OBELISK

Part of Tibereum Group

AUDITING REPORT

Version Notes

Version	No. Pages	Date	Revised By	Notes
1.0	Total: 26	2021-12-17	Zapmore, Donut	Audit Final

Audit Notes

Audit Date	2021-11-14 - 2021-12-16
Auditor/Auditors	Donut, Zenith
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	OB599365478

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.

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

Version Notes 2

Audit Notes 2

Disclaimer 2

Obelisk Auditing 3

Audit Information 3

Project Information 5

Audit of 1Swap 6

- Summary Table 7

Findings 8

- Manual Analysis 8
  - Base Pool Liquidity Is Left In Router 8
  - Pools Are Implied To Be Part Of Other Pools 9
  - Amplification Factor Does Not Match Whitepaper Calculations 10
  - Fee Changes Based On Token Count 12
  - Admin Can Claim All Fees 13
  - Changing Amplification Coefficient Of Can Cause Loss Of Funds 14
  - Pool Can Have Duplicate Tokens 16
- Static Analysis 17
  - Older Solidity Version 17
- On-Chain Analysis 18
  - Low Timelock Delay 18
  - Fee Distributor Is Externally Owned Account 19
  - No Timelock 20
  - Modified Timelock 21

Appendix A - Reviewed Documents 22

- Revisions 22
- Imported Contracts 22
- Externally Owned Accounts 23
- External Contracts 23

Appendix B - Risk Ratings 24

Appendix C - Finding Statuses 24

Appendix D - Audit Procedure 25

Project Information

Name	1Swap
Description	1Swap Finance is building a decentralized application ecosystem on the Moonriver Ecosystem.
Website	https://1swap.fi/
Contact	@nxtflow on TG
Contact information	@nxtflow on TG
Token Name(s)	N/A
Token Short	N/A
Contract(s)	See Appendix A
Code Language	Solidity
Chain	Moonriver

Audit of 1Swap

Obelisk was commissioned by 1Swap on the 2nd of November 2021 to conduct a comprehensive audit of 1Swaps' contracts. The following audit was conducted between the 14th of November 2021 and the 16th of December 2021. 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.

There was a mixture of findings during the audit of 1Swaps' contracts. The most severe of those were solved quickly by the project team. Issue #2 and issue #10 are both issues with a Medium Risk level. Issue #10 was solved on-chain by deploying a treasury (which is not part of the audit) that handles the collected fees. Findings #2 was not solved, but the project team left a comment for the reader to make their own assessment of the situation.

There were multiple Low-Risk findings and all of these were solved or mitigated besides issue #4 which is still open. Keep in mind the timelock on issue #9 is only 12hours long compared to the recommended 72 hours.

The informational findings are good to know while interacting with the project but don't directly impact 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 1Swap project**.**

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

Summary Table

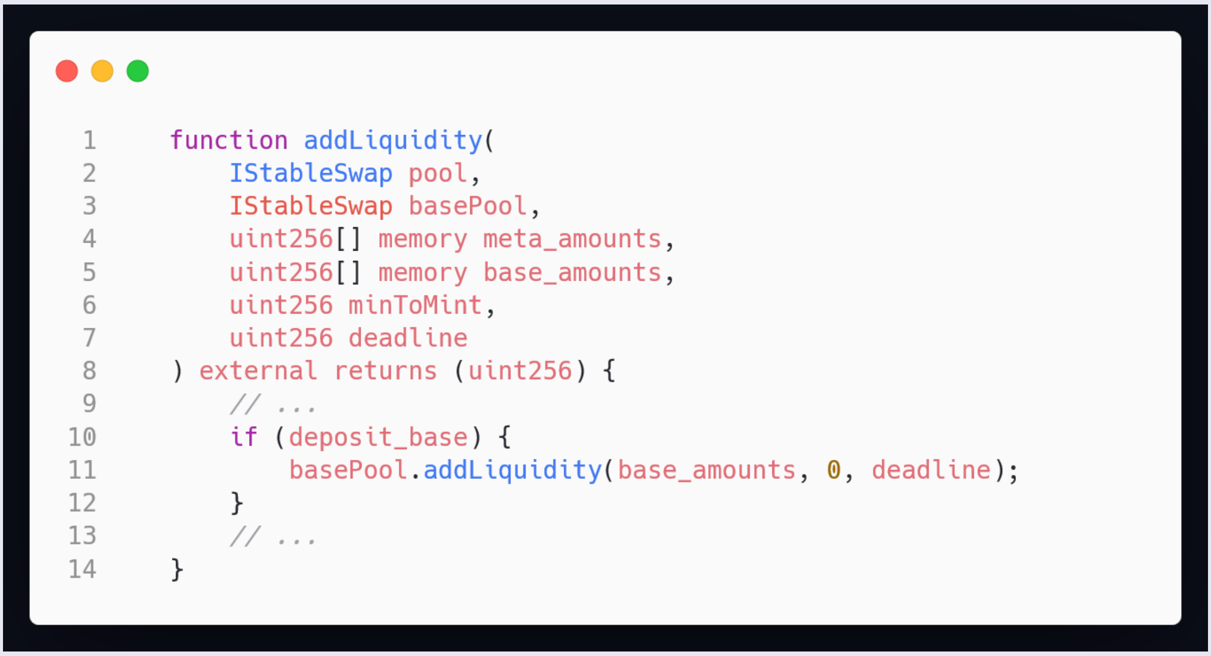
Finding	ID	Severity	Status
Base Pool Liquidity Is Left In Router	#0001	High Risk	Closed
Pools Are Implied To Be Part Of Other Pools	#0002	Medium Risk	Open
Amplification Factor Does Not Match Whitepaper Calculations	#0003	Low Risk	Closed
Fee Changes Based On Token Count	#0004	Low Risk	Open
Admin Can Claim All Fees	#0005	Low Risk	Mitigated
Changing Amplification Coefficient Of Can Cause Loss Of Funds	#0006	Informational	Open
Pool Can Have Duplicate Tokens	#0007	Informational	Open
Older Solidity Version	#0008	Informational	Open
Low Timelock Delay	#0009	Low Risk	Open
Fee Distributor Is Externally Owned Account	#0010	Medium Risk	Mitigated
No Timelock	#0011	Low Risk	Mitigated
Modified Timelock	#0012	Informational	Mitigated

Findings

Manual Analysis

Base Pool Liquidity Is Left In Router

FINDING ID	#0001
SEVERITY	High Risk
STATUS	Closed
LOCATION	StableSwapRouter.sol -> 50-90



```
function addLiquidity(
    IStableSwap pool,
    IStableSwap basePool,
    uint256[] memory meta_amounts,
    uint256[] memory base_amounts,
    uint256 minToMint,
    uint256 deadline
) external returns (uint256) {
    // ...
    if (deposit_base) {
        basePool.addLiquidity(base_amounts, 0, deadline);
    }
    // ...
}
```

DESCRIPTION	Adding liquidity via the router allows adding liquidity to a pool and a base pool. The amount added to the base pool is not returned to the user or required to be a component of the primary pool. This base liquidity can be retrieved from the router at a later time by any other user.
RECOMMENDATION	Ensure that the base pool is part of the primary pool or that unused amounts are returned.
RESOLUTION	A check was added to ensure that all base pool tokens created during the first add liquidity step are deposited into the second pool.

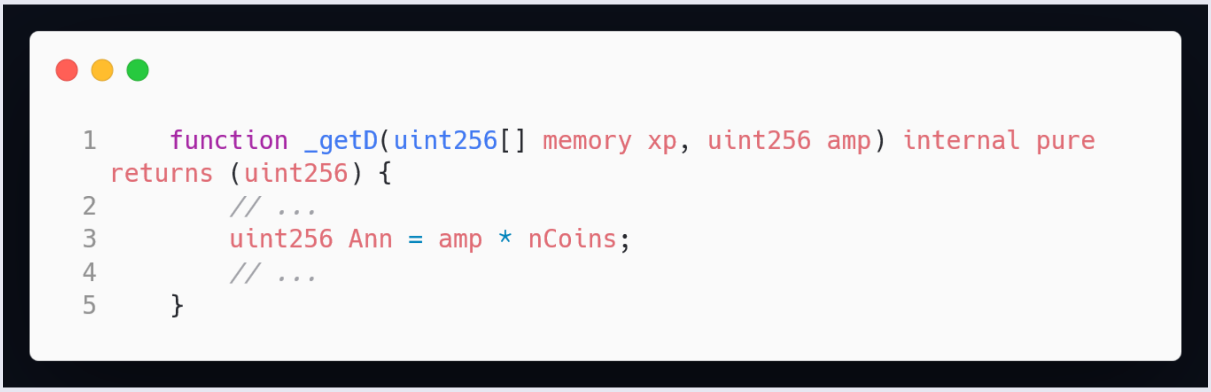
Pools Are Implied To Be Part Of Other Pools

FINDING ID	#0002
SEVERITY	Medium Risk
STATUS	Open
LOCATION	StableSwapRouter.sol

DESCRIPTION	The StableSwap invariant assumes that the relative value of pooled tokens is effectively identical. However, the router implies that pool tokens can be used as components of other pools. Because pools accumulate fees, the pool tokens will change in value relative to other tokens. This will likely lead to the loss of liquidity as the pool can stabilize far from the balanced point.
RECOMMENDATION	Do not use stableswap pools as components of other pools.
RESOLUTION	Project Team Comment: “The 1swap approach is the same with Curve's basepool being a part of Curve's meta pool”

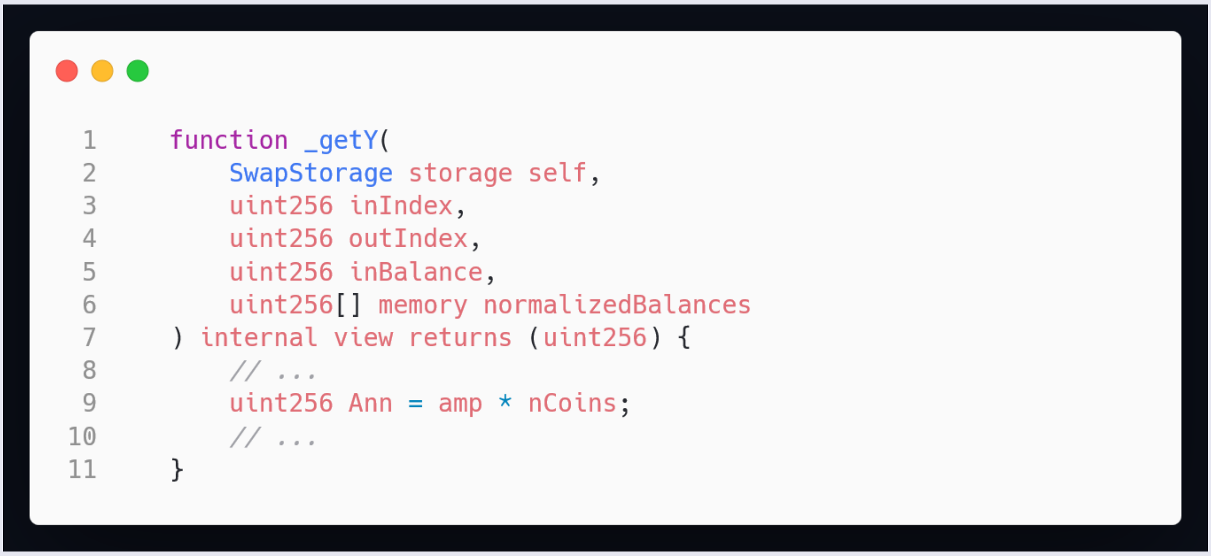
Amplification Factor Does Not Match Whitepaper Calculations

FINDING ID	#0003
SEVERITY	Low Risk
STATUS	Closed
LOCATION	StableSwapStorage.sol -> 388-417



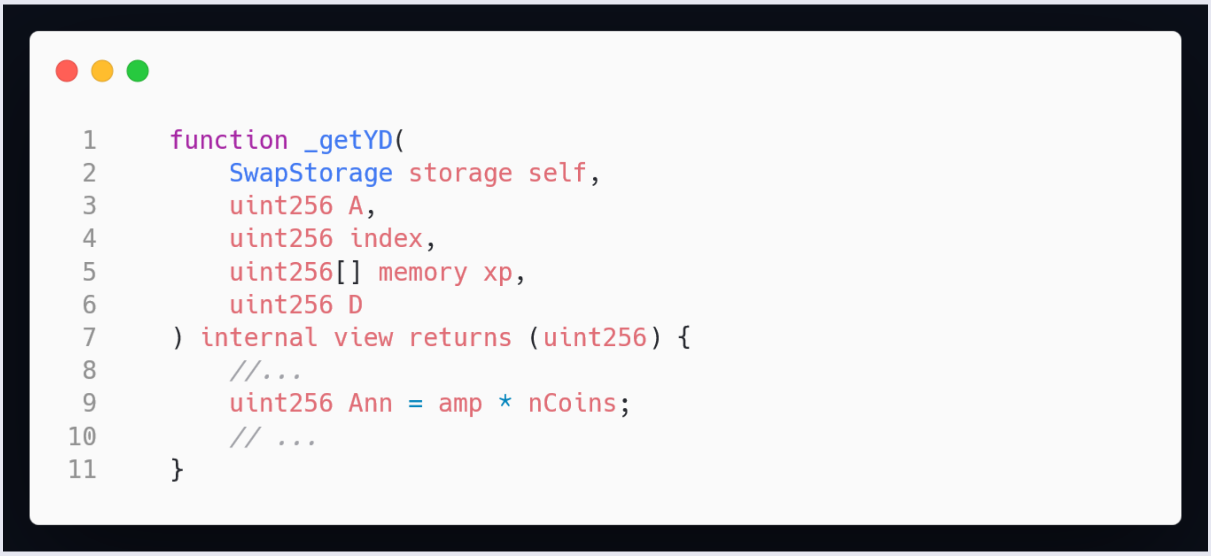
```
function _getD(uint256[] memory xp, uint256 amp) internal pure returns (uint256) {
    // ...
    uint256 Ann = amp * nCoins;
    // ...
}
```

LOCATION	StableSwapStorage.sol -> 430-472



```
function _getY(
    SwapStorage storage self,
    uint256 inIndex,
    uint256 outIndex,
    uint256 inBalance,
    uint256[] memory normalizedBalances
) internal view returns (uint256) {
    // ...
    uint256 Ann = amp * nCoins;
    // ...
}
```

LOCATION	StableSwapStorage.sol -> 524-560

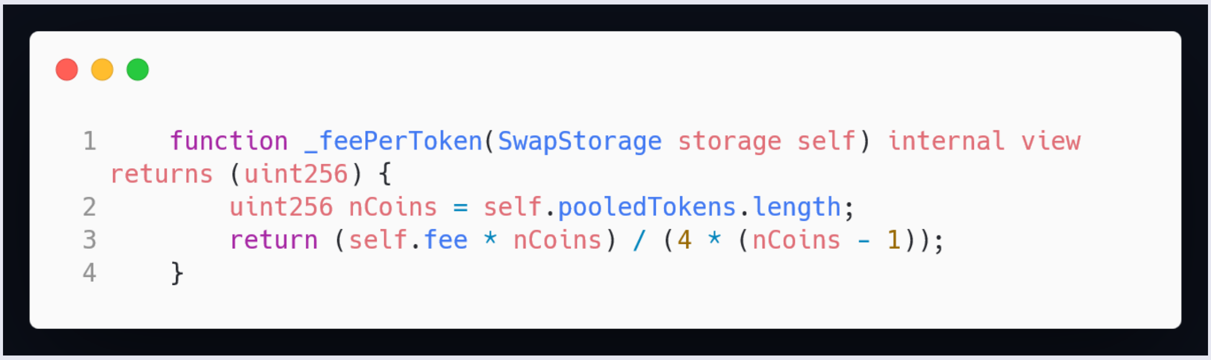


```
function _getYD(
    SwapStorage storage self,
    uint256 A,
    uint256 index,
    uint256[] memory xp,
    uint256 D
) internal view returns (uint256) {
    //...
    uint256 Ann = amp * nCoins;
    // ...
}
```

DESCRIPTION	The calculation of the swap invariant does not match the equation described in the stableswap white paper. In particular, the value of Ann is supposed to be A * n^n, not A * n. This will cause the amplification factor to act as if it was smaller than intended. The implementation noted matches the implementation used in Curve's vyper implementation. Refer to: ● StableSwap whitepaper ● Crypto pools whitepaper ● Curve vyper implementation
RECOMMENDATION	Fix the calculation of the amplification factor or confirm that this is the intended behavior.
RESOLUTION	Project Team Comment: “intended behavior”

Fee Changes Based On Token Count

FINDING ID	#0004
SEVERITY	Low Risk
STATUS	Open
LOCATION	[StableSwapStorage.sol -> 519-522]()

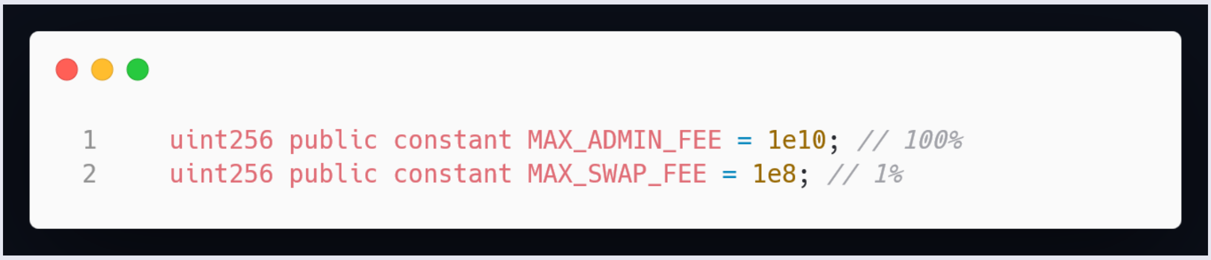


```
    function _feePerToken(SwapStorage storage self) internal view returns (uint256) {
        uint256 nCoins = self.pooledTokens.length;
        return (self.fee * nCoins) / (4 * (nCoins - 1));
    }
```

DESCRIPTION	The total fee varies based on the number of tokens in the pool. The base fee will be multiplied by the following factors based on the number of tokens in the pool: 2: 1/2 3: 3/8 4: 1/3 5: 5/16
RECOMMENDATION	Clarify the purpose of this variable fee rate and confirm that it is intended.
RESOLUTION	N/A

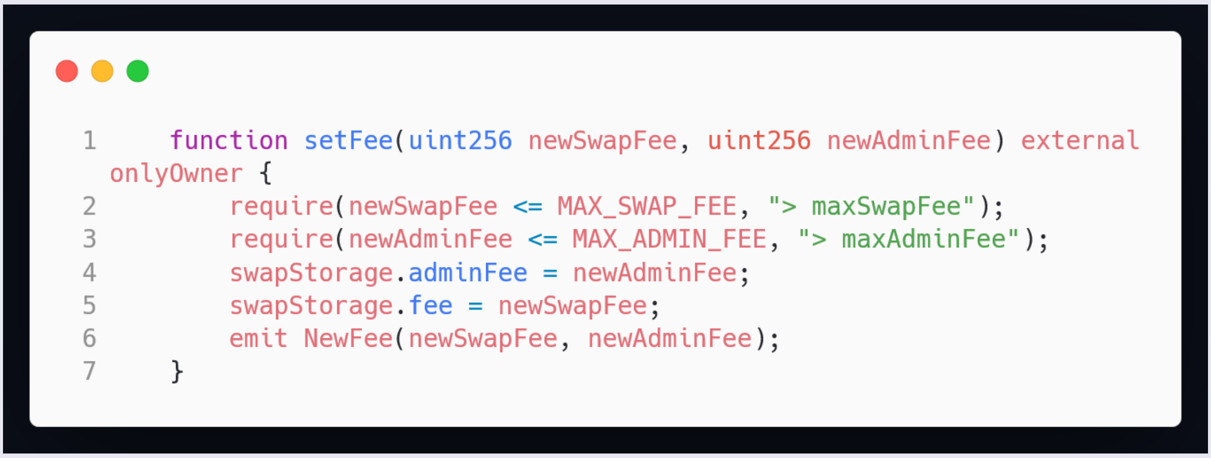
Admin Can Claim All Fees

FINDING ID	#0005
SEVERITY	Low Risk
STATUS	Mitigated
LOCATION	StableSwap.sol -> 20-21



```
uint256 public constant MAX_ADMIN_FEE = 1e10; // 100%
uint256 public constant MAX_SWAP_FEE = 1e8; // 1%
```

LOCATION	StableSwap.sol -> 209-215

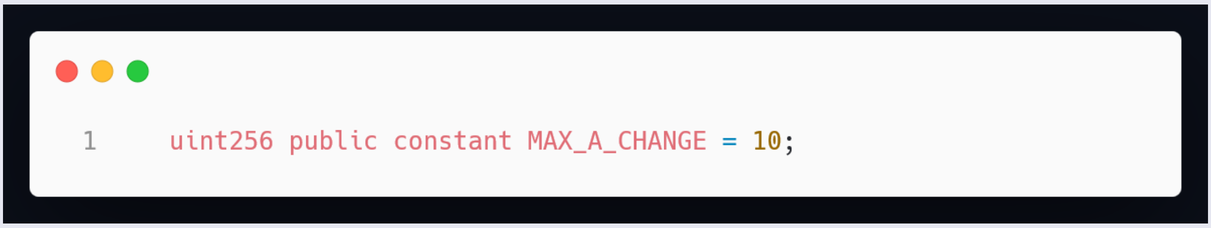


```
function setFee(uint256 newSwapFee, uint256 newAdminFee) external onlyOwner {
    require(newSwapFee <= MAX_SWAP_FEE, "> maxSwapFee");
    require(newAdminFee <= MAX_ADMIN_FEE, "> maxAdminFee");
    swapStorage.adminFee = newAdminFee;
    swapStorage.fee = newSwapFee;
    emit NewFee(newSwapFee, newAdminFee);
}
```

DESCRIPTION	Max admin fee may be set such that liquidity providers do not get any fees.
RECOMMENDATION	Consider reducing *MAX_ADMIN_FEE* such that it cannot be 100% of fees, or utilize a timelock such that transparency is provided on team fee-share. A timelock should be included regardless to ensure that users have ample notice before major protocol parameters are changed (such as alterations to A).
RESOLUTION	Project team has stated that this is intended behaviour. Project Team Comment: “the admin fee is 100% since the very beginning of project launch. we want to grow protocol-owned liquidity. Liquidity providers earn by farming 1swap, not by collecting swap fees.”

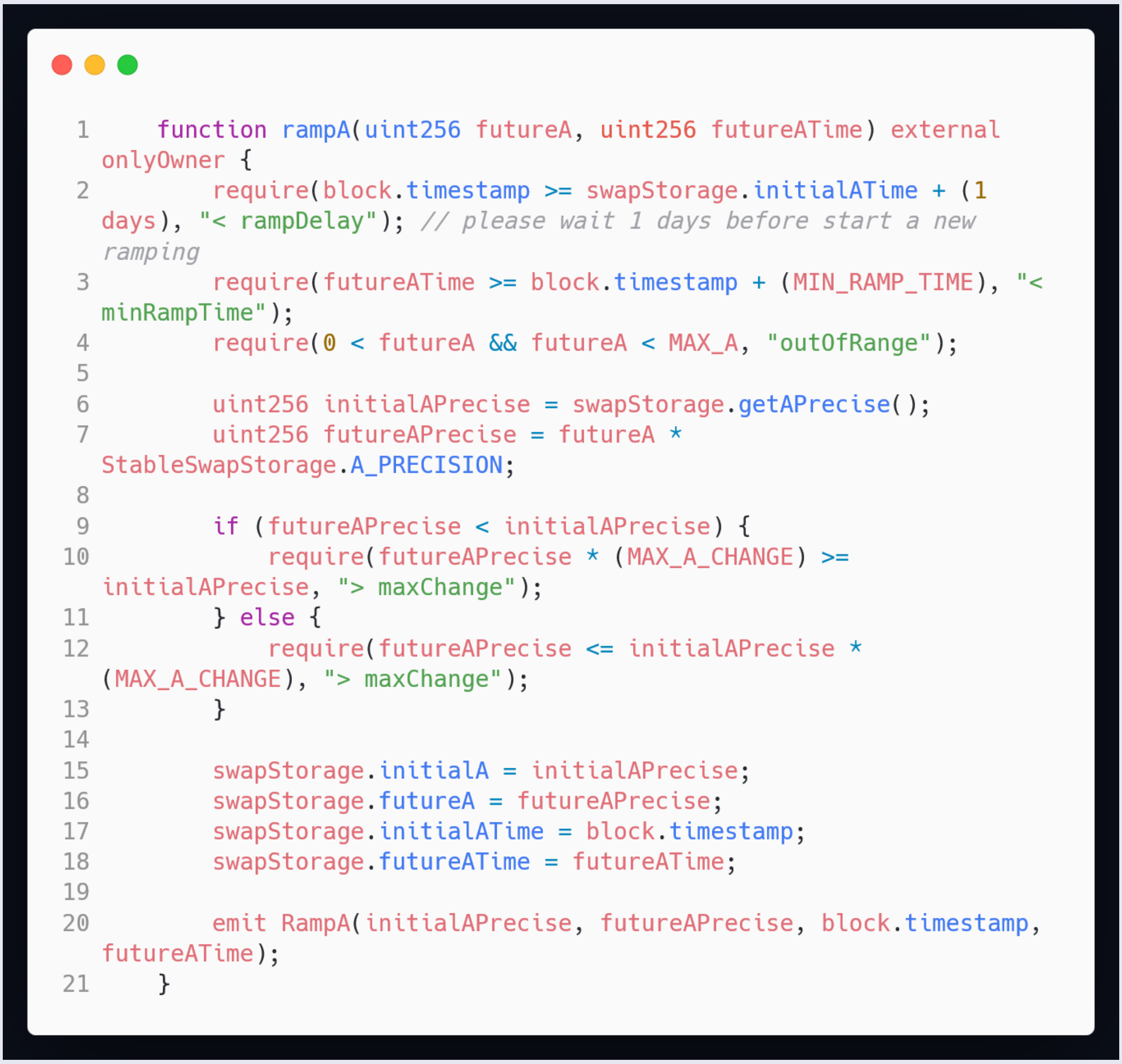
Changing Amplification Coefficient Of Can Cause Loss Of Funds

FINDING ID	#0006
SEVERITY	Informational
STATUS	Open
LOCATION	StableSwap.sol -> 19



```
    uint256 public constant MAX_A_CHANGE = 10;
```

LOCATION	StableSwap.sol -> 224-244

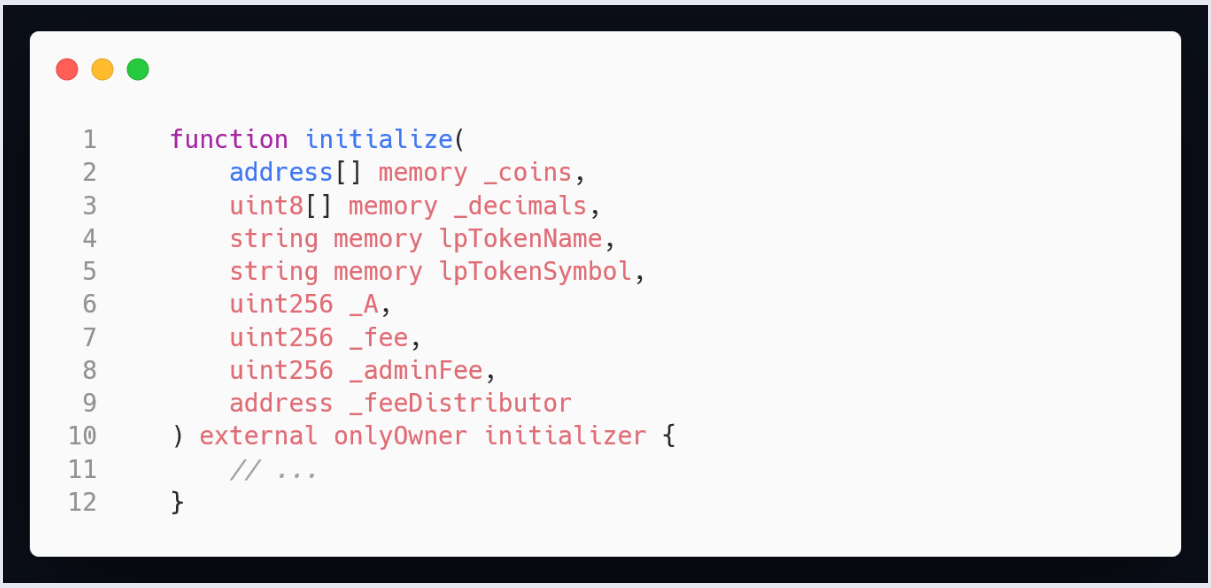


```
    function rampA(uint256 futureA, uint256 futureATime) external onlyOwner {
        require(block.timestamp >= swapStorage.initialATime + (1 days), "< rampDelay"); // please wait 1 days before start a new ramping
        require(futureATime >= block.timestamp + (MIN_RAMP_TIME), "< minRampTime");
        require(0 < futureA && futureA < MAX_A, "outOfRange");

        uint256 initialAPrecise = swapStorage.getAPrecise();
        uint256 futureAPrecise = futureA * StableSwapStorage.A_PRECISION;

        if (futureAPrecise < initialAPrecise) {
            require(futureAPrecise * (MAX_A_CHANGE) >= initialAPrecise, "> maxChange");
        } else {
            require(futureAPrecise <= initialAPrecise * (MAX_A_CHANGE), "> maxChange");
        }

        swapStorage.initialA = initialAPrecise;
        swapStorage.futureA = futureAPrecise;
        swapStorage.initialATime = block.timestamp;
        swapStorage.futureATime = futureATime;

        emit RampA(initialAPrecise, futureAPrecise, block.timestamp, futureATime);
    }
```

DESCRIPTION	The rate of change of the amplification coefficient is limited to a factor of 10 per day. If the rate changes significantly while the pool is unbalanced, the value of the pool can effectively decrease, resulting in the loss of user funds. Refer to: https://medium.com/@peter_4205/curve-vulnerability-report-a1d7630140ec
RECOMMENDATION	No changes required
RESOLUTION	The project team has acknowledged the finding.

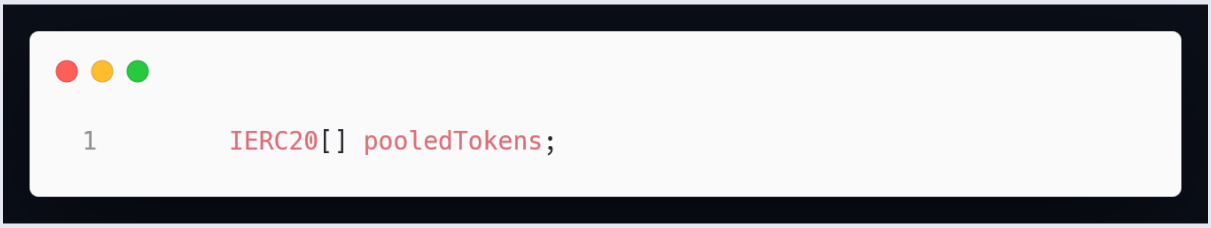
Pool Can Have Duplicate Tokens

FINDING ID	#0007
SEVERITY	Informational
STATUS	Open
LOCATION	[StableSwap.sol -> 39-75]



```
function initialize(
    address[] memory _coins,
    uint8[] memory _decimals,
    string memory lpTokenName,
    string memory lpTokenSymbol,
    uint256 _A,
    uint256 _fee,
    uint256 _adminFee,
    address _feeDistributor
) external onlyOwner initializer {
    // ...
}
```

LOCATION	StableSwapStorage.sol -> 53



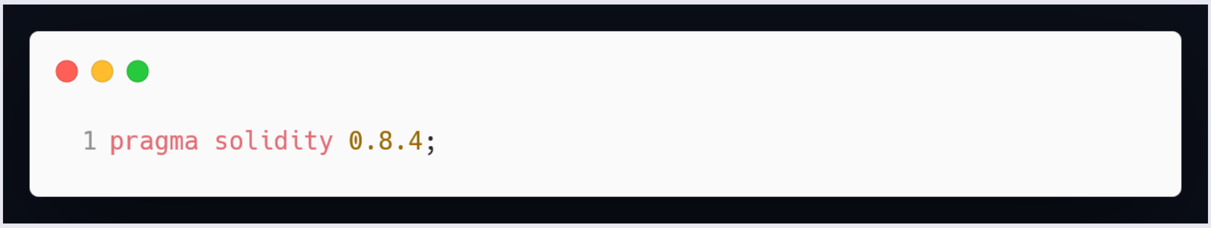
```
    IERC20[] pooledTokens;
```

DESCRIPTION	No checks are provided to ensure that a pooled token is not duplicated. While the swap mechanism is able to support such a situation, it is likely to be confusing.
RECOMMENDATION	Ensure that pools with duplicate tokens are not deployed.
RESOLUTION	The project team has acknowledged the finding. Note: Duplicate tokens will be checked on-chain and the finding will be changed to “mitigated” upon confirmation.

Static Analysis

Older Solidity Version

FINDING ID	#0008
SEVERITY	Informational
STATUS	Open
LOCATION	All files



```
pragma solidity 0.8.4;
```

DESCRIPTION	Later stable solidity releases typically include a number of bug fixes. As such, it is preferable to take the latest stable release of a major solidity version.
RECOMMENDATION	Update pragmas to *pragma solidity 0.8.10*.
RESOLUTION	N/A

On-Chain Analysis

Low Timelock Delay

FINDING ID	#0009
SEVERITY	Low Risk
STATUS	Open
LOCATION	Timelock 0x2f681d381072d4B66B3e98729158959Bc6Ea2329

DESCRIPTION	The timelock contract has a 12-hour delay. Obelisk recommends a delay of at least 72 hours.
RECOMMENDATION	Increase the timelock delay.
RESOLUTION	N/A

Fee Distributor Is Externally Owned Account

FINDING ID	#0010
SEVERITY	Medium Risk
STATUS	Mitigated
LOCATION	StableSwapSwap - 1S3P LP token 0xb578a396e56388CbF398a12Dea9eb6B01b7c777f StableSwapSwap - BUSD/1S3P Metapool 0x008db1Cef0958e7f87A107b58F0dede796ce7962 StableSwapSwap - FRAX/1S3P Metapool 0xB7900b1824f84C9c5043A799D4Eb4053FEcdeF0b StableSwapSwap - MIM/1S3P Metapool 0x23A479A83e4FaC12C2096Ab1D79Ea7a788f4489E StableSwapSwap - AVAXUSD/1S3P Metapool 0x7179F2C31763f395082489588534F4abb3Dd4Be6 StableSwapSwap - WANUSD/1S3P Metapool 0x02A105939Dc0C47cb6bD04f320dAa77Bd9E3Bb0D

DESCRIPTION	The fee distributor is an externally owned account. Admin/FeeDistributor 0xB1aC902b5D81D58739a1404b05ff34722c4d3C71 Documentation suggests that the fees will be sent to a treasury address. https://docs.1swap.fi/products/1swap-dex
RECOMMENDATION	Set the fee distributor to a treasury contract.
RESOLUTION	The fee distributor was set to a treasury contract. Treasury address: 0x7121Ff977f1D152D307e33145EE436884d1F8632 Note: Treasury not included in audit scope.

No Timelock

FINDING ID	#0011
SEVERITY	Low Risk
STATUS	Mitigated
LOCATION	Stableswap - FRAX/1S3P Metapool 0xB7900b1824f84C9c5043A799D4Eb4053FEcdeF0b Stableswap - WANUSD/1S3P Metapool 0x02A105939Dc0C47cb6bD04f320dAa77Bd9E3Bb0D

DESCRIPTION	The noted contracts have not had their ownership transferred to the timelock contract yet.
RECOMMENDATION	Transfer ownership to the timelock contract.
RESOLUTION	Ownership of the contracts was transferred to the timelock.

Modified Timelock

FINDING ID	#0012
SEVERITY	Informational
STATUS	Mitigated
LOCATION	Timelock 0x2f681d381072d4B66B3e98729158959Bc6Ea2329

DESCRIPTION	The timelock was modified from the typical timelock contract with an additional "proposer" mechanism. Addresses marked as "proposers" may queue transactions on the timelock, but cannot execute them. No proposers other than the existing admin address were observed.
RECOMMENDATION	Do not add additional proposers. Ensure that the old admin is removed from the proposers if the admin is transferred.
RESOLUTION	N/A

Appendix A - Reviewed Documents

Document	**Address**
interfaces/IStableSwap.sol	N/A
stableswap/LPToken.sol	1S3P LP token 0x17da5445F3Cd02b3F1cD820E6DE55983fe80CF85 BUSD/1S3P Metapool 0x2D5913437accb1119dd82E7584942fed3574F034 FRAX/1S3P Metapool 0xD9E781d93cc29155C4506f7906f9ba39e2e04573 MIM/1S3P Metapool 0xaa78E3e69068b83A060377591FbB4598Fa9e4737 AVAXUSD/1S3P Metapool 0x5D57CD76d3Fc5a3ec8B94FdF16Ff6aa7340140fE WANUSD/1S3P Metapool 0x03C1695815b6619e2377a14Ad63831c7a9AC198E
stableswap/OwnerPausable.sol	N/A
stableswap/StableSwap.sol	1S3P LP token 0xb578a396e56388CbF398a12Dea9eb6B01b7c777f BUSD/1S3P Metapool 0x008db1Cef0958e7f87A107b58F0dede796ce7962
stableswap/StableSwapRouter.sol	0x3A9364357E4Acfe0Bc930B87377fCbE02DD6cb19
stableswap/StableSwapStorage.sol	N/A
Timelock	0x2f681d381072d4B66B3e98729158959Bc6Ea2329

Revisions

Revision 1	570a2c0d4cf7fdbb1468e7e7caa79a5d79e28398
Revision 2	c1c8c278cadb2132b7fdc2df53468b2773f5d079

Imported Contracts

OpenZeppelin	4.3.0

Externally Owned Accounts

Admin/FeeDistributor	0xB1aC902b5D81D58739a1404b05ff34722c4d3C71

External Contracts

These contracts are not part of the audit scope.

Contract	**Address**
1Swap Treasury	0x7121Ff977f1D152D307e33145EE436884d1F8632
BUSD	0x5D9ab5522c64E1F6ef5e3627ECCc093f56167818
DAI	0x80A16016cC4A2E6a2CACA8a4a498b1699fF0f844
FRAX	0x1A93B23281CC1CDE4C4741353F3064709A16197d
MIM	0x0caE51e1032e8461f4806e26332c030E34De3aDb
USDC	0xE3F5a90F9cb311505cd691a46596599aA1A0AD7D
USDT	0xB44a9B6905aF7c801311e8F4E76932ee959c663C
DAI.e	0x26dFff76D9123A1C79279AbC29B676c48A8BD77e
USDC.e	0xD8B99eae34afDF1a9bFA5770066404ee4468d0f2
USDT.e	0xf97C8556Af29089D5d1627096958187b11F1915C
USDC.m	0x748134b5F553F2bcBD78c6826De99a70274bDEb3
USDT.m	0xE936CAA7f6d9F5C9e907111FCAf7c351c184CDA7

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