InterFi NETWORK

SMART CONTRACT AUDIT

interfinetwork

hello@interfi.network

https://interfi.network

PREPARED FOR

RWA COIN - PRESALE CONTRACT

INTRODUCTION

Auditing Firm	InterFi Network
Client Firm	RWA Coin
Methodology	Automated Analysis, Manual Code Review
Language	Solidity
Contract	0x836ff3751C60a6Ac8a603d2DFBc8c93a731EB8e5
Blockchain	Ethereum Chain
Centralization	Active Ownership
Commit	3491e09570523586a032c4630138294ef7a7e29f
Website	https://rwacoin.io/
Telegram	https://t.me/+1_P7pKzTu7g1YWUx
X (Twitter)	https://x.com/RWA2coin
Report Date	November 02, 2024

INTERFI CONFIDENTIAL INTERFI AUDIT REPORT INTERFI CONFIDENTIAL INTERFI AUDIT REPORT INTERFI CONFIDENTIAL INTERFI AUDIT REPORT INTERFI CONFIDENTIAL INTERFI AUDIT REPORT INTERFI CONFIDENTIAL

ℹ Verify the authenticity of this report on our website: https://www.github.com/interfinetwork

EXECUTIVE SUMMARY

InterFi has performed the automated and manual analysis of solidity codes. Solidity codes were reviewed for common contract vulnerabilities and centralized exploits. Here's a quick audit summary:

Status	Critical 🔴	Major 🟠	Medium 🟡	Minor 🟢	Unknown 🟤
Open	0	0	1	4	0
Acknowledged	1	0	1	1	2
Resolved	1	1	1	4	0
Important Functions	`BuyWithETH, BuyWithUSDT, BuyWithUSDC, claim`				
Noteworthy Privileges	**`setBlacklist`**`, setaggregatorv3, settoken, setPresalePricePerUsdt, setmaxTokeninPresale, recoverERC20, setUSDT, setUSDC, releaseFunds,`				

INTERFI CONFIDENTIAL AUDIT REPORT

ℹ RWA Coin is raising funds directly through its own presale platform, without the use of external or third-party services. There is no public evidence of KYC verification through any external or recognized KYC services. As of this token audit, any associated decentralized application (dApp) has not been subjected to an external audit by a recognized third-party security firm. Given the lack of external KYC verification and absence of third-party audits of dApp, potential users are advised to exercise caution and perform thorough due diligence before participating in project or presale.

ℹ Please note that smart contracts deployed on blockchains aren't resistant to exploits, vulnerabilities and/or hacks. Blockchain and cryptography assets utilize new and emerging technologies. These technologies present a high level of ongoing risks. For a detailed understanding of risk severity, source code vulnerability, and audit limitations, kindly review the audit report thoroughly.

TABLE OF CONTENTS

TABLE OF CONTENTS 3

SCOPE OF WORK.......... 5

AUDIT METHODOLOGY 6

RISK CATEGORIES 8

CENTRALIZED PRIVILEGES 9

AUTOMATED ANALYSIS.......... 10

INHERITANCE GRAPH 14

MANUAL REVIEW.......... 15

DISCLAIMERS 34

ABOUT INTERFI NETWORK..........37

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

SCOPE OF WORK

InterFi was consulted by RWA Coin to conduct the smart contract audit of their solidity source codes. The audit scope of work is strictly limited to mentioned solidity file(s) only:

○ RwaPresale.sol

ℹ If source codes are not deployed on the main net, they can be modified or altered before main-net deployment. Verify the contract's deployment status below:

Public Contract Link	
https://etherscan.io/address/0x836ff3751c60a6ac8a603d2dfbc8c93a731eb8e5#code	
Contract Name	RwaPresale
Compiler Version	0.8.9
License	MIT

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL

AUDIT METHODOLOGY

Smart contract audits are conducted using a set of standards and procedures. Mutual collaboration is essential to performing an effective smart contract audit. Here's a brief overview of InterFi's auditing process and methodology:

CONNECT

- The onboarding team gathers source codes, and specifications to make sure we understand the size, and scope of the smart contract audit.

AUDIT

- Automated analysis is performed to identify common contract vulnerabilities. We may use the following third-party frameworks and dependencies to perform the automated analysis:
    - Remix IDE Developer Tool
    - Open Zeppelin Code Analyzer
    - SWC Vulnerabilities Registry
    - DEX Dependencies, e.g., Pancakeswap, Uniswap
- Simulations are performed to identify centralized exploits causing contract and/or trade locks.
- A manual line-by-line analysis is performed to identify contract issues and centralized privileges. We may inspect below mentioned common contract vulnerabilities, and centralized exploits:

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL

Centralized Exploits	○ Token Supply Manipulation ○ Access Control and Authorization ○ Assets Manipulation ○ Ownership Control ○ Liquidity Access ○ Stop and Pause Trading ○ Ownable Library Verification

Common Contract Vulnerabilities	○ Integer Overflow ○ Lack of Arbitrary limits ○ Incorrect Inheritance Order ○ Typographical Errors ○ Requirement Violation ○ Gas Optimization ○ Coding Style Violations ○ Re-entrancy ○ Third-Party Dependencies ○ Potential Sandwich Attacks ○ Irrelevant Codes ○ Divide before multiply ○ Conformance to Solidity Naming Guides ○ Compiler Specific Warnings ○ Language Specific Warnings

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL

REPORT

- The auditing team provides a preliminary report specifying all the checks which have been performed and the findings thereof.
- The client’s development team reviews the report and makes amendments to solidity codes.
- The auditing team provides the final comprehensive report with open and unresolved issues.

PUBLISH

- The client may use the audit report internally or disclose it publicly.

ℹ It is important to note that there is no pass or fail in the audit, it is recommended to view the audit as an unbiased assessment of the safety of solidity codes.

RISK CATEGORIES

A successful external attack may allow the external attacker to directly exploit. A successful centralization-related exploit may allow the privileged role to directly exploit. All risks which are identified in the audit report are categorized:

Risk Type	Definition
Critical 🔴	These risks pose immediate and severe threats, such as asset theft, data manipulation, or complete loss of contract functionality. They are often easy to exploit and can lead to significant, irreparable damage. Immediate fix is required.
Major 🟠	These risks can significantly impact code performance and security, and they may indirectly lead to asset theft and data loss. They can allow unauthorized access or manipulation of sensitive functions if exploited. Fixing these risks are important.
Medium 🟡	These risks may create attack vectors under certain conditions. They may enable minor unauthorized actions or lead to inefficiencies that can be exploited indirectly to escalate privileges or impact functionality over time.
Minor 🟢	These risks may include inefficiencies, lack of optimizations, code-style violations. These should be addressed to enhance overall code quality and maintainability.
Unknown 🟤	These risks pose uncertain severity to the contract or those who interact with it. Immediate fix is required to mitigate risk uncertainty.

INTERFI CONFIDENTIAL AUDIT REPORT

All statuses which are identified in the audit report are categorized here:

Status Type	Definition
Open	Risks are open.
Acknowledged	Risks are acknowledged, but not fixed.
Resolved	Risks are acknowledged and fixed.

CENTRALIZED PRIVILEGES

Centralization risk is the most common cause of cryptography asset loss. When a smart contract has a privileged role, the risk related to centralization is elevated.

There are some well-intended reasons have privileged roles, such as:

- Privileged roles can be granted the power to `pause()` the contract in case of an external attack.
- Privileged roles can use functions like, `include()`, and `exclude()` to add or remove wallets from fees, swap checks, and transaction limits. This is useful to run a presale and to list on an exchange.

Authorizing privileged roles to externally-owned-account (EOA) is dangerous. Lately, centralization-related losses are increasing in frequency and magnitude.

- The client can lower centralization-related risks by implementing below mentioned practices:
- Privileged role's private key must be carefully secured to avoid any potential hack.
- Privileged role should be shared by multi-signature (multi-sig) wallets.
- Authorized privilege can be locked in a contract, user voting, or community DAO can be introduced to unlock the privilege.
- Renouncing the contract ownership, and privileged roles.
- Remove functions with elevated centralization risk.

INTERFI CONFIDENTIAL AUDIT REPORT

ℹ Understand the project's initial asset distribution. Assets in the liquidity pair should be locked. Assets outside the liquidity pair should be locked with a release schedule.

AUTOMATED ANALYSIS

Symbol	Definition
🛑	Function modifies state
💵	Function is payable
🔒	Function is internal
🔐	Function is private
❗	Function is important

TERFI
NFIDENTIAL

INTER
CONFIDEN

```
| **Address** | Library |  |||
| └ | isContract | Internal 🔒 |   | |
| └ | sendValue | Internal 🔒 | 🛑  | |
| └ | functionCall | Internal 🔒 | 🛑  | |
| └ | functionCall | Internal 🔒 | 🛑  | |
| └ | functionCallWithValue | Internal 🔒 | 🛑  | |
| └ | functionCallWithValue | Internal 🔒 | 🛑  | |
| └ | functionStaticCall | Internal 🔒 |   | |
| └ | functionStaticCall | Internal 🔒 |   | |
| └ | functionDelegateCall | Internal 🔒 | 🛑  | |
| └ | functionDelegateCall | Internal 🔒 | 🛑  | |
| └ | _verifyCallResult | Private 🔐 |   | |
||||||
| **SafeERC20** | Library |  |||
| └ | safeTransfer | Internal 🔒 | 🛑  | |
| └ | safeTransferFrom | Internal 🔒 | 🛑  | |
| └ | safeApprove | Internal 🔒 | 🛑  | |
| └ | safeIncreaseAllowance | Internal 🔒 | 🛑  | |
| └ | safeDecreaseAllowance | Internal 🔒 | 🛑  | |
| └ | _callOptionalReturn | Private 🔐 | 🛑  | |
||||||
| **Context** | Implementation |  |||
| └ | _msgSender | Internal 🔒 |   | |
| └ | _msgData | Internal 🔒 |   | |
||||||
| **IERC20** | Interface |  |||
| └ | totalSupply | External ❗️ |   |NO❗️ |
| └ | balanceOf | External ❗️ |   |NO❗️ |
| └ | transfer | External ❗️ | 🛑  |NO❗️ |
| └ | allowance | External ❗️ |   |NO❗️ |
| └ | approve | External ❗️ | 🛑  |NO❗️ |
| └ | transferFrom | External ❗️ | 🛑  |NO❗️ |
||||||
| **SafeMath** | Library |  |||
| └ | add | Internal 🔒 |   | |
| └ | sub | Internal 🔒 |   | |
| └ | sub | Internal 🔒 |   | |
| └ | mul | Internal 🔒 |   | |
| └ | div | Internal 🔒 |   | |
| └ | div | Internal 🔒 |   | |
| └ | mod | Internal 🔒 |   | |
| └ | mod | Internal 🔒 |   | |
||||||
| **Ownable** | Implementation | Context |||
| └ | <Constructor> | Public ❗️ | 🛑  |NO❗️ |
| └ | owner | Public ❗️ |   |NO❗️ |
```

INTERFI CONFIDENTIAL

```
| └ | renounceOwnership | Public ❗ | 🛑  | onlyOwner |

| └ | transferOwnership | Public ❗ | 🛑  | onlyOwner |

||||||

| **ReentrancyGuard** | Implementation |  |||

| └ | <Constructor> | Public ❗ | 🛑  |NO❗ |

| └ | _nonReentrantBefore | Private 🔐 | 🛑  | |

| └ | _nonReentrantAfter | Private 🔐 | 🛑  | |

| └ | _reentrancyGuardEntered | Internal 🔒 |   | |

||||||

| **AggregatorV3Interface** | Interface |  |||

| └ | decimals | External ❗ |   |NO❗ |

| └ | description | External ❗ |   |NO❗ |

| └ | version | External ❗ |   |NO❗ |

| └ | getRoundData | External ❗ |   |NO❗ |

| └ | latestRoundData | External ❗ |   |NO❗ |

||||||

| **RwaPresale** | Implementation | Ownable, ReentrancyGuard |||

| └ | <Constructor> | Public ❗ | 🛑  |NO❗ |

| └ | <Receive Ether> | External ❗ |  💵 |NO❗ |

| └ | getLatestPriceETH | Public ❗ |   |NO❗ |

| └ | setaggregatorv3 | External ❗ | 🛑  | onlyOwner |

| └ | BuyWithETH | External ❗ |  💵 | nonReentrant |

| └ | BuyWithUSDT | External ❗ | 🛑  | nonReentrant |

| └ | BuyWithUSDC | External ❗ | 🛑  | nonReentrant |

| └ | claim | External ❗ | 🛑  | nonReentrant |

| └ | getValuePerUsdt | Public ❗ |   |NO❗ |

| └ | setPresalePricePerUsdt | External ❗ | 🛑  | onlyOwner |

| └ | stopPresale | External ❗ | 🛑  | onlyOwner |
```

TERFI
NFIDENTIAL

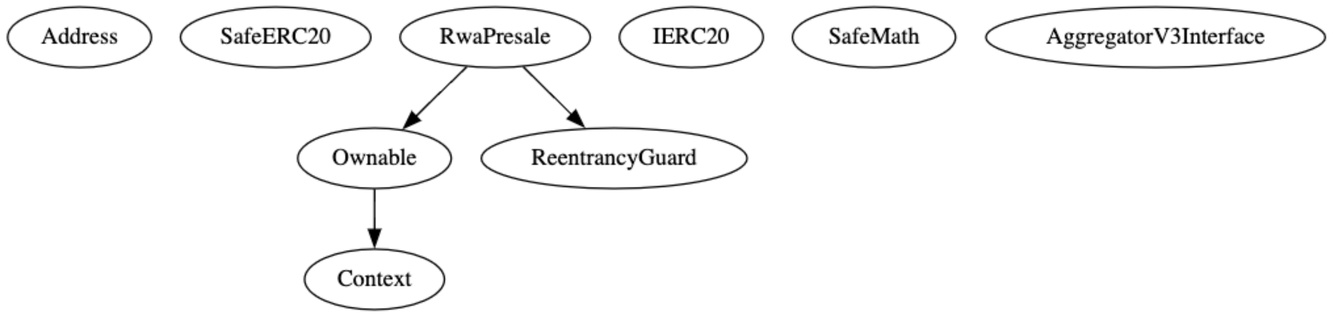
INTER
CONFIDEN

```
| └ | resumePresale | External ❗ | 🛑 | onlyOwner |
| └ | setmaxTokeninPresale | External ❗ | 🛑 | onlyOwner |
| └ | recoverERC20 | External ❗ | 🛑 | onlyOwner |
| └ | settoken | External ❗ | 🛑 | onlyOwner |
| └ | setUSDT | External ❗ | 🛑 | onlyOwner |
| └ | setUSDC | External ❗ | 🛑 | onlyOwner |
| └ | setBlacklist | External ❗ | 🛑 | onlyOwner |
| └ | releaseFunds | External ❗ | 🛑 | onlyOwner |
| └ | ETHToToken | Public ❗ |   |NO❗ |
| └ | changefeeReceiver | External ❗ | 🛑 | onlyOwner |
| └ | StartClaim | External ❗ | 🛑 | onlyOwner |
| └ | StopClaim | External ❗ | 🛑 | onlyOwner |
```

INTERFI
CONFIDENTIAL

INTERFI
CONFIDENTIAL

INHERITANCE GRAPH



INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL

MANUAL REVIEW

Identifier	Definition	Severity
CEN-01	Centralized privileges	Critical 🔴
CEN-01-01	Privileged role has authority to blacklist participants	
CEN-01-02	Privileged role can withdraw tokens and USDT from contract	
CEN-01-03	Privileged role can set fees to any value	

Important `onlyOwner` centralized privileges are listed below:

```
renounceOwnership
transferOwnership
setaggregatorv3
setPresalePricePerUsdt
stopPresale
resumePresale
setmaxTokeninPresale
recoverERC20
settoken
setUSDT
setUSDC
setBlacklist
releaseFunds
changefeeReceiver
StartClaim
StopClaim
```

TERFI
NFIDENTIAL

INTER
CONFIDEN

RECOMMENDATION

Securing private keys or access credentials of deployers, contract owners, operators, and other roles with privileged access is crucial to prevent single points of failure that can compromise contract security.

Use of multi-signature wallets is recommended – These wallets require multiple authorizations to execute sensitive contract functions, reducing the risk associated with single-party control.

Use of decentralized governance model is recommended – This model allows token holders and stakeholders to actively participate in decision-making, such as contract upgrades and parameter adjustments, enhancing overall security and resilience.

ACKNOWLEDGEMENT

RWA Coin team argued that centralized and controlled privileges are used as required.

TERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
NFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

Identifier	Definition	Severity
CEN-03	Lack of circuit breaker	Minor 🟢

Smart contract lacks a circuit breaker mechanism, which can be crucial in halting operations in case of a detected vulnerability, bug, or attack.

`Pausable` library from *OpenZeppelin* can be used to pause smart contract.

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Implement a circuit breaker mechanism that can be activated by authorized addresses.

Identifier	Definition	Severity
LOG-01	Insufficient input boundaries	Minor 🟢

Below mentioned functions are set without sufficient input boundaries:

```
setPresalePricePerUsdt
setmaxTokeninPresale
```

TERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
NFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Establish clear upper price boundaries. All operational parameters remain within safe and rational ranges.

Identifier	Definition	Severity
LOG-02	Potential front-running	Minor 🟢

Buy functions are vulnerable to front running attacks. A malicious actor can watch the transaction pool and execute a transaction with higher gas fees to precede the original transaction, possibly affecting the price or availability of tokens, especially when nearing the presale cap.

```
BuyWithETH
BuyWithUSDT
BuyWithUSDC
```

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL

RECOMMENDATION

Implement commit-reveal schemes or transaction ordering to protect against front-running. Use decentralized exchange mechanism to determine prices and execute buys, reducing the predictability of transactions.

Identifier	Definition	Severity
LOG-03	Re-entrancy	Critical 🔴

Below mentioned functions are used without Re-entrancy guard:

`BuyWithETH`: Transfers Ether before updating `Claimable[msg.sender]` and `TokenSold`.

`BuyWithUSDT` and `BuyWithUSDC:` Transfer tokens before updating `Claimable[msg.sender]` and `TokenSold`.

`claim:` Transfers tokens before updating `Claimable[msg.sender]` to zero.

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Use Checks-Effects-Interactions (CEI) pattern when transferring control to external entities. This design pattern ensures that all state changes are completed before external interactions occur. Additionally, implement re-entrancy guard to block recursive calls from external contracts.

RESOLUTION

RWA Coin team has added `nonReentrant` modifier to these functions. Function logic is updated to follow Checks-Effects-Interactions (CEI) pattern.

Identifier	Definition	Severity
LOG-05	Incorrect balance checks	Major 🟠

Contract methods do not verify the success of the token transfer from the user to the contract, relying solely on `safeTransferFrom` without additional checks to ensure that the correct `amount` of tokens was indeed transferred.

```
BuyWithUSDT: IERC20(USDT).safeTransferFrom(msg.sender, address(this), _amt);
BuyWithUSDC: IERC20(USDC).safeTransferFrom(msg.sender, address(this), _amt);
```

`claim` function does not verify whether there are enough tokens in the contract's balance to fulfill the claim.

```
claim: token.transfer(msg.sender, claimable);
```

TERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
NFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Confirm that the actual token balance of the contract increases by the expected amount before proceeding with token transfers

RESOLUTION

RWA Coin team has added appropriate checks to confirm that the actual token balance of the contract increases by the expected amount before proceeding with token transfers. However, there’s still a room for improvement as suggested in LOG-05-01.

Identifier	Definition	Severity
LOG-05-01	Logical issues with `claim` condition	Medium 🟡

Balance check in `claim` function uses the condition `claimable >= token.balanceOf(address(this)).` This line has a logical flaw and should instead check if `claimable <= token.balanceOf(address(this)).` The current condition will revert if claimable is less than the contract balance, which would prevent legitimate claims from proceeding.

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL

RECOMMENDATION

Modify balance check logic in `claim` function.

```
require(claimable <= token.balanceOf(address(this)), "Not sufficient tokens available");
```

Identifier	Definition	Severity
COD-01	Use of `tx.origin` for sender authentication	Medium 🟡

Smart contract uses `tx.origin` check that the caller is not a contract in buy functions. This is unsafe as it can be manipulated in complex call contexts, particularly in cases involving multiple contracts.

Usage of `tx.origin == msg.sender` in buy functions:

```
BuyWithETH
BuyWithUSDT
BuyWithUSDC
```

TERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
NFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Replace `tx.origin` with `msg.sender` for direct sender checks, and use adequate authentication mechanisms that are less susceptible to manipulation.

RESOLUTION

RWA Coin team has removed usage of `tx.origin`.

Identifier	Definition	Severity
COD-02	Potential information leakage through public functions	Minor 🟢

These utility functions are publicly accessible, which lead to information leakage that aids in other attacks:

```
getLatestPriceETH
getValuePerUsdt
contractbalance
ETHToToken
```

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL

RECOMMENDATION

Adjust the visibility of functions based on their usage. Functions that do not need to be accessed externally should be marked `internal` or `private` to reduce the attack surface.

ACKNOWLEDGEMENT

RWA Coin team argued that contract logic requires that these utility functions are publicly accessible.

Identifier	Definition	Severity
COD-03	Reliance on single price feed aggregator	Unknown ●

Smart contract relies on `AggregatorV3Interface` for ETH price, which introduces a single point of failure and potential price manipulation risk.

`getLatestPriceETH` uses `priceFeedETH.latestRoundData()`.

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Utilize multiple price feeds from different sources to determine the average or median price, reducing dependency on a single price feed.

ACKNOWLEDGEMENT

RWA Coin team argued that `AggregatorV3Interface` is widely used, and it is considered safe and accurate.

Identifier	Definition
COD-04	Note regarding flash loan abuse

Smart contract does not directly interact with flash loans, but its economic mechanisms - like token pricing and referral payments, can be susceptible to manipulation using flash loans. An attacker can use flash loans to manipulate balances or the outcome of conditional checks temporarily.

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL

Identifier	Definition	Severity
COD-05	Missing zero address validation	Minor ●

Below mentioned functions are missing zero address input validation:

```
setaggregatorv3
settoken
setUSDT
setUSDC
changefeeReceiver
```

TERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
NFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Validate if the modified address is `dead(0)` or not.

RESOLUTION

RWA Coin team has added zero address checks to mentioned functions.

Identifier	Definition	Severity
COD-06	Potential denial of service (DoS)	Medium 🟡

`claim` function transfers tokens to the caller based on their claimable amount but does not account for the scenario where the contract does not have enough tokens to fulfill all claims. This can lead to denial of service (DoS) if users are unable to claim their tokens if the contract balance is too low.

TERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
NFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Add checks before attempting to transfer tokens to ensure that the contract holds enough tokens to meet `claim` requests.

ACKNOWLEDGEMENT

RWA Coin team has added check to ensure that the contract holds enough tokens to process individual transfers. However, it does not solve the underlying risk of running out of tokens overall, which could prevent some users from claiming.

Identifier	Definition	Severity
COD-07	Unsafe typecasting	Minor 🟢

`ETHToToken` function performs arithmetic operations that could potentially result in precision loss due to integer division. This will lead to incorrect calculations of the number of tokens to be credited.

`numberOfTokens` is calculated as `uint256 numberOfTokens = (ETHToUSD * (TokenPricePerUsdt)) / (1e8);`

TERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
NFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Order of operations must preserve precision, or use a higher precision for intermediate calculations.

Identifier	Definition	Severity
COD-10	Direct and indirect dependencies	Unknown ●

Smart contract extensively interacts with third-party protocols and external libraries, including `ERC20` tokens (`USDT, USDC`), `AggregatorV3Interface` for price feeds, and *OpenZeppelin* `SafeERC20` library for token operations. While these components are assumed to be secure and reliable within the scope of this audit, they represent a significant dependency risk. Any vulnerabilities, bugs, or malicious changes in these external entities can directly impact the contract's functionality and security.

For instance, upgrades to these protocols or libraries could introduce incompatible changes, increase transaction fees, or deprecate features that the contract relies on.

Additionally, reliance on a single price feed - `AggregatorV3Interface`, for ETH price exposes the contract to risks of price manipulation or failures in the data source.

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Inspect third party dependencies regularly, and mitigate severe impacts whenever necessary.

ACKNOWLEDGEMENT

RWA Coin team will inspect third party dependencies regularly, and push upgrades whenever required.

Identifier	Definition	Severity
COD-12	Lack of event-driven architecture	Minor 🟢

Smart contract uses function calls to update state, which can make it difficult to track and analyze changes to the contract over time. Event omission reduces transparency and makes tracking changes through external applications or services more difficult.

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Implement event emissions for all state-changing actions within the contract. For example, emit an event after tokens are credited to a user's claimable balance or when tokens are sold.

RESOLUTION

RWA Coin team has added event emissions for most of the state-changing actions.

Identifier	Definition	Severity
VOL-01	Identical code	Minor 🟢

Identical code found in:

```
recoverERC20
EmergencyUSDT
```

TERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
NFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

RECOMMENDATION

Remove redundant and identical code.

RESOLUTION

RWA Coin team has removed identical code.

Identifier	Definition	Severity
COM-01	Floating pragma	Minor 🟢
COM-02	Multiple pragma directives	

Compiler is set to `^0.8.0`

Multiple pragmas are used in the smart contract.

INTERFI CONFIDENTIAL

RECOMMENDATION

Pragma should be fixed to stable compiler version. Fixing pragma ensures compatibility and prevents the contract from being compiled with incompatible compiler versions.

RESOLUTION

Smart contract is deployed with stable compiler.

DISCLAIMERS

InterFi Network provides the easy-to-understand audit of solidity source codes (commonly known as smart contracts).

The smart contract for this particular audit was analyzed for common contract vulnerabilities, and centralization exploits. This audit report makes no statements or warranties on the security of the code. This audit report does not provide any warranty or guarantee regarding the absolute bug-free nature of the smart contract analyzed, nor do they provide any indication of the client's business, business model or legal compliance. This audit report does not extend to the compiler layer, any other areas beyond the programming language, or other programming aspects that could present security risks. Cryptographic tokens are emergent technologies, they carry high levels of technical risks and uncertainty. You agree that your access and/or use, including but not limited to any services, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. This audit report could include false positives, false negatives, and other unpredictable results.

TERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
NFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

CONFIDENTIALITY

This report is subject to the terms and conditions (including without limitations, description of services, confidentiality, disclaimer and limitation of liability) outlined in the scope of the audit provided to the client. This report should not be transmitted, disclosed, referred to, or relied upon by any individual for any purpose without InterFi Network's prior written consent.

NO FINANCIAL ADVICE

This audit report does not indicate the endorsement of any particular project or team, nor guarantees its security. No third party should rely on the reports in any way, including to make any decisions to buy or sell a product, service or any other asset. The information provided in this report does not constitute investment advice, financial advice, trading advice, or any other sort of advice and you should not treat any of the report's content as such. This audit report should not be used in any way to make decisions around investment or involvement. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort.

FOR AVOIDANCE OF DOUBT, SERVICES, INCLUDING ANY ASSOCIATED AUDIT REPORTS OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL, REGULATORY, OR OTHER ADVICE.

TECHNICAL DISCLAIMER

ALL SERVICES, AUDIT REPORTS, SMART CONTRACT AUDITS, OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF ARE PROVIDED "AS IS" AND "AS AVAILABLE" AND WITH ALL FAULTS AND DEFECTS WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, INTERFI NETWORK HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO SERVICES, AUDIT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, INTERFI NETWORK SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM THE COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL

WITHOUT LIMITING THE FOREGOING, INTERFI NETWORK MAKES NO WARRANTY OF ANY KIND THAT ALL SERVICES, AUDIT REPORTS, SMART CONTRACT AUDITS, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET THE CLIENT'S OR ANY OTHER INDIVIDUAL'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULT, BE COMPATIBLE OR WORK WITH ANY SOFTWARE, SYSTEM, OR OTHER SERVICES, OR BE SECURE, ACCURATE, COMPLETE, FREE OF HARMFUL CODE, OR ERROR-FREE.

TIMELINESS OF CONTENT

The content contained in this audit report is subject to change without any prior notice. InterFi Network does not guarantee or warrant the accuracy, timeliness, or completeness of any report you access using the internet or other means, and assumes no obligation to update any information following the publication.

LINKS TO OTHER WEBSITES

This audit report provides, through hypertext or other computer links, access to websites and social accounts operated by individuals other than InterFi Network. Such hyperlinks are provided for your reference and convenience only and are the exclusive responsibility of such websites' and social accounts' owners. You agree that InterFi Network is not responsible for the content or operation of such websites and social accounts and that InterFi Network shall have no liability to you or any other person or entity for the use of third-party websites and social accounts. You are solely responsible for determining the extent to which you may use any content at any other websites and social accounts to which you link from the report.

INTERFI AUDIT REPORT INTERFI CONFIDENTIAL INTERFI AUDIT REPORT INTERFI CONFIDENTIAL INTERFI AUDIT REPORT INTERFI CONFIDENTIAL INTERFI AUDIT REPORT

ABOUT INTERFI NETWORK

InterFi Network provides intelligent blockchain solutions. We provide solidity development, testing, and auditing services. We have developed 150+ solidity codes, audited 1000+ smart contracts, and analyzed 500,000+ code lines. We have worked on major public blockchains e.g., Ethereum, Binance, Cronos, Doge, Polygon, Avalanche, Metis, Fantom, Bitcoin Cash, Velas, Oasis, etc.

InterFi Network is built by engineers, developers, UI experts, and blockchain enthusiasts. Our team currently consists of 4 core members, and 6+ casual contributors.

Website: https://interfi.network

Email: hello@interfi.network

GitHub: https://github.com/interfinetwork

INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTERFI INTER
CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDENTIAL AUDIT REPORT CONFIDEN

Telegram (Engineering): https://t.me/interfiaudits

Telegram (Onboarding): https://t.me/interfisupport interfinetwork

hello@interfi.network

https://interfi.network

SMART CONTRACT AUDITS | SOLIDITY DEVELOPMENT AND TESTING

RELENTLESSLY SECURING PUBLIC AND PRIVATE BLOCKCHAINS