verichains

SECURITY AUDIT OF

LEISUREMETA DIAMOND

Public Report

Jul 09, 2024

Verichains Lab

info@verichains.io

https://www.verichains.io

Driving Technology > Forward

ABBREVIATIONS

Name	Description
Ethereum	An open source platform based on blockchain technology to create and distribute smart contracts and decentralized applications.
Ether (ETH)	A cryptocurrency whose blockchain is generated by the Ethereum platform. Ether is used for payment of transactions and computing services in the Ethereum network.
Smart contract	A computer protocol intended to digitally facilitate, verify or enforce the negotiation or performance of a contract.
Solidity	A contract-oriented, high-level language for implementing smart contracts for the Ethereum platform.
Solc	A compiler for Solidity.
ERC20	ERC20 (BEP20 in Binance Smart Chain or *x*RP20 in other chains) tokens are blockchain-based assets that have value and can be sent and received. The primary difference with the primary coin is that instead of running on their own blockchain, ERC20 tokens are issued on a network that supports smart contracts such as Ethereum or Binance Smart Chain.

EXECUTIVE SUMMARY

This Security Audit Report was prepared by Verichains Lab on Jul 09, 2024. We would like to thank the LeisureMeta for trusting Verichains Lab in auditing smart contracts. Delivering high-quality audits is always our top priority.

This audit focused on identifying security flaws in code and the design of the LeisureMeta Diamond. The scope of the audit is limited to the source code files provided to Verichains. Verichains Lab completed the assessment using manual, static, and dynamic analysis techniques.

During the audit process, the audit team had identified no vulnerable issues in the smart contracts code.

TABLE OF CONTENTS

1. MANAGEMENT SUMMARY

1.1. About LeisureMeta Diamond

The LeisureMeta Token (LM) is a ERC20 token used in the Leisure Metaverse ecosystem. The LM token is used as a form of payment for fees incurred from user activities within the Leisure Metaverse and all fees are burned, helping to prevent inflation and maintain a healthy ecosystem. The token also serves as a bridge between the virtual and real economies. Users are rewarded with LM tokens for participating in activities and collecting NFTs, with the rewards being based on the total user activity score.

1.2. Audit scope

This audit focused on identifying security flaws in code and the design of the LeisureMeta Diamond. It was conducted on the source code provided by the LM LLC team.

The latest version of the following files was made available in the course of the review:

SHA1 Sum	File
ae340cd944f350e2c69dcee2a7f3b168cd1fff59	./facets/DiamondCutFacet.sol
d8504de948dc73aaeaaaa600b498cfdcfd1b3697	./facets/OwnershipFacet.sol
7016a84d9f750e8c4b19efaf0bbc9165f18422b1	./facets/TransferFacet.sol
c3c36b389b98227792a3996b07964f0e62d49a52	./facets/ControlFacet.sol
7cb94b16efa4a9533335107ffd89abf4020a262e	./facets/DiamondLoupeFacet.sol
4084215347fde3c46669d6b8f71cd373b7f21e1f	./upgradeInitializers/DiamondInit.sol
afb877aefb639aec79dadbace95d988f719f79b1	./LeisureMeta.sol
9ca6787c638fe437e1612de7b4053df6802a6960	./libraries/LibAppStore.sol
1eb69356ac09f9c741b96bc06b0dcef4c3870d36	./libraries/LibDiamond.sol
250605d245aa829f802ba9ff9025af720802ac6b	./Diamond.sol
336e67d665bdd3b88604403dc1f3a673aa8e70fd	./interfaces/IDiamondCut.sol
4679c00a0922108b6d356bf0d99ffe88e37aaa77	./interfaces/IDiamondLoupe.sol
eff747b414b6cb1ab4ba92dd382782b10cb10d33	./interfaces/IERC165.sol
a81008415a43b5bd1cf70911d8ef13850c3bec23	./interfaces/IERC173.sol

1.3. Audit methodology

Our security audit process for smart contract includes two steps:

- Smart contract codes are scanned/tested for commonly known and more specific vulnerabilities using public and RK87, our in-house smart contract security analysis tool.
- Manual audit of the codes for security issues. The contracts are manually analyzed to look for any potential problems.

Following is the list of commonly known vulnerabilities that were considered during the audit of the smart contract:

- Integer Overflow and Underflow
- Timestamp Dependence
- Race Conditions
- Transaction-Ordering Dependence
- DoS with (Unexpected) revert
- DoS with Block Gas Limit
- Gas Usage, Gas Limit and Loops
- Redundant fallback function
- Unsafe type Inference
- Reentrancy
- Explicit visibility of functions state variables (external, internal, private and public)
- Logic Flaws

For vulnerabilities, we categorize the findings into categories as listed in table below, depending on their severity level:

SEVERITY LEVEL	DESCRIPTION
CRITICAL	A vulnerability that can disrupt the contract functioning; creates a critical risk to the contract; required to be fixed immediately.
HIGH	A vulnerability that could affect the desired outcome of executing the contract with high impact; needs to be fixed with high priority.
MEDIUM	A vulnerability that could affect the desired outcome of executing the contract with medium impact in a specific scenario; needs to be fixed.
LOW	An issue that does not have a significant impact, can be considered as less important.

Table 1. Severity levels

1.4. Disclaimer

LeisureMeta acknowledges that the security services provided by Verichains, are conducted to the best of their professional abilities but cannot guarantee 100% coverage of all security vulnerabilities. LeisureMeta understands and accepts that despite rigorous auditing, certain vulnerabilities may remain undetected. Therefore, LeisureMeta agrees that Verichains shall not be held responsible or liable, and shall not be charged for any hacking incidents that occur due to security vulnerabilities not identified during the audit process.

1.5. Acceptance Minute

This final report served by Verichains to the LeisureMeta will be considered an Acceptance Minute. Within 7 days, if no any further responses or reports is received from the LeisureMeta, the final report will be considered fully accepted by the LeisureMeta without the signature.

2. AUDIT RESULT

2.1. Overview

The LeisureMeta Diamond was written in `Solidity` language, with the required versions being `^0.8.19`. The source code was written based on OpenZeppelin's library.

2.2. LeisureMeta.sol

Table 2 lists some properties of the audited LeisureMeta Diamond (as of the report writing time).

PROPERTY	VALUE
Name	LeisureMeta
Symbol	LM
Decimals	18
Total Supply	5,000,000,000 x10^{18} Note: the number of decimals is 18, so the total representation token will be 5,000,000,000 or 5 billion.

Table 2. The LeisureMeta Diamond properties

This contract extends `ERC20Burnable`, `Ownable` and `Pausable` contracts. With `Ownable`, by default, the token Owner is the contract deployer but he can transfer ownership to another address at any time. `ERC20Burnable` allows token holders to destroy both their own tokens and those that they have an allowance for.

The contract has an owner, and the transfer token is pausable by the owner. The token is also burnable, meaning the owner or any other user can burn (permanently destroy) their tokens.

With this contract, the owner can lock and revoke your tokens for a certain period of time.

There are 4 types of locks, and for each lock type, the token owner will transfer tokens to the user's wallet address and lock them for multiple periods, with each period being 30 days (1 month):

- `daoLock`: there are 60 stages, and the tokens will be unlocked after 1 month from the `_dDay`.

- `saleLock`: an amount is divided by 7 and locked for 10 months, starting unlocking 4 months after `_dDay`.
- `generalLock`: an amount is divided by 20 and locked for 25 months, starting unlocking from 6 months after `_dDay`.
- `customLock`: the token owner can freely choose the number of stages and decide when to start them.

The value of the variable `_dDay` is set by the token owner.

The token owner has the ability to revoke any locked tokens from any address locked by generalLock and customLock using the `revoke()` function.

2.3. Diamond proxy implementation

The LeisureMeta Diamond is a upgradeable smart contract that implements the Diamond proxy pattern. The Diamond is a pattern for building complex smart contracts that can be upgraded. The Diamond is based on the EIP-2535 standard.

There are many facets:

- `Diamond Cut Facet`: A diamond contains within it a mapping of function selectors to facet addresses. This facet that are add/replace/remove any number of functions and optionally execute a function with `delegatecall` are protected by contract owner checking.
- `Diamond Loupe Facet`: Provide information about and visualize diamonds.
- `Ownership Facet`: Manage ownership of the diamond, such as transfer ownership to another address, renounce ownership. All functions at this facet are protected by contract owner checking.
- `Control Facet`: Setup config by the owner, such as `setTokenStorage`, `storeToken`, `setGateway`, `setBoundary`, `setDeployedContract`, `addApprovers`, `removeApprovers`. All functions at this facet are protected by contract owner checking.*Notes: owner can transfer tokens from this contract to `tokenStorage` address.*
- `Transfer Facet`: Transaction management, such as `addTransaction`, `confirmTransaction`, `revokeTransaction`, and `executeTransaction`. All functions at this facet are protected by gateway and approver checking.

Following security considerations in the EIP-2535 standard, we reviewed logic of contracts and following security issues:

SECURITY ISSUE	PASSED/FAILED
Ownership and Authentication	Passed. All operation functions are protected by operator user checking.

SECURITY ISSUE	PASSED/FAILED
Arbitrary Execution with `diamondCut`	Passed. Because the `DiamondCut` function is protected by contract owner checking
Do Not Self Destruct	Passed
Avoid Function Selector Clash	Passed
Transparency	Passed

Table 3. Security considerations in the EIP-2535 standard

2.4. Findings

During the audit process, the audit team found no vulnerability in the given version of LeisureMeta Diamond but we have a best practice recommendation for the LeisureMeta Diamond team.

#	Issue	Severity	Status
1	Missing emit event in some functions	INFORMATIVE	

2.4.1. Missing emit event in some functions - INFORMATIVE

Affected files:

- contracts/facets/ControlFacet.sol

Emit event allows people and software to monitor changes to a contract. If any bad acting function is added to a diamond then it can be seen.

Functions have not emitted events: `addApprovers`, `removeApprovers`, `setBoundary`, `setDeployedContract`.

RECOMMENDATION

Add emit event in these functions.

UPDATES

- *Aug 14, 2023*: This issue has been acknowledged and fixed by the LeisureMeta team.

APPENDIX

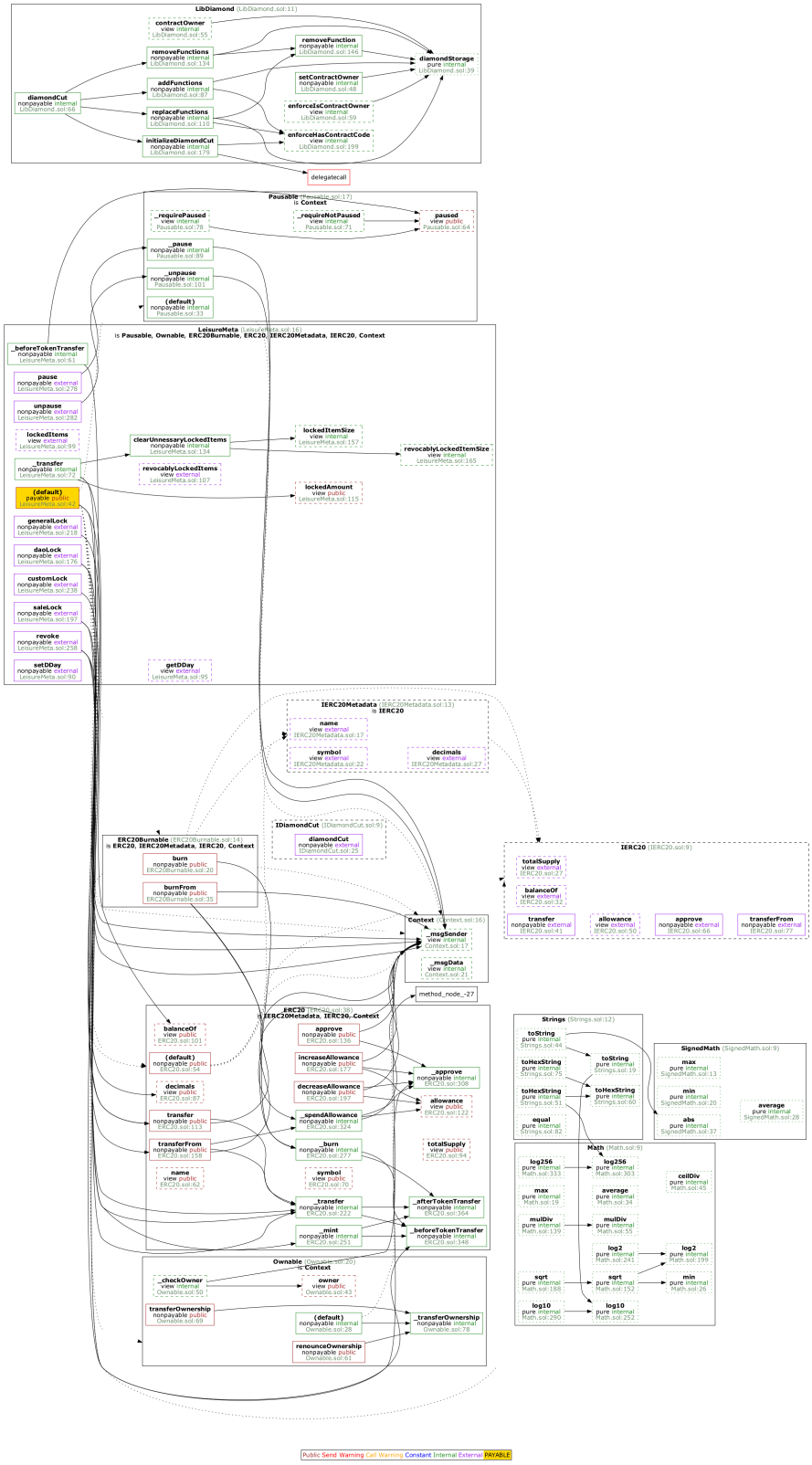
LibDiamond (LibDiamond.sol:11)
contractOwner
removeFunctions
addFunctions
replaceFunctions
initializeDiamondCut
removeFunction
setContractOwner
enforceIsContractOwner
enforceHasContractCode
diamondStorage
diamondCut
delegatecall
Pausable (Pausable.sol:17)
is Context
_requirePaused
_requireNotPaused
paused
_pause
_unpause
(default)
LeisureMeta (LeisureMeta.sol:16)
is Pausable, Ownable, ERC20Burnable, ERC20, IERC20Metadata, IERC20, Context
_beforeTokenTransfer
pause
unpause
lockedItems
_transfer
(default)
generalLock
daoLock
customLock
saleLock
revoke
setDDay
clearUnnessaryLockedItems
revocablyLockedItems
lockedItemSize
revocablyLockedItemSize
lockedAmount
getDDay
IERC20Metadata (IERC20Metadata.sol:13)
is IERC20
name
symbol
decimals
IDiamondCut (IDiamondCut.sol:9)
diamondCut
ERC20Burnable (ERC20Burnable.sol:14)
is ERC20, IERC20Metadata, IERC20, Context
burn
burnFrom
IERC20 (IERC20.sol:9)
totalSupply
balanceOf
transfer
allowance
approve
transferFrom
Context (Context.sol:16)
_msgSender
_msgData
method_node_-27
ERC20 (ERC20.sol:18)
is IERC20Metadata, IERC20, Context
balanceOf
(default)
decimals
transfer
transferFrom
name
approve
increaseAllowance
decreaseAllowance
spendAllowance
_burn
_transfer
_mint
_approve
allowance
totalSupply
symbol
_afterTokenTransfer
_beforeTokenTransfer
Ownable (Ownable.sol:20)
is Context
_checkOwner
owner
transferOwnership
renounceOwnership
_transferOwnership
Strings (Strings.sol:12)
toString
toHexString
equal
SignedMath (SignedMath.sol:9)
max
min
average
abs
Math (Math.sol:9)
log256
ceilDiv
mulDiv
log2
sqrt
log10
Public Send Warning Call Warning Constant Internal External PAYABLE

Image 1. LeisureMeta Diamond's LeisureMeta call graph

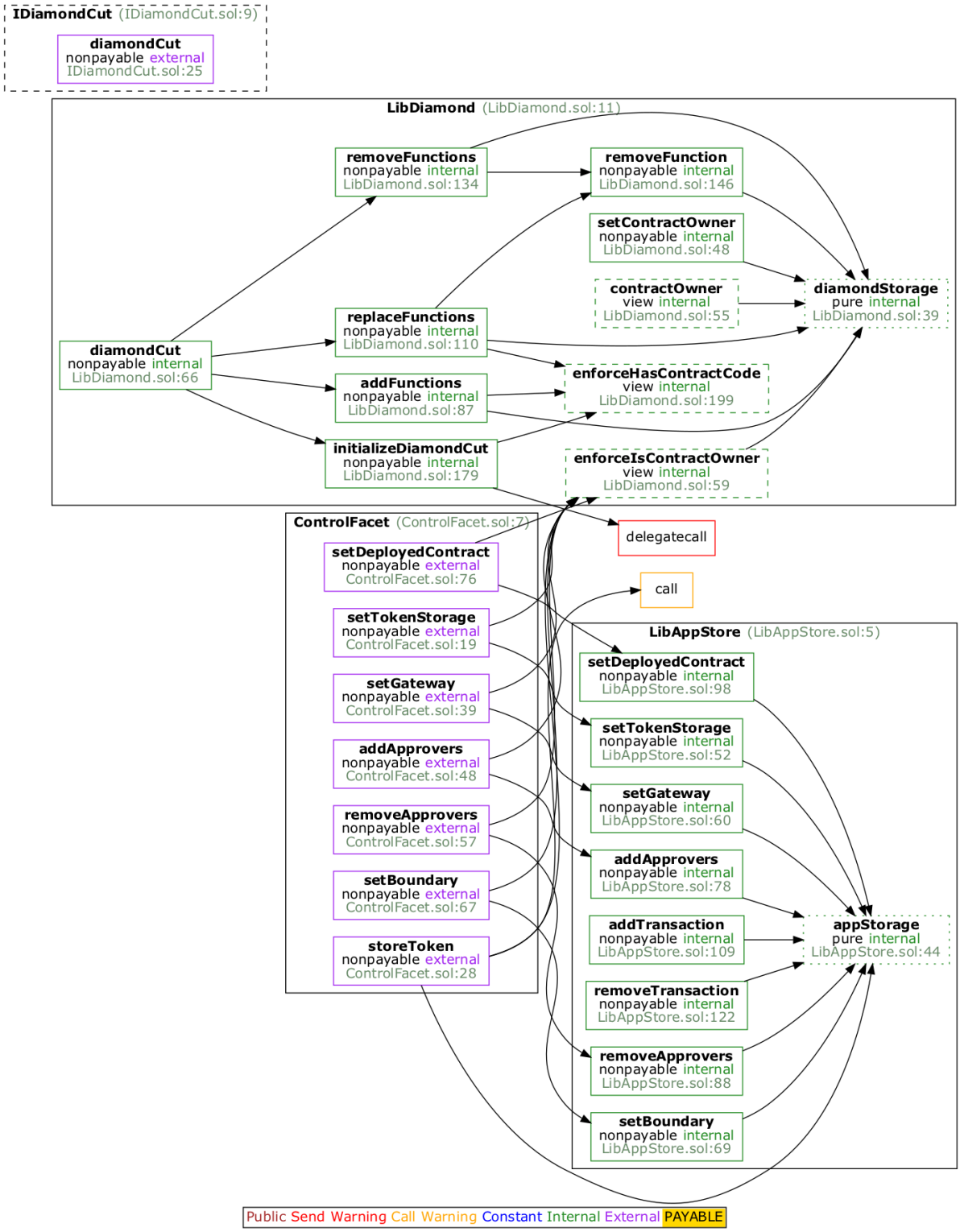

Image 2. LeisureMeta Diamond's ControlFacet call graph

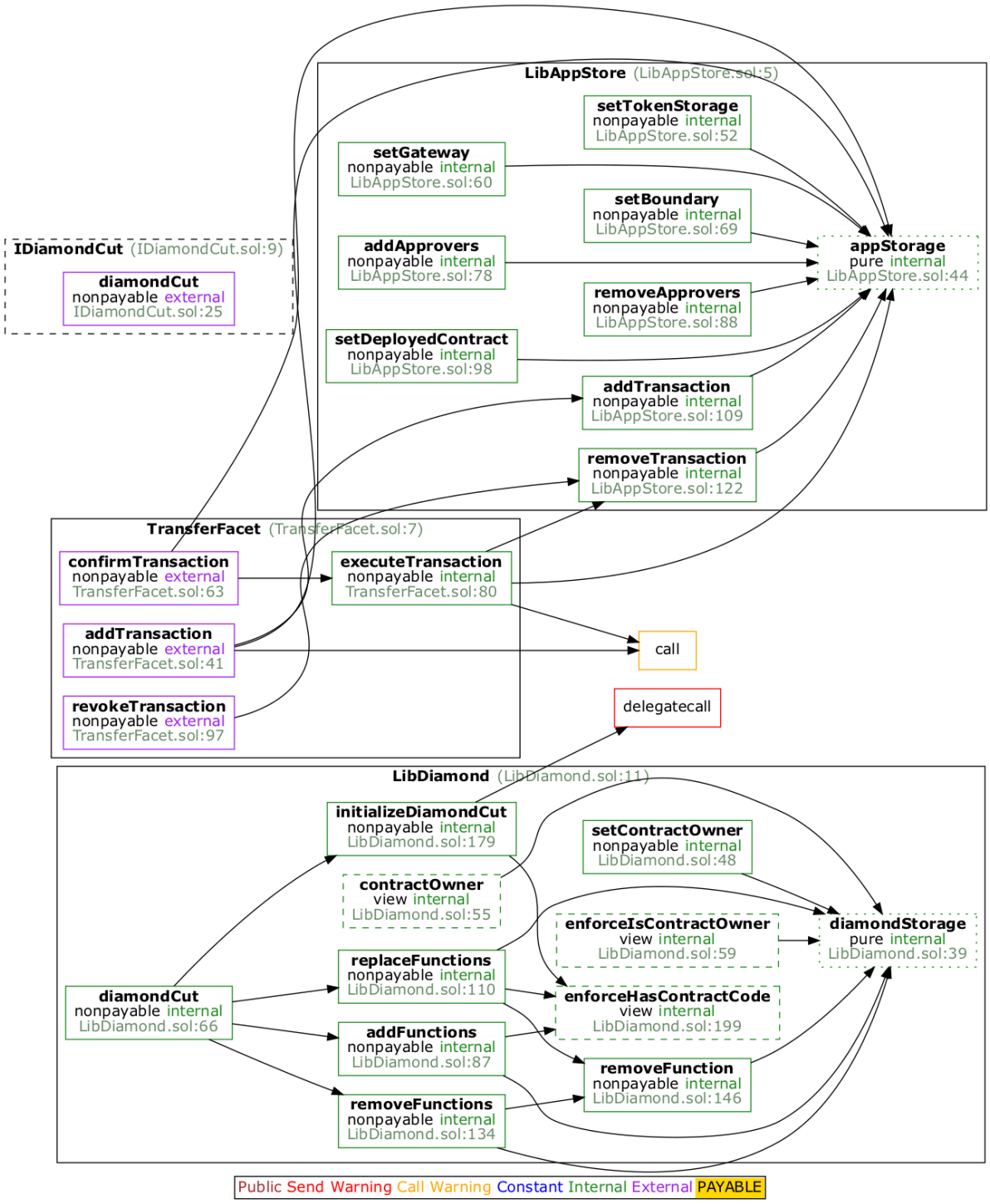

Image 3. LeisureMeta Diamond's TransferFacet call graph

3. VERSION HISTORY

Version	Date	Status/Change	Created by
1.0	*Jul 28, 2023*	Public Report	Verichains Lab
1.1	*Aug 14, 2023*	Public Report	Verichains Lab
1.2	*Jul 09, 2024*	Public Report	Verichains Lab

Table 4. Report versions history