OBELISK

Part of Tibereum Group

AUDITING REPORT

Version Notes

Version	No. Pages	Date	Revised By	Notes
1.0	Total: 19	2022-03-18	Donut, ByFixter	Audit Final

Audit Notes

Audit Date	2022-02-26 - 2022-03-18
Auditor/Auditors	Donut, ByFixter
Auditor/Auditors Contact Information	contact@obeliskauditing.com
Notes	Specified code and contracts are audited for security flaws. UI/UX (website), logic, team, and tokenomics are not audited.
Audit Report Number	OB555551472

Disclaimer

This audit is not financial, investment, or any other kind of advice and is for informational purposes only. This report is not a substitute for doing your own research and due diligence. Obelisk is not responsible or liable for any loss, damage, or otherwise caused by reliance on this report for any purpose. Obelisk has based this audit report solely on the information provided by the audited party and on facts that existed before or during the audit being conducted. Obelisk is not responsible for any outcome, including changes done to the contract/contracts after the audit was published. This audit is fully objective and only discerns what the contract is saying without adding any opinion to it. The audit is paid by the project but neither the auditors nor Obelisk has any other connection to the project and has no obligations other than to publish an objective report. Obelisk will always publish its findings regardless of the outcome of the findings. The audit only covers the subject areas detailed in this report and unless specifically stated, nothing else has been audited. Obelisk assumes that the provided information and material were not altered, suppressed, or misleading. This report is published by Obelisk, and Obelisk has sole ownership of this report. Use of this report for any reason other than for informational purposes on the subjects reviewed in this report including the use of any part of this report is prohibited without the express written consent of Obelisk. In instances where an auditor or team member has a personal connection with the audited project, that auditor or team member will be excluded from viewing or impacting any internal communication regarding the specific audit.

Obelisk Auditing

Defi is a relatively new concept but has seen exponential growth to a point where there is a multitude of new projects created every day. In a fast-paced world like this, there will also be an enormous amount of scams. The scams have become so elaborate that it’s hard for the common investor to trust a project, even though it could be legit. We saw a need for creating high-quality audits at a fast phase to keep up with the constantly expanding market. With the Obelisk stamp of approval, a legitimate project can easily grow its user base exponentially in a world where trust means everything. Obelisk Auditing consists of a group of security experts that specialize in security and structural operations, with previous work experience from among other things, PricewaterhouseCoopers. All our audits will always be conducted by at least two independent auditors for maximum security and professionalism.

As a comprehensive security firm, Obelisk provides all kinds of audits and project assistance.

Audit Information

The auditors always conducted a manual visual inspection of the code to find security flaws that automatic tests would not find. Comprehensive tests are also conducted in a specific test environment that utilizes exact copies of the published contract.

While conducting the audit, the Obelisk security team uses best practices to ensure that the reviewed contracts are thoroughly examined against all angles of attack. This is done by evaluating the codebase and whether it gives rise to significant risks. During the audit, Obelisk assesses the risks and assigns a risk level to each section together with an explanatory comment. Take note that the comments from the project team are their opinion and not the opinion of Obelisk.

Table of Contents

Project Information

Name	Summit
Description	Summit Defi is bringing new and unique features to the #DeFi space, starting off with what we are calling "Yield Multiplying" launching first on $FTM
Website	https://ftm.summitdefi.com/
Contact	https://twitter.com/SummitDefi
Contact information	@architect_dev on TG
Token Name(s)	N/A
Token Short	N/A
Contract(s)	See Appendix A
Code Language	Solidity
Chain	Fantom

Audit of Summit v2 Yieldwolf

The audited and deployed contract works as intended at its current form.

Obelisk was commissioned by Summit on the 23rd of February 2022 to conduct a comprehensive audit of Summits’ YieldWolf contracts. The following audit was conducted between the 26th of February 2022 and the 18th of March 2022. Two of Obelisk's security experts went through the related contracts manually using industry standards to find if any vulnerabilities could be exploited either by the project team or users.

During the audit of Summits’ YieldWolf contract, we found a single issue worth noting which is issue #1. It’s partially mitigated, and is not an issue as long as it’s accounted for by the project team.

The informational findings are good to know while interacting with the project but don’t directly damage the project in its current state, hence it’s up to the project team if they deem that it’s worth solving these issues.

The team has not reviewed the UI/UX, logic, team, or tokenomics of the Summit project**.**

Please read the full document for a complete understanding of the audit.

Summary Table

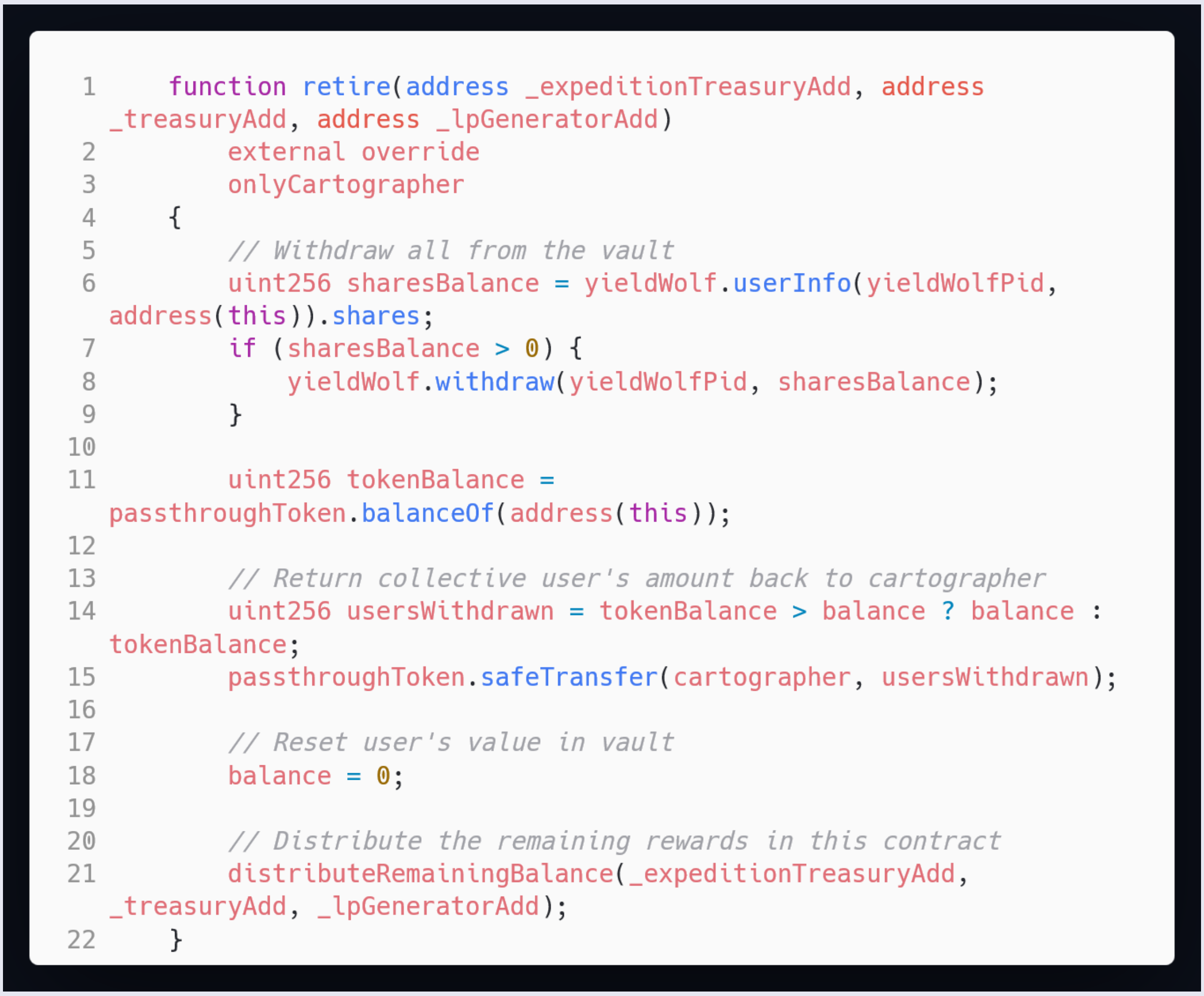
Finding	ID	Severity	Status
Pass Throughs Does Not Account For Fees When Enacting Or Retiring	#0001	Medium Risk	Partially Mitigated
No Events Emitted For Changes To Protocol Values	#0002	Informational	Closed

Findings

Manual Analysis

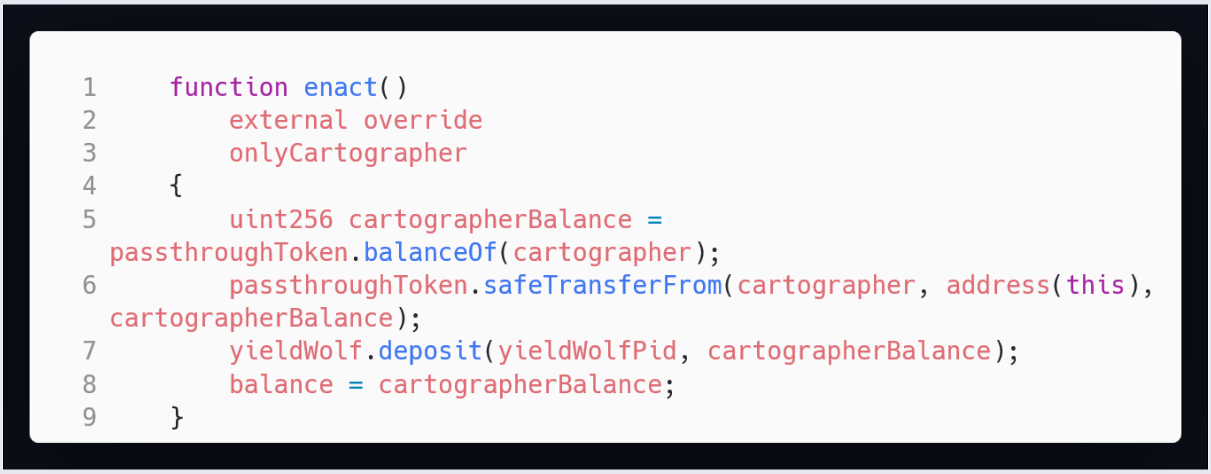
Pass Throughs Does Not Account For Fees When Enacting Or Retiring

FINDING ID	#0001
SEVERITY	Medium Risk
STATUS	Partially Mitigated
LOCATION	YieldWolfPassthrough.sol -> 236-257



```
    function retire(address _expeditionTreasuryAdd, address _treasuryAdd, address _lpGeneratorAdd)
        external override
        onlyCartographer
    {
        // Withdraw all from the vault
        uint256 sharesBalance = yieldWolf.userInfo(yieldWolfPid, address(this)).shares;
        if (sharesBalance > 0) {
            yieldWolf.withdraw(yieldWolfPid, sharesBalance);
        }

        uint256 tokenBalance = passthroughToken.balanceOf(address(this));

        // Return collective user's amount back to cartographer
        uint256 usersWithdrawn = tokenBalance > balance ? balance : tokenBalance;
        passthroughToken.safeTransfer(cartographer, usersWithdrawn);

        // Reset user's value in vault
        balance = 0;

        // Distribute the remaining rewards in this contract
        distributeRemainingBalance(_expeditionTreasuryAdd, _treasuryAdd, _lpGeneratorAdd);
    }
```

LOCATION	YieldWolfPassthrough.sol -> 161-169



```
    function enact()
        external override
        onlyCartographer
    {
        uint256 cartographerBalance = passthroughToken.balanceOf(cartographer);
        passthroughToken.safeTransferFrom(cartographer, address(this), cartographerBalance);
        yieldWolf.deposit(yieldWolfPid, cartographerBalance);
        balance = cartographerBalance;
    }
```

DESCRIPTION	The passthrough contract does not take any fees into account when enacting or retiring. If transfer, deposit, or withdrawal fees are present, users may not be able to withdraw all their funds until the underlying contract provides sufficient earnings.
RECOMMENDATION	Ensure that user deposits are correctly accounted for when retiring passthrough contracts.
RESOLUTION	Project team comment: “Retire is withdrawing the full amount in the vault. If the fees are covered by the earnings of the vault (withdrawn amount > running balance uint) then the running balance is sent to the cartographer, else the withdrawn amount is sent.” Obelisk Comment: “As long as the withdrawal fee has been covered by the current rewards, this will work. The project team should ensure that changing strategies is done sparingly.”

Static Analysis

No Events Emitted For Changes To Protocol Values

FINDING ID	#0002
SEVERITY	Informational
STATUS	Closed
LOCATION	YieldWolfPassthrough.sol -> 118-121: *function addExtraEarnToken(address _extraEarnToken) public onlyOwner {* YieldWolfPassthrough.sol -> 122-125: *function removeExtraEarnToken(address _extraEarnToken) public onlyOwner*

DESCRIPTION	Functions that change important variables should emit events such that users can more easily monitor the change.
RECOMMENDATION	Emit events from these functions.
RESOLUTION	Events were added to these functions.

On-Chain Analysis

No Findings

External Addresses

Externally Owned Accounts

Owner

ACCOUNT	0x3a7679E3662bC7c2EB2B1E71FA221dA430c6f64B
USAGE	Spooky BSHARE-FTM LP 0xbA74A5C08Ee6B8D63Cd51A80D236602637feBB71 Spooky BASED-TOMB LP 0x65810243e044a532272994856643078E65ef9611 VolatileV1 AMM - BOO/xBOO 0x77bF5EBc3912a091E1cDceF9041Dffe7b8639BC8 StableV1 AMM - USDC/MIM 0x940a44Fe2b1c6BB0b21170995Fd9BD57b45a7CfA *YieldWolfPassthrough.owner* - Variable
IMPACT	● receives elevated permissions as owner, operator, or other

External Contracts

These contracts are not part of the audit scope.

Cartographer

ADDRESS	0x876F890135091381c23Be437fA1cec2251B7c117
USAGE	Spooky BSHARE-FTM LP 0xbA74A5C08Ee6B8D63Cd51A80D236602637feBB71 Spooky BASED-TOMB LP 0x65810243e044a532272994856643078E65ef9611 VolatileV1 AMM - BOO/xBOO 0x77bF5EBc3912a091E1cDceF9041Dffe7b8639BC8 StableV1 AMM - USDC/MIM 0x940a44Fe2b1c6BB0b21170995Fd9BD57b45a7CfA *YieldWolfPassthrough.cartographer* - Variable, no setter
IMPACT	● receives transfer of tokens deposited by users

YieldWolf Vaults

ADDRESS	0x876F890135091381c23Be437fA1cec2251B7c117
USAGE	Spooky BSHARE-FTM LP 0xbA74A5C08Ee6B8D63Cd51A80D236602637feBB71 Spooky BASED-TOMB LP 0x65810243e044a532272994856643078E65ef9611 VolatileV1 AMM - BOO/xBOO 0x77bF5EBc3912a091E1cDceF9041Dffe7b8639BC8 StableV1 AMM - USDC/MIM 0x940a44Fe2b1c6BB0b21170995Fd9BD57b45a7CfA *YieldWolfPassthrough.yieldWolf* - Variable, no setter
IMPACT	● receives transfer of tokens deposited by users

External Tokens

These contracts are not part of the audit scope.

Spooky BSHARE-FTM LP

ADDRESS	0x6F607443DC307DCBe570D0ecFf79d65838630B56
USAGE	Spooky BSHARE-FTM LP 0xbA74A5C08Ee6B8D63Cd51A80D236602637feBB71 *YieldWolfPassthrough.passthroughToken* - Variable, no setter *YieldWolfPassthrough.token* - Variable, no setter
IMPACT	● ERC20 Token

Spooky BASED-TOMB LP

ADDRESS	0xaB2ddCBB346327bBDF97120b0dD5eE172a9c8f9E
USAGE	Spooky BASED-TOMB LP 0x65810243e044a5322729948566643078E65ef9611 *YieldWolfPassthrough.passthroughToken* - Variable, no setter *YieldWolfPassthrough.token* - Variable, no setter
IMPACT	● ERC20 Token

VolatileV1 AMM - BOO/xBOO

ADDRESS	0x5804F6C40f44cF7593F73cf3aa16F7037213A623
USAGE	VolatileV1 AMM - BOO/xBOO 0x77bF5EBc3912a091E1cDceF9041Dffe7b8639BC8 *YieldWolfPassthrough.passthroughToken* - Variable, no setter *YieldWolfPassthrough.token* - Variable, no setter
IMPACT	● ERC20 Token

StableV1 AMM - USDC/MIM

ADDRESS	0x940a44Fe2b1c6BB0b21170995Fd9BD57b45a7CfA
USAGE	StableV1 AMM - USDC/MIM 0x940a44Fe2b1c6BB0b21170995Fd9BD57b45a7CfA *YieldWolfPassthrough.passthroughToken* - Variable, no setter

	YieldWolfPassthrough.token - Variable, no setter
IMPACT	- ERC20 Token

Appendix A - Reviewed Documents

Document	**Address**
interfaces/IPassthrough.sol	N/A
YieldWolfPassthrough.sol	Spooky BSHARE-FTM LP 0xbA74A5C08Ee6B8D63Cd51A80D236602637feBB71 Spooky BASED-TOMB LP 0x65810243e044a53227299485664307 8E65ef9611 VolatileV1 AMM - BOO/xBOO 0x77bF5EBc3912a091E1cDceF9041Dffe7b8639BC8 StableV1 AMM - USDC/MIM 0x940a44Fe2b1c6BB0b21170995Fd9BD57b45a7CfA

Revisions

Revision 1	3f0a2c6a5cf193dd1c1cf3ad53f31a2321acaecb
Revision 2	1eb72d5852b287060b1b1551b2c0906b88c9f0f7
Revision 3	da875a482e5c36169e82a7993ef4d93b153df3d3

Imported Contracts

OpenZeppelin	4.3.0

Appendix B - Risk Ratings

Risk	Description
High Risk	A fatal vulnerability that can cause the loss of all Tokens / Funds.
Medium Risk	A vulnerability that can cause the loss of some Tokens / Funds.
Low Risk	A vulnerability which can cause the loss of protocol functionality.
Informational	Non-security issues such as functionality, style, and convention.

Appendix C - Finding Statuses

Closed	Contracts were modified to permanently resolve the finding.
Mitigated	The finding was resolved by other methods such as revoking contract ownership. The issue may require monitoring, for example in the case of a time lock.
Partially Closed	Contracts were updated to fix the issue in some parts of the code.
Partially Mitigated	Fixed by project specific methods which cannot be verified on chain. Examples include compounding at a given frequency.
Open	The finding was not addressed.

Appendix D - Audit Procedure

A typical Obelisk audit uses a combination of the three following methods:

Manual analysis consists of a direct inspection of the contracts to identify any security issues. Obelisk auditors use their experience in software development to spot vulnerabilities. Their familiarity with common contracts allows them to identify a wide range of issues in both forked contracts as well as original code.

Static analysis is software analysis of the contracts. Such analysis is called “static” as it examines the code outside of a runtime environment. Static analysis is a powerful tool used by auditors to identify subtle issues and to verify the results of manual analysis.

On-chain analysis is the audit of the contracts as they are deployed on the block-chain. This procedure verifies that:

- deployed contracts match those which were audited in manual/static analysis;
- contract values are set to reasonable values;
- contracts are connected so that interdependent contract function correctly;
- and the ability to modify contract values is restricted via a timelock or DAO mechanism. (We recommend a timelock value of at least 72 hours)

Each obelisk audit is performed by at least two independent auditors who perform their analysis separately.

After the analysis is complete, the auditors will make recommendations for each issue based on best practice and industry standards. The project team can then resolve the issues, and the auditors will verify that the issues have been resolved with no new issues introduced.

Our auditing method lays a particular focus on the following important concepts:

- Quality code and the use of best practices, industry standards, and thoroughly tested libraries.
- Testing the contract from different angles to ensure that it works under a multitude of circumstances.
- Referencing the contracts through databases of common security flaws.

Follow Obelisk Auditing for the Latest Information

ObeliskOrg ObeliskOrg

OBELISK

Part of Tibereum Group