having a permissioned and centralized whitelisting process: in both of these, the hacker could also impersonate the whitelist, making the protection offered by the whitelist not more useful.

The DeFi Basket web app will only ever interact by default with trusted bridges. In other words, when the transaction is generated via the DeFi Basket web app, each and every parameter passed to bridgeAddresses will match a bridge that is trusted and has been audited.

In the nature of keeping the protocol decentralized and permissionless, we do not use a hardcoded whitelist array. And only after we have audited those bridges does it become part of the official trusted bridges, at which point users will be able to select it from the UI by default.

Another reason to not use a hardcoded whitelist array is to foster the development of users' own bridges, which will in the future be shared with the wider DeFi Basket community. As has already been stated, these bridges will only become trusted bridges and be added to DeFi Basket's web app after a thorough auditing process.

To further protect our users, we have created an open source verifier for the transactions. This provides an additional layer of security for the users and enables the protocol to remain permissionless and decentralized by design. It allows a user to ensure all interactions only occur with trusted and audited bridges. The checker is available at https://defibasket.github.io/checker/ and its source code at https://github.com/defibasket/defibasket-transaction-checker.

Trusted bridges can also be checked in the documentation: https://docs.defibasket.org/developer/bridges/trusted-bridges.

WAL-01 | Boolean Equality

Category	Severity	Location	Status
Coding Style	● Informational	Wallet.sol (main): 63	✓ Resolved

Description

Detects the comparison to boolean constants. Boolean constants can be used directly and do not need to be compare to true or false.

Recommendation

We advise removing the equality to the boolean constant and referring to the following codes:

```
63    if (!isSuccess) {
```

Alleviation

The team heeded the advice and resolved this issue in commit `4637d3c8e6cf0497e85443f563de4de1acc44c44`.

Appendix

Finding Categories

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.

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.

CERTIK