Glow Swap

Audit Report

Mon Jan 22 2024

contact@scalebit.xyz https://twitter.com/scalebit_

ScaleBit

Glow Swap Audit Report

1 Executive Summary

1.1 Project Information

Description	A concentrated liquidity swap protocol.
Type	Dex
Auditors	ScaleBit
Timeline	Mon Dec 12 2022 – Fri Jan 19 2024
Languages	Solidity
Platform	B^2 Network
Methods	Architecture Review, Unit Testing, Manual Review
Source Code	https://github.com/glowswap/glow-contracts
Commits	98d5f7090fd2c7658748bcc707b30646cd258a76

1.2 Files in Scope

The following are the SHA1 hashes of the original reviewed files.

ID	File	SHA-1 Hash
TIC	projects/v3-core/contracts/libraries/Tick.sol	37ef664ced74a41e7a2f438cdbf99527566f1aab
LGSM	projects/v3-core/contracts/libraries/LowGasSafeMath.sol	1bee2d0f85bc054e3b63a7e92c67d237a49c650c
SCA	projects/v3-core/contracts/libraries/SafeCast.sol	c3b25ed7fa205de6cc2075d96e27908d43f21671
FP9	projects/v3-core/contracts/libraries/FixedPoint96.sol	3a3ab5c10385c523c1738b9eb9d86dcd5f59c3f4
POS	projects/v3-core/contracts/libraries/Position.sol	0d6be19a8ba07321743fc90010a969b0fe26e301
FMA	projects/v3-core/contracts/libraries/FullMath.sol	0c531e95498282fc6ad5856e6273b7675b15bea0
SMA	projects/v3-core/contracts/libraries/SwapMath.sol	585ec272ca9a5a9b5d4645178b64fb52003e6091
ORA	projects/v3-core/contracts/libraries/Oracle.sol	49519e7e73e076479b04d0027d342468126e4cba
LMA	projects/v3-core/contracts/libraries/LiquidityMath.sol	2d440d1d862d4612b08243581f9232887b489c09
FP1	projects/v3-core/contracts/libraries/FixedPoint128.sol	22517ba8d668bb4e86a45f3f29ed077d72fb7608
THE	projects/v3-core/contracts/libraries/TransferHelper.sol	09f4e335c7ed41383bf2f04bf278169218994fc8
TBI	projects/v3-core/contracts/libraries/TickBitmap.sol	f14dad9bee719bffd0bd7fc54d2da37f289561d8

TMA	projects/v3-core/contracts/libraries/TickMath.sol	7eee6a798a068e6eaaa63ce8f432ee193e0ff2e0
UMA	projects/v3-core/contracts/libraries/UnsafeMath.sol	d3e3ff1ab78e03ccec335ab6da4ea76b578cb422
BMA	projects/v3-core/contracts/libraries/BitMath.sol	82ee70afdc183819ee3705d274a506a42f1e278b
SPM	projects/v3-core/contracts/libraries/SqrtPriceMath.sol	0bf7a6c27c88689b4ade289bf0683adabe90a570
TRSO	projects/v3-periphery/contracts/libraries/TokenRatioSortOrder.sol	84ff0b5257a032c234bf53b3866a857edd30512b
PAT	projects/v3-periphery/contracts/libraries/Path.sol	2504b1a543392240bddbe04efef9c47cecdc704b
CID	projects/v3-periphery/contracts/libraries/ChainId.sol	a2ffce157a73a5b87024ed2bb54f9c3ae19b04c3
HST	projects/v3-periphery/contracts/libraries/HexStrings.sol	fc19854bf736b050ab6a78bb595cef7a43699b45
THE	projects/v3-periphery/contracts/libraries/TransferHelper.sol	6cedc556d3cf7b972f78e7c64670ebfbd7f4c9cc
BLI	projects/v3-periphery/contracts/libraries/BytesLib.sol	747be1412bfe71b5c06f4bbfa7cb7b2c968bfdcc
PKE	projects/v3-periphery/contracts/libraries/PositionKey.sol	6cc88dd5fd105faa25c6f048b0e7da4e50263c8b
MUL	projects/v3-periphery/contracts/base/Multicall.sol	e48264609451e31ffea549e7db3e30815080505c
BTI	projects/v3-periphery/contracts/base/BlockTimestamp.sol	e9433e812b02a43ae225b797863e5102e802ef27
PPA	projects/v3-periphery/contracts/base/PeripheryPayments.sol	ba48c46d36b30ed6efcb5809b92eb422d49f3f2d

ERC7P	projects/v3-periphery/contracts/base/ERC721Permit.sol	402f58139a0bdc704f6b5737075454601084dc9c
PVA	projects/v3-periphery/contracts/base/PeripheryValidation.sol	078495af30569dfdb02365ae8340f54d03b04c96
SPE	projects/v3-periphery/contracts/base/SelfPermit.sol	bea7d24d2f467a5ad0a34d9d07c2b0e868506fc6
PIS	projects/v3-periphery/contracts/base/PeripheryImmutableState.sol	238ba15bdc60250ead1a2f21c8307d175c9d0880
NTPDOC	projects/v3-periphery/contracts/NonfungibleTokenPositionDescriptorOffChain.sol	29c8fdc6b1d4cabc2a07a7b187d7e67a85687647
SCA	projects/masterchef-v3/contracts/libraries/SafeCast.sol	0cd843e1c910d1119af2322434690839ebf09547
MUL	projects/masterchef-v3/contracts/utils/Multicall.sol	8137902e1c2215f98bd78d6e5a49752945133822
ENU	projects/masterchef-v3/contracts/Enumerable.sol	3d4bcab22971615ffb50b69501cef461608db671
GV3P	projects/v3-core/contracts/GlowV3Pool.sol	968241b6109e9d6a151c9a49cf5004876205d702
GV3F	projects/v3-core/contracts/GlowV3Factory.sol	5f88e1af46490850966e4875e27479201c69748f
GV3PD	projects/v3-core/contracts/GlowV3PoolDeployer.sol	e8ade571de29723718728a5dbf0bf8dcc1fa8347
LAM	projects/v3-periphery/contracts/libraries/LiquidityAmounts.sol	d06fb9e57c4dd13e4cc0cb661d4299d63cc37706
OLI	projects/v3-periphery/contracts/libraries/OracleLibrary.sol	e3a5306e21f1097bc608d117d2299a04f1b41e29
PVA	projects/v3-periphery/contracts/libraries/PositionValue.sol	c303c28813caec514910d8d359de3442e5e6e7b2

SPMP	projects/v3-periphery/contracts/libraries/SqrtPriceMathPartial.sol	204f0a49b98bb99782f2e73e71b50c48a3414a06
NFTD	projects/v3-periphery/contracts/libraries/NFTDescriptor.sol	8f8059fd88b85ccbe73a058f2a5893cdc8c8212f
NFTSVG	projects/v3-periphery/contracts/libraries/NFTSVG.sol	f2b3936a9017f3934643a7f4ec65f0e509a793f3
CVA	projects/v3-periphery/contracts/libraries/CallbackValidation.sol	451d0e09a2b376dc540e686fb0a6073189a06a2c
PTC	projects/v3-periphery/contracts/libraries/PoolTicksCounter.sol	b311cab3d1e5dee156bde3a84554940e645da72f
PAD	projects/v3-periphery/contracts/libraries/PoolAddress.sol	f81feeee7c3cbce546c522ae704d0779556fc69c
QUO	projects/v3-periphery/contracts/lens/Quoter.sol	b37e9bf0c8bdb31eec6b7e84d0f7a46fe45c0fbe
GIM	projects/v3-periphery/contracts/lens/GlowInterfaceMulticall.sol	0115a49653ed9217de4e9911a5fba00250050cb3
TLE	projects/v3-periphery/contracts/lens/TickLens.sol	66f36c8a18270aba87faa0eaf043ada868024609
NPM	projects/v3-periphery/contracts/NonfungiblePositionManager.sol	1f60e6693c158e15aa7bfa382f63e4c046cdff6c
NFTDE	projects/v3-periphery/contracts/NFTDescriptorEx.sol	2000dc775ca2f5061049d38f422229f16bc9e772
SRO	projects/v3-periphery/contracts/SwapRouter.sol	724b830e21e95ac1bad6d0ec09d00f39dc6bd89a
NTPD	projects/v3-periphery/contracts/NonfungibleTokenPositionDescriptor.sol	56496d26c0e3bb785df59780c582d7a2d6dc65f6
PFL	projects/v3-periphery/contracts/examples/PairFlash.sol	6e74c02867f685e0ea95784103cdf825048d1070

LMA1	projects/v3-periphery/contracts/base/LiquidityManagement.sol	3635748407fd5b781b7ba515f789ad43c60ef6b6
PIN	projects/v3-periphery/contracts/base/PoolInitializer.sol	ee977e20acb18d4dfd6c26e46ff311ab40067a12
PPWF	projects/v3-periphery/contracts/base/PeripheryPaymentsWithFee.sol	f0a2eb425e42c7da05943671209351646e225656
THE2	projects/masterchef-v3/contracts/libraries/TransferHelper.sol	7e5dfe7c4f58bafff0b92bfaa1196df7e2502925
MCV3	projects/masterchef-v3/contracts/MasterChefV3.sol	31b6d42248cc38d5883fad51a35af6366d37f356
VAU	projects/masterchef-v3/contracts/Vault.sol	ec011e084847b6fa8bbf11f49e3e0427df0c081b
LTI	projects/v3-lm-pool/contracts/libraries/LmTick.sol	5097523544eca8daecd0d42adad27184d203f5e1
GV3LP	projects/v3-lm-pool/contracts/GlowV3LmPool.sol	2061ea80a70cac811b2354b74d3e1f90228d3d81
GV3LPD	projects/v3-lm-pool/contracts/GlowV3LmPoolDeployer.sol	5051ccae436fdaa1b6a530e593ed49b63a6c2454

1.3 Issue Statistic

Item	Count	Fixed	Acknowledged
Total	3	3	0
Informational	1	1	0
Minor	1	1	0
Medium	0	0	0
Major	1	1	0
Critical	0	0	0

1.4 ScaleBit Audit Breakdown

ScaleBit aims to assess repositories for security-related issues, code quality, and compliance with specifications and best practices. Possible issues our team looked for included (but are not limited to):

- Transaction-ordering dependence
- Timestamp dependence
- Integer overflow/underflow
- Number of rounding errors
- Unchecked External Call
- Unchecked CALL Return Values
- Functionality Checks
- Reentrancy
- Denial of service / logical oversights
- Access control
- Centralization of power
- Business logic issues
- Gas usage
- Fallback function usage
- tx.origin authentication
- Replay attacks
- Coding style issues

1.5 Methodology

The security team adopted the **"Testing and Automated Analysis"**, **"Code Review"** and **"Formal Verification"** strategy to perform a complete security test on the code in a way that is closest to the real attack. The main entrance and scope of security testing are stated in the conventions in the "Audit Objective", which can expand to contexts beyond the scope according to the actual testing needs. The main types of this security audit include:

(1) Testing and Automated Analysis

Items to check: state consistency / failure rollback / unit testing / value overflows / parameter verification / unhandled errors / boundary checking / coding specifications.

(2) Code Review

The code scope is illustrated in section 1.2.

(3) Audit Process

- Carry out relevant security tests on the testnet or the mainnet;
- If there are any questions during the audit process, communicate with the code owner in time. The code owners should actively cooperate (this might include providing the latest stable source code, relevant deployment scripts or methods, transaction signature scripts, exchange docking schemes, etc.);
- The necessary information during the audit process will be well documented for both the audit team and the code owner in a timely manner.

2 Summary

This report has been commissioned by Glow Swap to identify any potential issues and vulnerabilities in the source code of the Glow Swap smart contract, as well as any contract dependencies that were not part of an officially recognized library. In this audit, we have utilized various techniques, including manual code review and static analysis, to identify potential vulnerabilities and security issues.

During the audit, we identified 3 issues of varying severity, listed below.

ID	Title	Severity	Status
GV3–1	Lack of Events Emit	Minor	Fixed
IMC–1	Lack `indexed` In Event	Informational	Fixed
VAU–1	Vault `emergencyWithdraw` Design Issue	Major	Fixed

3 Participant Process

Here are the relevant actors with their respective abilities within the Glow Swap Smart Contract:

Admin

- The Admin can set the statue of the pool through `setEmergency`.
- The Admin can set the receiver address through `setReceiver`.
- The Admin can set the LMPoolDeployer address through `setLMPoolDeployer`.
- The Admin can set the operatorAddress address through `setOperator`.
- The Admin can add a new farm pool through `add`.
- The Admin can update the pool's `REWARD` allocation point and `PeriodDuration`, `FarmBooster` address and through `set`, `setPeriodDuration` and `updateFarmBoostContract`.

User

- The User can open a position and get `lp` Token through `mint`.
- The User can increase the liquidity of his position through `increaseLiquidity`.
- The User can decrease the liquidity of his position through `decreaseLiquidity`.
- The User can collect the reward of his position through `collect`.
- The User can deposit their farm pool LP NFT to for staking and get reward through transfer NFT to `MasterChef`.
- The User can withdraw LP tokens from pool through `withdraw`.
- The User can update the liquidity of their NFT position through `updateLiquidity` / `increaseLiquidity` / `decreaseLiquidity`.
- If the User wants to quit the staking he can burn his LP NFT through `burn`.

4 Findings

GV3-1 Lack of Events Emit

Severity: Minor

Status: Fixed

Code Location:

projects/v3-lm-pool/contracts/GlowV3LmPool.sol#66

Descriptions:

The smart contract lacks appropriate events for monitoring sensitive operations, which could make it difficult to track sensitive actions or detect potential issues.

Suggestion:

It is recommended to emit events for those sensitive functions.

IMC-1 Lack `indexed` In Event

Severity: Informational

Status: Fixed

Code Location:

projects/masterchef-v3/contracts/interfaces/IMasterChefV3.sol#153-155

Descriptions:

Index event fields make the field more quickly accessible to off-chain tools that parse events. However, note that each index field costs extra gas during emission, so it's not necessarily best to index the maximum allowed per event (three fields). Each event should use three indexed fields if there are three or more fields and gas usage is not particularly of concern for the events in question. If there are fewer than three fields, all of the fields should be indexed.

Suggestion:

It is recommended to add `indexed` modifier in the event.

Resolution:

The client followed the suggestion and fixed this issue.

VAU-1 Vault `emergencyWithdraw` Design Issue

Severity: Major

Status: Fixed

Code Location:

projects/masterchef-v3/contracts/Vault.sol#28-42

Descriptions:

In the `emergencyWithdraw` function if the `_token` address is equal to `WETH`, the `amount` is calculated by `address(this).balance`. However, `WETH` is still an ERC20 Token, which doesn't get the correct amount of `WETH` owned by the contract in this way, resulting in `WETH` not being withdrawn correctly.

Suggestion:

It is recommended to fix this issue and changes to correct the transfer logic.

Resolution:

The client followed the suggestion and fixed this issue.

Appendix 1

Issue Level

- **Informational** issues are often recommendations to improve the style of the code or to optimize code that does not affect the overall functionality.
- **Minor** issues are general suggestions relevant to best practices and readability. They don't post any direct risk. Developers are encouraged to fix them.
- **Medium** issues are non-exploitable problems and not security vulnerabilities. They should be fixed unless there is a specific reason not to.
- **Major** issues are security vulnerabilities. They put a portion of users' sensitive information at risk, and often are not directly exploitable. All major issues should be fixed.
- **Critical** issues are directly exploitable security vulnerabilities. They put users' sensitive information at risk. All critical issues should be fixed.

Issue Status

- **Fixed:** The issue has been resolved.
- **Partially Fixed:** The issue has been partially resolved.
- **Acknowledged:** The issue has been acknowledged by the code owner, and the code owner confirms it's as designed, and decides to keep it.

Appendix 2

Disclaimer

This report is based on the scope of materials and documents provided, with a limited review at the time provided. Results may not be complete and do not include all vulnerabilities. The review and this report are provided on an as-is, where-is, and as-available basis. You agree that your access and/or use, including but not limited to any associated services, products, protocols, platforms, content, and materials, will be at your own risk. A report does not imply an endorsement of any particular project or team, nor does it guarantee its security. These reports should not be relied upon in any way by any third party, including for the purpose of making any decision to buy or sell products, services, or any other assets. TO THE FULLEST EXTENT PERMITTED BY LAW, WE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, IN CONNECTION WITH THIS REPORT, ITS CONTENT, RELATED SERVICES AND PRODUCTS, AND YOUR USE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NOT INFRINGEMENT.