Certified
Certified Blockchain Security Professional™

Certified
Certified Solidity Developer™

OVERVIEW

PROJECT SUMMARY

Project	**FINEVIP**
Platform	Ethereum
Language	Solidity

AUDIT SUMMARY

Date	13-02-2023	
Audit Type	Static Analysis, Manual Review	
Audit Result	**Pending**	
Auditor	**Jarmo van de Seijp**	https://tinyurl.com/Jvdseijp

RISK SUMMARY

Risk Level	Total	Found	Pending	Resolved	Acknowledgde	Objected
Critical	1	1	1	0	0	0
Major	2	2	1	1	0	0
Medium	2	2	0	0	1	1
Minor	7	7	4	1	0	2
Informative	13	13	2	0	10	1
Discussion	0	0	0	0	0	0

FINDINGS

unchecked return value

Description:

The contract staking.sol makes use of the **TransferFrom** function as part of the **staking** function. However, the results are unchecked.

Category	Risk Level	Number of Findings	Status
Return Value	Medium	1	Objected

Objection from the project developer:
no, don't need to return value for transfer and transferFrom function
if it would be failed, error will be returned

Lacking zero-address check

Description:

The **recoverFINEVIP** functions sends tokens to an arbitrary address, but does not check whether this address is valid, or the zero-address.

Category	Risk Level	Number of Findings	Status
zero-adress	minor	1	pending

Centralized privilege

Description:

The owner can use **changePercent** to set any arbitrary percentage, including ones that are unrealistic or not executable.

Category	Risk Level	Number of Findings	Status
centralization	Major	1	resolved

Unchecked withdraw ability

Description:

The contract allows the owner to withdraw all USDC funds from the contract, without taking into account allocation for staked users

Category	Risk Level	Number of Findings	Status
Centralized Privilege	Medium	1	Acknowledged

Note from the project owner:
About this part, yes I need to have full access to the USDC. Because this staking contract are not meant to be for long term, it's only going to be for 2 months. We use this as community bootstrap token sales
So later in April something, the staking reward will change and whatever USDC remain there, I may need to withdraw it or however I see fit.
This is not a public staking platform. As you can see there is only 500,000 FINEVIP which is worth 500,000 USDC. So we are selling it to our private investor and communities only.

Unrestricted burn access

Description:

The owner of the project, by appointing an operator, has the unlimited and unrestricted power to burn an arbitrary amount from a wallet of their chosing.

Category	Risk Level	Number of Findings	Status
Centralization	Critical	1	pending

Default function visibility

SWC-ID: SWC-100
Relationship:

CWE-710: Improper Adherence to Coding Standards

Description:
Functions that are only intended for external calls, do not need a function visibility type specified as public by default.

Category	Risk Level	Number of Findings	Status
SWC-100	informative	**10**	acknowledged

Push-Over-Pull

Relationship:
CWE-710: Improper Adherence to Coding Standards

Description:

The transfer of the contract's ownership through the function **transferOwnership()** only has 1 check, which is to ensure that the new owner is not the 0 address. It does not, however, check whether or not the ownership can be accepted by the recipient **newOwner**. In the case of a transfer of ownership to an incorrect address, or a smart contract that is not able to use the privileged functions, ownership of the contract is lost permanently with no way of getting it back. It is therefore advisable to use a pull method as opposed to push, in which case the **newOwner** would have to pro-actively accept ownership upon receiving it.

Category	Risk Level	Number of Findings	Status
Push over Pull	Minor	**1**	pending

Redundant Code

SWC-ID: SWC-135
Relationship:

CWE-1164: Irrelevant Code

Description:
The contracts makes external calls to check the decimals of the USDT and USDC coins. Since this call always returns 6 for either, the code can be optimzed by explicitly setting 6 as the decimals value, in stead of making the external call.

Category	Risk Level	Number of Findings	Status
SWC-135	informative	**2**	Objected

Missing Events

Description:

The variables **setSwapAddress**, **increaseReferralAmount** and **changePercent**, setSalePrice, setReferralPercentplay an important role in the smart contract, since they are key players
in its initial and subsequent ecosystem. The change of these variables is not emitted as an event. This may cause 3rd party applications as well as users to miss the changes to their respective variables, causing unwanted outcome for users or aggregators

note from the project developer: It is unneccesary to add events for setSwapAddress and IncreaseReferralAmount

Category	Risk Level	Number of Findings	Status
Missing events	minor	**5**	Resolved/ Objected

Centralized minting privilege

Description:

The owner can **mint** an arbitrary amount of tokens to any address without restrictions. This means that the value of the token can potentially be infinitely dilluted in case of compromise of the **owner** account.

Category	Risk Level	Number of Findings	Status
Centralization	Major	**1**	Pending

Typo in require statement

Description:

#137 contains a typo: "no enough USDC"

Category	Risk Level	Number of Findings	Status
Typo	informative	**1**	Pending

AUDIT RESULT

Basic Coding Bugs

1. Constructor Mismatch

o Description: Whether the contract name and its constructor are not identical to each other.

o Result: PASSED

o Severity: Critical

Ownership Takeover

o Description: Whether the set owner function is not protected.

o Result: PASSED

o Severity: Critical

Redundant Fallback Function

o Description: Whether the contract has a redundant fallback function.

o Result: PASSED

o Severity: Critical

Overflows & Underflows

Description: Whether the contract has general overflow or underflow Vulnerabilities

o Result: PASSED

o Severity: Critical

Reentrancy

o Description: Reentrancy is an issue when code can call back into your contract and change state, such as withdrawing ETHs.

o Result: PASSED

o Severity: Critical

MONEY-Giving Bug

o Description: Whether the contract returns funds to an arbitrary address.

o Result: PASSED

o Severity: High

Blackhole

o Description: Whether the contract locks ETH indefinitely: merely in without out.

o Result: PASSED

o Severity: High

Unauthorized Self-Destruct

o Description: Whether the contract can be killed by any arbitrary address.

o Result: PASSED

o Severity: Medium

Revert DoS

o Description: Whether the contractis vulnerable to DoSattack because of unexpected revert.

o Result: PASSED

o Severity: Medium

Unchecked External Call

o Description: Whether the contract has any external call without checking the return value.

o Result: PASSED

o Severity: Medium

Gasless Send

o Description: Whether the contractis vulnerable to gasless send.

o Result: PASSED

o Severity: Medium

Send Instead of Transfer

o Description: Whether the contract uses send instead of transfer.

o Result: PASSED

o Severity: Medium

Costly Loop

o Description: Whether the contract has any costly loop which may lead to Out-Of-Gas exception.

o Result: PASSED

o Severity: Medium

(Unsafe) Use of Untrusted Libraries

o Description: Whether the contract use any suspicious libraries.

o Result: PASSED

o Severity: Medium

(Unsafe) Use of Predictable Variables

o Description: Whether the contract contains any randomness variable, but its value can be predicated.

o Result: PASSED

o Severity: Medium

Transaction Ordering Dependence

o Description: Whether the final state of the contract depends on the order of the transactions.

o Result: PASSED

o Severity: Medium

. Deprecated Uses

o Description: Whether the contract use the deprecated tx.origin to perform the authorization.

o Result: PASSED

o Severity: Medium