THE ARCADIA GROUP

Audit of YFValue Vaults

a report authored by

Connor Martin

reviewed by

Van Cam Pham, PhD

innovative fortuna iuvat

September 18th, 2020

Executive Summary

A Representative Party of the YFValue Decentralized Organization ("YFValue") engaged The Arcadia Group ("Arcadia"), a software development, research and security company, to conduct a review of the following YFValue Vault Smart Contracts ("YFV Vaults") on the yfv-finance/vaults repo at Commit a705a78e6a033d1bdec99d9d4c38fd26b3cc3015.

YFVGovernanceVault.sol
YFVController.sol

Arcadia completed the review using various methods primarily consisting of dynamic and static analysis. This process included a line by line analysis of the in-scope contracts, optimization analysis, analysis of key functionalities and limiters, and reference against intended functionality.

The flattened contract in question for this audit is deployed at
0x07eb8CB8AEdB581a2d73cc29F6c7860226808Ca2

Dynamic Findings

YFVController.sol

Severity	Lines	Description
Medium	327	amountOutMin is hard coded to zero leaving contract susceptible to flashswap or other liquidity manipulation attacks.
Medium	274	Nested IF function should have a ELSE revert() resolution if conditions are not met.

YFVGovernancevault.sol

Severity	Lines	Description
Low	506, 709	Modifier located in function stack, should be located before functions begin.
Low	709	checkNextEpoch modifier has nested IF ELSE logic that needs clarification
Low	711	checkNextEpoch modifier is contingent on block.timestamp, which can be altered by miners

Medium	381	Current Epoch uses uint8, not uint256 and so it does not inherit safemath.sol. That means it is vulnerable to underflow / overflow if the IF/ELSE logic does not function as expected. If currentEpoch is used in any functionality in the future it will be vulnerable.

Static Findings

YFVController.sol

Severity	Lines	Description
Low	264	A call to a user-supplied address is executed. An external message call to an address specified by the caller is executed. Note that the callee account might contain arbitrary code and could re-enter any function within this contract. Reentering the contract in an intermediate state may lead to unexpected behaviour. Make sure that no state modifications are executed after this call and/or reentrancy guards are in place
Low	264	Requirement violation. A requirement was violated in a nested call and the call was reverted as a result. Make sure valid inputs are provided to the nested call (for instance, via passed arguments).
Low	269	Multiple calls are executed in the same transaction. This call is executed following another call within the same transaction. It is possible that the call never gets executed if a prior call fails permanently. This might be caused intentionally by a malicious callee. If possible, refactor the code such that each transaction only executes one external call or make sure that all callees can be trusted (i.e. they’re part of your own codebase).

YFVGovernancevault.sol

Severity	Lines	Description
Medium	420	Use of "tx.origin" as a part of authorization control. refer to previous tx.origin issue.

Disclaimer

While best efforts and precautions have been taken in the preparation of this document, The Arcadia Group and the Authors assume no responsibility for errors, omissions, or for damages resulting from the use of the provided information. Additionally Arcadia would like to emphasize that use of Arcadia's services does not guarantee the security of a smart contract or set of smart contracts and does not guarantee against attacks. One audit on its own is not enough for a project to be considered secure; that categorization can only be earned through extensive peer review and battle testing over an extended period of time.