RMDT Coin

Security Assessment

CertiK Assessed on Nov 10th, 2025

CertiK Assessed on Nov 10th, 2025

RMDT Coin

The security assessment was prepared by CertiK.

Executive Summary

TYPES	ECOSYSTEM	METHODS
ERC-20, Vesting	Binance Smart Chain (BSC)	Formal Verification, Manual Review, Static Analysis

LANGUAGE	TIMELINE
Solidity	Preliminary comments published on 10/28/2025 Final report published on 11/10/2025

Vulnerability Summary

9	6	0	3	0
Total Findings	Resolved	Partially Resolved	Acknowledged	Declined

1	Centralization	1 Acknowledged	Centralization findings highlight privileged roles & functions and their capabilities, or instances where the project takes custody of users' assets.
0	Critical		Critical risks are those that impact the safe functioning of a platform and must be addressed before launch. Users should not invest in any project with outstanding critical risks.
1	Major	1 Acknowledged	Major risks may include logical errors that, under specific circumstances, could result in fund losses or loss of project control.
2	Medium	1 Resolved, 1 Acknowledged	Medium risks may not pose a direct risk to users' funds, but they can affect the overall functioning of a platform.
3	Minor	3 Resolved	Minor risks can be any of the above, but on a smaller scale. They generally do not compromise the overall integrity of the project, but they may be less efficient than other solutions.
2	Informational	2 Resolved	Informational errors are often recommendations to improve the style of the code or certain operations to fall within industry best practices. They usually do not affect the overall functioning of the code.

TABLE OF CONTENTS | RMDT COIN

CODEBASE | RMDT COIN

Repository

https://testnet.bscscan.com/address/0x3B21003d04B3469f023757Ee694aFB2d9B57aa6b#code
https://testnet.bscscan.com/address/0x394EC10C49F79B5F7B1781755A7f7dba0A7c076a#code
https://testnet.bscscan.com/address/0x297914A8cb286b8cd5c7A2B0867e837c441Dd159#code
https://testnet.bscscan.com/address/0x8ffD4967180c4460A5f2BF42Af185D3a71311B1F#code

AUDIT SCOPE | RMDT COIN

testnet
contracts/RMDT_Token_Gasless.sol
contracts/RMDT_Vesting_Gasless.sol

APPROACH & METHODS | RMDT COIN

This audit was conducted for RMDT Coin to evaluate the security and correctness of the smart contracts associated with the RMDT Coin project. The assessment included a comprehensive review of the in-scope smart contracts. The audit was performed using a combination of Formal Verification, Manual Review, and Static Analysis.

The review process emphasized the following areas:

- Architecture review and threat modeling to understand systemic risks and identify design-level flaws.
- Identification of vulnerabilities through both common and edge-case attack vectors.
- Manual verification of contract logic to ensure alignment with intended design and business requirements.
- Dynamic testing to validate runtime behavior and assess execution risks.
- Assessment of code quality and maintainability, including adherence to current best practices and industry standards.

The audit resulted in findings categorized across multiple severity levels, from informational to critical. To enhance the project's security and long-term robustness, we recommend addressing the identified issues and considering the following general improvements:

- Improve code readability and maintainability by adopting a clean architectural pattern and modular design.
- Strengthen testing coverage, including unit and integration tests for key functionalities and edge cases.
- Maintain meaningful inline comments and documentations.
- Implement clear and transparent documentation for privileged roles and sensitive protocol operations.
- Regularly review and simulate contract behavior against newly emerging attack vectors.

FINDINGS | RMDT COIN

9	0	1	1	2	3	2
Total Findings	Critical	Centralization	Major	Medium	Minor	Informational

This report has been prepared for RMDT Coin to identify potential vulnerabilities and security issues within the reviewed codebase. During the course of the audit, a total of 9 issues were identified. Leveraging a combination of Formal Verification, Manual Review & Static Analysis the following findings were uncovered:

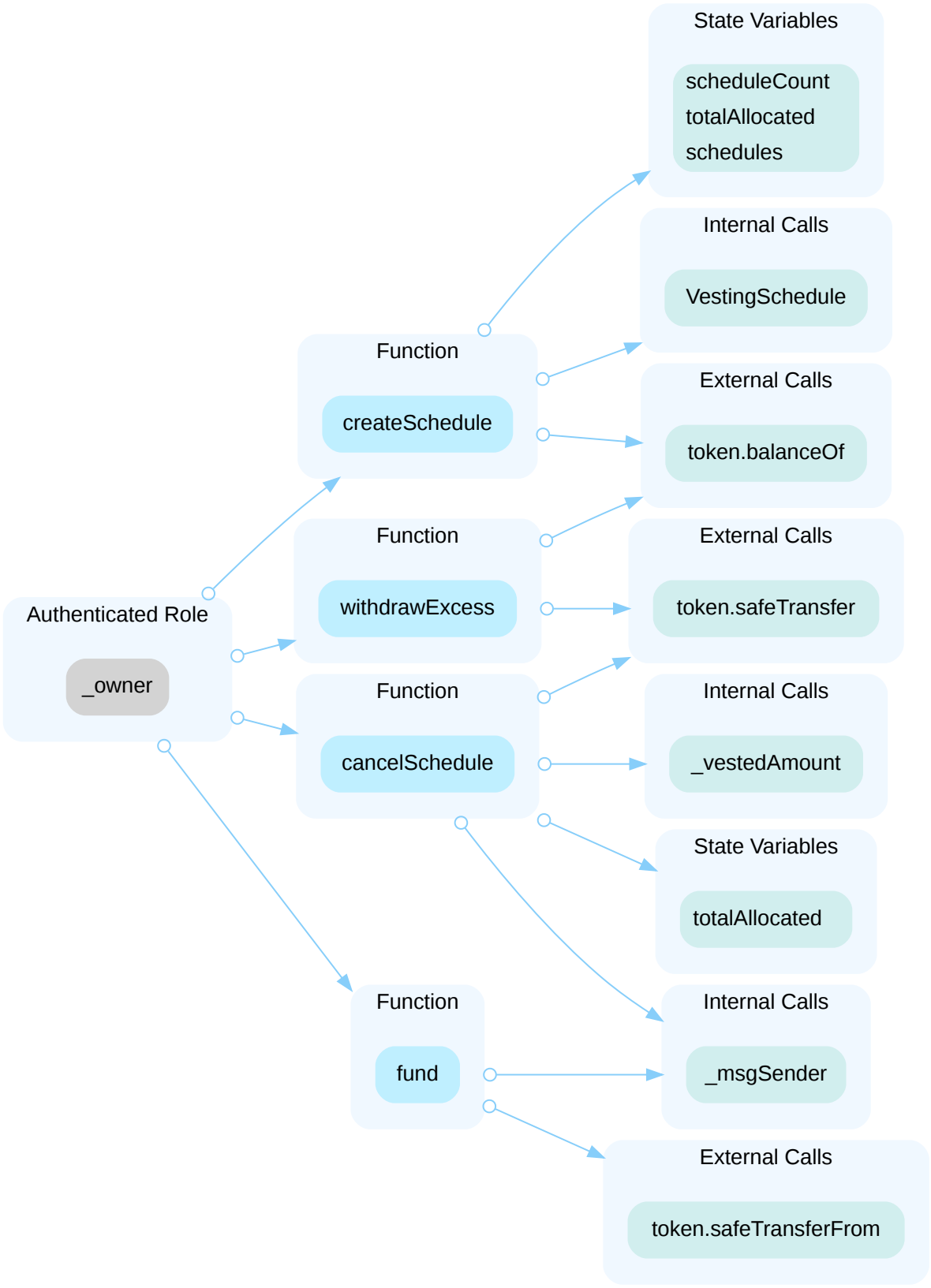
ID	Title	Category	Severity	Status
RMC-02	**Centralization Related Risks**	**Centralization**	**Centralization**	**Acknowledged**
RMC-03	**Initial Token Distribution**	**Centralization**	**Major**	**Acknowledged**
RMC-04	Unknown Implementation Of TrustedForwarder	Logical Issue	Medium	Acknowledged
RMC-05	`withdrawExcess` Does Not Account For Claimed Tokens	Volatile Code	Medium	Resolved
RMC-06	`MetaTransferRelayed` , `MetaClaimRelayed` And `MetaClaimAllRelayed` Event Logs Original User Instead Of Relayer	Logical Issue	Minor	Resolved
RMC-07	Incorrect Total Allocation Validation In `createSchedule` Function	Volatile Code	Minor	Resolved
RMC-08	Confusion Regarding Claimable Balances	Design Issue	Minor	Resolved
RMC-09	Redundant Override Of `transfer` Function	Design Issue	Informational	Resolved
RMC-10	Unused Custom Error `ScheduleExists`	Code Optimization	Informational	Resolved

RMC-02 | Centralization Related Risks

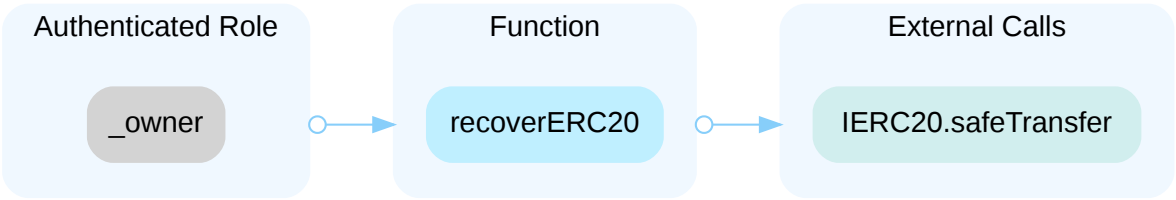
Category	Severity	Location	Status
Centralization	Centralization	contracts/RMDT_Token_Gasless.sol: 94; contracts/RMDT_Vesting_Gasless.sol: 109, 118, 332, 362	Acknowledged

Description

In the contract `RMDTVestingGasLess` the role `_owner` has authority over the functions shown in the diagram below. Any compromise to the `_owner` account may allow the hacker to take advantage of this authority and fund the contract, create a vesting schedule for a beneficiary, cancel a schedule, withdraw tokens in this contract that exceed total allocated.



In the contract `RMDTTokenGasLess` the role `_owner` has authority over the functions shown in the diagram below. Any compromise to the `_owner` account may allow the hacker to take advantage of this authority and recover accidentally sent ERC20 tokens.



Recommendation

The risk describes the current project design and potentially makes iterations to improve in the security operation and level of decentralization, which in most cases cannot be resolved entirely at the present stage. We advise the client to carefully manage the privileged account's private key to avoid any potential risks of being hacked. In general, we strongly recommend centralized privileges or roles in the protocol be improved via a decentralized mechanism or smart-contract-based accounts with enhanced security practices, e.g., multisignature wallets. Indicatively, here are some feasible suggestions that would also mitigate the potential risk at a different level in terms of short-term, long-term and permanent:

Short Term:

Timelock and Multi sign (⅔, ⅗) combination *mitigate* by delaying the sensitive operation and avoiding a single point of key management failure.

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;
  AND
- Assignment of privileged roles to multi-signature wallets to prevent a single point of failure due to the private key compromised;
  AND
- A medium/blog link for sharing the timelock contract and multi-signers addresses information with the public audience.

Long Term:

Timelock and DAO, the combination, *mitigate* by applying decentralization and transparency.

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;
  AND
- Introduction of a DAO/governance/voting module to increase transparency and user involvement.
  AND
- A medium/blog link for sharing the timelock contract, multi-signers addresses, and DAO information with the public audience.

Permanent:

Renouncing the ownership or removing the function can be considered *fully resolved*.

- Renounce the ownership and never claim back the privileged roles.
  OR

- Remove the risky functionality.

Alleviation

[CertiK, 11/06/2025]: In the contract `RMDTTokenGasLess`, the owner role has been replaced with the `RESCUER_ROLE`, which is responsible for recovering accidentally sent ERC20 tokens. A new `WRAPPER_MANAGER_ROLE` has also been added to manage the wrapper contract address.

In the contract `RMDTVestingGasLess`, the owner role has been replaced with the `VESTING_MANAGER_ROLE`, which is responsible for funding the contract and canceling vesting schedules, as well as withdrawing excess tokens. Additionally, a `SCHEDULER_ROLE` has been introduced to create vesting schedules for beneficiaries.

It should be noted that the centralization risk issue still exists. CertiK strongly encourages the project team to periodically revisit the private key security management of all addresses related to centralized roles.

RMC-03 | Initial Token Distribution

Category	Severity	Location	Status
Centralization	**Major**	**contracts/RMDT_Token_Gasless.sol: 48**	**Acknowledged**

Description

`50_000_000_000` of the `RMDT` tokens are sent to the contract deployer or one or several externally-owned account (EOA) addresses. This is a centralization risk because the deployer or the owner(s) of the EOAs can distribute tokens without obtaining the consensus of the community. Any compromise to these addresses may allow a hacker to steal and sell tokens on the market, resulting in severe damage to the project.

Recommendation

It is recommended that the team be transparent regarding the initial token distribution process. The token distribution plan should be published in a public location that the community can access. The team should make efforts to restrict access to the private keys of the deployer account or EOAs. A multi-signature (⅔, ⅗) wallet can be used to prevent a single point of failure due to a private key compromise. Additionally, the team can lock up a portion of tokens, release them with a vesting schedule for long-term success, and deanonymize the project team with a third-party KYC provider to create greater accountability.

Alleviation

[RMDT Coin, 11/06/2025]: Currently Gnosis safe does not support BNB smart chain testnet as their supportive chain , we are using EOA . When later used for production we would be using multi signature wallet from Gnosis safe

RMC-04 | Unknown Implementation Of TrustedForwarder

Category	Severity	Location	Status
Logical Issue	Medium	contracts/RMDT_Token_Gasless.sol: 39; contracts/RMDT_Vesting_Gasless.sol: 95	Acknowledged

Description

Both the `RMDTVestingGasLess` and the `RMDTTokenGasLess` contracts extend `ERC2771Context` and utilize a `trustedForwarder` address for meta transactions. The audit scope does not contain `trustedForwarder`, and its signature verification scheme is unknown. If the signature verification is implemented incorrectly, the `trustedForwarder` contract could potentially execute transaction on behalf of other users without their consent.

Reference: EIP-2771

Recommendation

The team should make every effort to ensure the functional correctness of out-of-scope contracts.

Alleviation

[RMDT Coin, 11/06/2025]: Trusted forwarder is a inherited from openzeppeline's MinimalForwarder

RMC-05 | `withdrawExcess` Does Not Account For Claimed Tokens

Category	Severity	Location	Status
Volatile Code	Medium	contracts/RMDT_Vesting_Gasless.sol: 365	Resolved

Description

The `withdrawExcess` function is intended to allow the owner to withdraw tokens in the contract that exceed the total allocated amount.

However, the function does not subtract the already claimed tokens from the total allocated calculation. As a result, tokens that have already been claimed by beneficiaries are still considered “allocated,” effectively locking them in the contract. This may prevent withdrawal of tokens that are actually available.

Impact:

- Owner cannot recover excess tokens that are technically free/unallocated.
- Causes unnecessary token lock-up and inefficient fund management.
- Could lead to operational friction if large amounts of unclaimed tokens accumulate.

Recommendation

Update the `withdrawExcess` logic to subtract the total claimed amount from the total allocated when computing withdrawable tokens. Only tokens that are truly unallocated and unclaimed should be eligible for withdrawal.

Alleviation

The client revised the code and resolved this issue on address 0x297914A8cb286b8cd5c7A2B0867e837c441Dd159

RMC-06 `MetaTransferRelayed` , `MetaClaimRelayed` And `MetaClaimAllRelayed` Event Logs Original User Instead Of Relayer

Category	Severity	Location	Status
Logical Issue	Minor	contracts/RMDT_Token_Gasless.sol: 74; contracts/RMDT_Vesting_Gasless.sol: 280, 323	Resolved

Description

The `MetaTransferRelayed` event has its first parameter labeled as `relayer`. However, the contract RMDTTokenGasLess extends ERC2771Context, meaning that _msgSender() returns the original signer (the user who authorized the meta-transaction) rather than the actual relayer who submitted the transaction.

As a result:

- The event incorrectly logs the user's address in the relayer field.
- Off-chain systems or analytics expecting the true relayer cannot reliably identify who executed the transaction.

```
32  event MetaTransferRelayed(address indexed relayer, address indexed from, address indexed to, uint256 amount);
```

The `MetaClaimRelayed` and `MetaClaimAllRelayed` events in the `RMDTVestingGasLess` contract have the same issue.

Recommendation

Emit the actual `msg.sender` (the relayer) in the event instead of `_msgSender()`.

Alleviation

The client revised the code and resolved this issue on address 0xB5F678579d028382Ef59D4E22C0a153E3F8cB081 and 0x8ffD4967180c4460A5f2BF42Af185D3a71311B1F.

RMC-07 | Incorrect Total Allocation Validation In `createSchedule` Function

Category	Severity	Location	Status
Volatile Code	Minor	contracts/RMDT_Vesting_Gasless.sol: 132~133	Resolved

Description

The `createSchedule` function validates that the new total allocated amount is less than the contract's current token balance to ensure sufficient funds for beneficiary claims.

However, the computed "new total allocated" value includes the claimed amount, even though claimed tokens have already been transferred out and no longer represent future obligations.

As a result, after a beneficiary claims their vested tokens, the contract balance increases, but the total allocated still counts the claimed amount. This causes the validation to fail unnecessarily unless the owner transfers additional tokens into the contract.

Impact:

- The owner may be forced to deposit more tokens than actually required.
- Vesting schedules may fail to be created even though sufficient unclaimed tokens exist.

Recommendation

Adjust the allocation validation logic to exclude previously claimed amounts from the total allocated computation. Only unclaimed or future claimable amounts should be considered when ensuring sufficient contract balance.

Alleviation

The client revised the code and resolved this issue on address 0x297914A8cb286b8cd5c7A2B0867e837c441Dd159

RMC-08 | Confusion Regarding Claimable Balances

Category	Severity	Location	Status
Design Issue	Minor	contracts/RMDT_Vesting_Gasless.sol: 332	Resolved

Description

The `cancelSchedule` function cancels an existing vesting schedule and returns unvested tokens to the owner while updating `totalAllocation`. However, any tokens that are already claimable by the beneficiary are not automatically transferred. Because `claimableOf` calculates claimable tokens based on `totalAllocation`, the `claimableOf` after cancellation returns a different value compared to before cancellation.

Recommendation

Automatically transfer all `claimable` tokens to the beneficiary at the time of schedule cancellation.

Alleviation

The client revised the code and resolved this issue on address 0x297914A8cb286b8cd5c7A2B0867e837c441Dd159

RMC-09 | Redundant Override Of `transfer` Function

Category	Severity	Location	Status
Design Issue	Informational	contracts/RMDT_Token_Gasless.sol: 80	Resolved

Description

The contract overrides the standard ERC20.transfer function but keeps the exact same implementation as the parent contract. This override does not introduce any new logic or modify existing behavior.

Recommendation

Remove the redundant override of the transfer function and rely on the inherited implementation from `ERC20`.

Alleviation

The client revised the code and resolved this issue on address 0x9f891BB8D18c3F14EB5955F4595bD0F6C81785f3.

RMC-10 | Unused Custom Error `ScheduleExists`

Category	Severity	Location	Status
Code Optimization	Informational	contracts/RMDT_Vesting_Gasless.sol: 25	Resolved

Description

The `RMDTVestingGasLess` contract defines a custom error `ScheduleExists` but never uses it in any function logic. This indicates leftover or incomplete code implementation.

Recommendation

Remove the unused custom error `ScheduleExists`, or implement proper validation logic (e.g., checking if a vesting schedule already exists before creating a new one) and use the defined error accordingly.

Alleviation

The client revised the code and resolved this issue on address 0xB5F678579d028382Ef59D4E22C0a153E3F8cB081.

OPTIMIZATIONS | RMDT COIN

ID	Title	Category	Severity	Status
RMC-01	Variables That Could Be Declared As Immutable	Gas Optimization	Optimization	Resolved

RMC-01 | Variables That Could Be Declared As Immutable

Category	Severity	Location	Status
Gas Optimization	Optimization	contracts/RMDT_Vesting_Gasless.sol: 56	Resolved

Description

The linked variables assigned in the constructor can be declared as `immutable`. Immutable state variables can be assigned during contract creation but will remain constant throughout the lifetime of a deployed contract. A big advantage of immutable variables is that reading them is significantly cheaper than reading from regular state variables since they will not be stored in storage.

Recommendation

We recommend declaring these variables as immutable. Please note that the `immutable` keyword only works in Solidity version `v0.6.5` and up.

Alleviation

The client revised the code and resolved this issue on address 0xB5F678579d028382Ef59D4E22C0a153E3F8cB081.

APPENDIX | RMDT COIN

Finding Categories

Categories	Description
Gas Optimization	Gas Optimization findings do not affect the functionality of the code but generate different, more optimal EVM opcodes resulting in a reduction on the total gas cost of a transaction.
Volatile Code	Volatile Code findings refer to segments of code that behave unexpectedly on certain edge cases and may result in vulnerabilities.
Logical Issue	Logical Issue findings indicate general implementation issues related to the program logic.
Centralization	Centralization findings detail the design choices of designating privileged roles or other centralized controls over the code.
Design Issue	Design Issue findings indicate general issues at the design level beyond program logic that are not covered by other finding categories.

DISCLAIMER | CERTIK

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to you ("Customer" or the "Company") in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes, nor may copies be delivered to any other person other than the Company, without CertiK's prior written consent in each instance.

This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts CertiK to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. CertiK's position is that each company and individual are responsible for their own due diligence and continuous security. CertiK's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

The assessment services provided by CertiK is subject to dependencies and under continuing development. 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. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third-parties.

ALL SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR 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, CERTIK HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, CERTIK SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM COURSE OF DEALING, USAGE, OR TRADE PRACTICE. WITHOUT LIMITING THE FOREGOING, CERTIK MAKES NO WARRANTY OF ANY KIND THAT THE SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET CUSTOMER'S OR ANY OTHER PERSON'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. WITHOUT LIMITATION TO THE FOREGOING, CERTIK PROVIDES NO WARRANTY OR

UNDERTAKING, AND MAKES NO REPRESENTATION OF ANY KIND THAT THE SERVICE WILL MEET CUSTOMER'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULTS, BE COMPATIBLE OR WORK WITH ANY OTHER SOFTWARE, APPLICATIONS, SYSTEMS OR SERVICES, OPERATE WITHOUT INTERRUPTION, MEET ANY PERFORMANCE OR RELIABILITY STANDARDS OR BE ERROR FREE OR THAT ANY ERRORS OR DEFECTS CAN OR WILL BE CORRECTED.

WITHOUT LIMITING THE FOREGOING, NEITHER CERTIK NOR ANY OF CERTIK'S AGENTS MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND, EXPRESS OR IMPLIED AS TO THE ACCURACY, RELIABILITY, OR CURRENCY OF ANY INFORMATION OR CONTENT PROVIDED THROUGH THE SERVICE. CERTIK WILL ASSUME NO LIABILITY OR RESPONSIBILITY FOR (I) ANY ERRORS, MISTAKES, OR INACCURACIES OF CONTENT AND MATERIALS OR FOR ANY LOSS OR DAMAGE OF ANY KIND INCURRED AS A RESULT OF THE USE OF ANY CONTENT, OR (II) ANY PERSONAL INJURY OR PROPERTY DAMAGE, OF ANY NATURE WHATSOEVER, RESULTING FROM CUSTOMER'S ACCESS TO OR USE OF THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS.

ALL THIRD-PARTY MATERIALS ARE PROVIDED "AS IS" AND ANY REPRESENTATION OR WARRANTY OF OR CONCERNING ANY THIRD-PARTY MATERIALS IS STRICTLY BETWEEN CUSTOMER AND THE THIRD-PARTY OWNER OR DISTRIBUTOR OF THE THIRD-PARTY MATERIALS.

THE SERVICES, ASSESSMENT REPORT, AND ANY OTHER MATERIALS HEREUNDER ARE SOLELY PROVIDED TO CUSTOMER AND MAY NOT BE RELIED ON BY ANY OTHER PERSON OR FOR ANY PURPOSE NOT SPECIFICALLY IDENTIFIED IN THIS AGREEMENT, NOR MAY COPIES BE DELIVERED TO, ANY OTHER PERSON WITHOUT CERTIK'S PRIOR WRITTEN CONSENT IN EACH INSTANCE.

NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST CERTIK WITH RESPECT TO SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS.

THE REPRESENTATIONS AND WARRANTIES OF CERTIK CONTAINED IN THIS AGREEMENT ARE SOLELY FOR THE BENEFIT OF CUSTOMER. ACCORDINGLY, NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH REPRESENTATIONS AND WARRANTIES AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST CERTIK WITH RESPECT TO SUCH REPRESENTATIONS OR WARRANTIES OR ANY MATTER SUBJECT TO OR RESULTING IN INDEMNIFICATION UNDER THIS AGREEMENT OR OTHERWISE.

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

Elevating Your Web3 Journey

Founded in 2017 by leading academics in the field of Computer Science from both Yale and Columbia University, CertiK is the largest blockchain security company that serves to verify the security and correctness of smart contracts and blockchain-based protocols. Through the utilization of our world-class technical expertise, alongside our proprietary, innovative tech, we're able to support the success of our clients with best-in-class security, all whilst realizing our overarching vision; provable trust for all throughout all facets of blockchain.