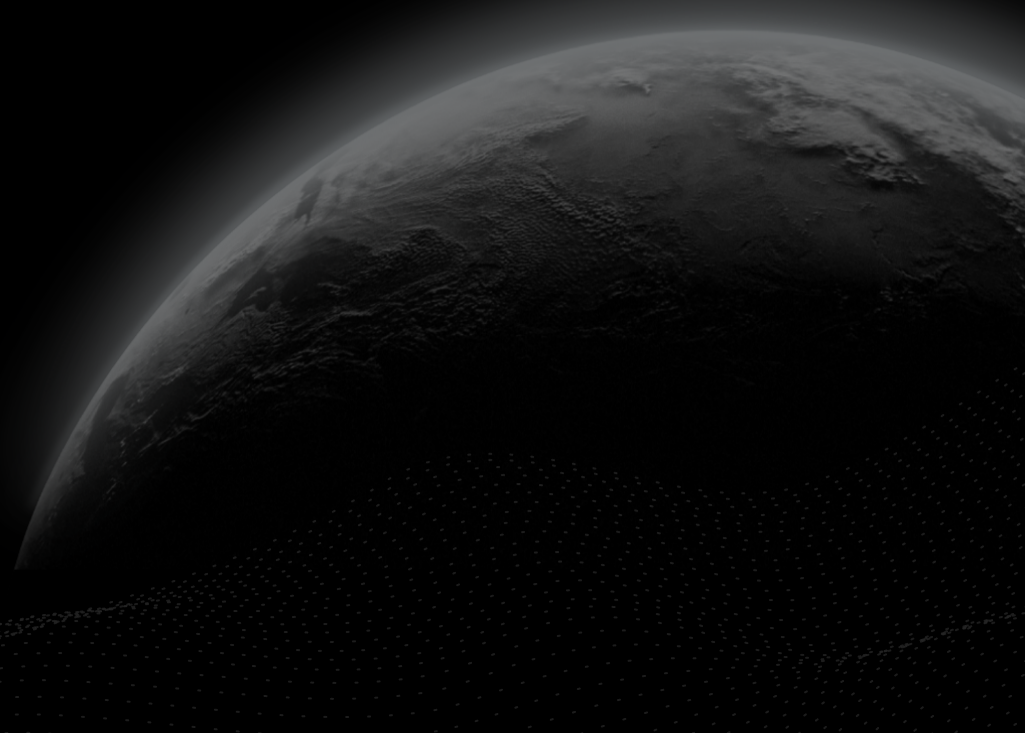
CERTIK

Security Assessment

MultiChain Foundation - Cardano

CertiK Verified on Nov 30th, 2022



CertiK Verified on Nov 30th, 2022

MultiChain Foundation - Cardano

The security assessment was prepared by CertiK, the leader in Web3.0 security.

Executive Summary

TYPES	ECOSYSTEM	METHODS
Bridge	Cardano	Manual Review, Static Analysis

LANGUAGE	TIMELINE	KEY COMPONENTS
Golang	Delivered on 11/30/2022	N/A

CODEBASE

https://github.com/anyswap/CrossChain-Router/

...View All

COMMITS

cec05b762aac4214df2b927ea1dc48a6477e8d92

...View All

Vulnerability Summary

8	3	0	1	4	0	0
Total Findings	Resolved	Mitigated	Partially Resolved	Acknowledged	Declined	Unresolved

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 Resolved	Major risks can include centralization issues and logical errors. Under specific circumstances, these major risks can lead to loss of funds and/or control of the project.
1	Medium	1 Resolved	Medium risks may not pose a direct risk to users' funds, but they can affect the overall functioning of a platform.
1	Minor	1 Acknowledged	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.
5	Informational	1 Resolved, 1 Partially Resolved, 3 Acknowledged	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 | MULTICHAIN FOUNDATION - CARDANO

CODEBASE | MULTICHAIN FOUNDATION - CARDANO

Repository

https://github.com/anyswap/CrossChain-Router/

Commit

cec05b762aac4214df2b927ea1dc48a6477e8d92

AUDIT SCOPE | MULTICHAIN FOUNDATION - CARDANO

19 files audited ● 4 files with Acknowledged findings ● 1 file with Partially Resolved findings
● 2 files with Resolved findings ● 12 files without findings

ID	File	SHA256 Checksum
ADD	tokens/cardano/address.go	7c0a9cc1bbda982e6417781690f66b168017c8a17c9c2b5e3874734f3eaacc40
BRI	tokens/cardano/bridge.go	8eaead84b6b31317650f0a3451f085c8ed3ccdccbd26e899d55e423df802810d
CCC	tokens/cardano/cardanoCmd.go	998b90ba3277328b5894acc81988d28fa36d3c5174844f160fe5179a6d131c74
VER	tokens/cardano/verifytx.go	9f007c4962176d0bada0d9a3c69c2ca22131f79456098b20e3451a25354e025f
INI	tokens/cardano/init.go	a21b92554af31a4ed3e1581d318cfd0dc7e3ab9d03ff1241fdc517c212ca28e9
BUI	tokens/cardano/buildtx.go	72221b8b0f81dcfbabe17a64b8e95e531834afd1c58c508d979d3110fe464491
SEN	tokens/cardano/sendtx.go	024aced637ac39c226f86dd3ab5f99cd23bea4e2506e7b8c2fa35f56cfb65dc9
SCD	tokens/cardano/tools/getStubChainID/main.go	739be7929f3920e22f5c70ce1e0aaa1ef8c63fe76f04f4090a97c59b8de04638
UCR	tokens/cardano/tools/queryUtxos/main.go	87e0449864f19850b0d5d915ee0fe18f3b4b564afa74f8a9201f9df5fdff6ffd
MAI	tokens/cardano/tools/scan/main.go	bb35be2a1d23c766646e5d90e1ea79f522a7d84c8bca23509dbcb228512bf9bd
TCR	tokens/cardano/tools/sendTransaction/main.go	4b2be0bc46819524025baaad9517e5cd7d23d0fa1fc71e273313aa274459d3cf
AGG	tokens/cardano/aggregate.go	9f7086f50af67ae3eaeb2a6e02c99c909d3f935eca9875ea5cf5b3ba832bd3d3
INS	tokens/cardano/instance.go	aca1665cab8dc1959d03066ee61594aebbf4c654bed8f525aabb5c769463b6ca
KEY	tokens/cardano/key.go	180d92fe2a063f92afe22a8b4484e5b72ca443b2af21025422ca064fc2590651

ID	File	SHA256 Checksum
REG	tokens/cardano/register.go	b6a514325ea513b875250389989edeb3ea0978774c5682aa1f27fcf3f588ec40
RPC	tokens/cardano/rpcClient.go	8cf4bdc58316788bff0d00f076dd3759fd01b4b2b9592c426571a6a570f9e870
SIG	tokens/cardano/signtx.go	391a581c656c0bedfe8c2eab3958d754b5fa4aac539a152d279ea49ac3e87f7e
TYP	tokens/cardano/type.go	a79bf47cde067e6798de233ed7e743ad9882dab44fe23cc8c29dd79b5c9e001a
UTI	tokens/cardano/utils.go	86ccb6510c2fb03a1f0d460d3a24b1f744734951852ac226666050e084ae550a

APPROACH & METHODS | MULTICHAIN FOUNDATION - CARDANO

This report has been prepared for MultiChain Foundation - Cardano to discover issues and vulnerabilities in the source code of the MultiChain Foundation - Cardano project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Manual Review and Static Analysis techniques.

The auditing process pays special attention to the following considerations:

- Testing the smart contracts against both common and uncommon attack vectors.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

The security assessment resulted in findings that ranged from critical to informational. We recommend addressing these findings to ensure a high level of security standards and industry practices. We suggest recommendations that could better serve the project from the security perspective:

- Testing the smart contracts against both common and uncommon attack vectors;
- Enhance general coding practices for better structures of source codes;
- Add enough unit tests to cover the possible use cases;
- Provide more comments per each function for readability, especially contracts that are verified in public;
- Provide more transparency on privileged activities once the protocol is live.

FINDINGS | MULTICHAIN FOUNDATION - CARDANO

8	0	1	1	1	5
Total Findings	Critical	Major	Medium	Minor	Informational

This report has been prepared to discover issues and vulnerabilities for MultiChain Foundation - Cardano. Through this audit, we have uncovered 8 issues ranging from different severity levels. Utilizing the techniques of Manual Review & Static Analysis to complement rigorous manual code reviews, we discovered the following findings:

ID	Title	Category	Severity	Status
BRI-01	Improper Overloading Of Functions	Logical Issue	Minor	Acknowledged
CAR-01	The Design Intent Of The TransactionChainingKeyCache	Logical Issue	Medium	Resolved
VER-01	Potential Vulnerability To Deep Rollback	Volatile Code	Major	Resolved
GLOBAL-01	Swap Rollback	Logical Issue	Informational	Acknowledged
GLOBAL-02	Process Review	Volatile Code	Informational	Acknowledged
BRI-02	Misleading Function Name	Coding Style	Informational	Acknowledged
CAR-02	Wrong Comments	Inconsistency	Informational	Resolved
CAR-03	Lack Of Input Validation	Volatile Code	Informational	Partially Resolved

BRI-01 | IMPROPER OVERLOADING OF FUNCTIONS

Category	Severity	Location	Status
Logical Issue	Minor	tokens/cardano/bridge.go: 82	Acknowledged

Description

The input param `url` is not used in the below function, thus this function has the same functionality as the same name function above this. Since this function will be used, it is extremely misleading.

```
func (b *Bridge) GetLatestBlockNumberOf(url string) (num uint64, err error) {
  if blockNumber, err := GetLatestBlockNumber(); err == nil {
    return blockNumber, nil
  } else {
    return 0, err
  }
}
```

Recommendation

It is recommended to either remove this method or refine the use of the entry.

Alleviation

The team acknowledged the finding, and decided to retain the code base unchanged.

CAR-01 THE DESIGN INTENT OF THE TRANSACTIONCHAININGKEYCACHE

Category	Severity	Location	Status
Logical Issue	Medium	tokens/cardano/buildtx.go: 331~337; tokens/cardano/sendtx.go: 18~29	Resolved

Description

As we can see in the function `SendTransaction()` (L8 sendtx.go), the `TransactionChainingKeyCache` record the UTXOs already consumed. And it will check the length of the `TransactionChainingKeyCache.SpentUtxoList`. If the length is more than 100, the function will delete the front elements of `SpentUtxoList` to keep the length no more than 100.

We found that the usage of the `TransactionChainingKeyCache` only occurs in the function `QueryUtxoOnChain()` (L324 buildtx.go). The `TransactionChainingKeyCache` filter the UTXOs which are not recorded in the `TransactionChainingKeyCache.SpentUtxoMap`.

Our understanding of the logic above is that the `TransactionChainingKeyCache` store the latest 100 spent UTXOs and filter the UTXOs to get the UTXOs not consumed when querying the UTXOs. But we don't understand the design intent of the `TransactionChainingKeyCache` clearly, and have the following doubts:

1. Using the top-level code, we see that `Aggregate()` is executed once a week, which we understand is to aggregate the UTXOs of the `mpc` addresses, how are these data generated?
2. The UXTOs after `aggregate()` should no longer exist, so why do they still need to be filtered?
3. If the data `b.GetUtxosByAddress()` contains the UTXOs already consumed, will the length 100 be enough? If the length is not enough, some data may be deleted but still needed to be used to filter the UTXOs in the function `QueryUtxoOnChain()`. In this case, the same UTXO may be used twice.

Recommendation

Suggest the client to reconsider the design here.

Alleviation

[MultiChain]:

1. It will do `QueryUtxoOnChain` again when aggregate() `aggregate.go` #L79
2. Yes. but it's harmless as well I think. The rest `uxtos` will be removed when tx success.
3. We decided to remove the cache when we got `rx` result. https://github.com/anyswap/CrossChain-Router/commit/4da72f4de05b818537a496abbaa93562334bd506

VER-01 | POTENTIAL VULNERABILITY TO DEEP ROLLBACK

Category	Severity	Location	Status
Volatile Code	Major	tokens/cardano/verifytx.go: 123~129	Resolved

Description

A **rollback** refers to the chain of events that occur when a node discovers that its local version of the blockchain is different from the canonical one that the other nodes agree on. In order to prevent a contradiction of state, the differing node must discard the last couple of blocks that are different from the target blockchain. This `discarding of blocks` is what is known as the **rollback**, and can ultimately call for a loss of transaction outputs and changes of state that occurred within the blocks that were discarded. Therefore, if a blockchain of length `n` has a rollback of length `k`, then the state of the blockchain after the rollback will be at block `n-k`.

Please see the attached link for further information regarding Rollbacks on Cardano

---

In the linked code, the block threshold check is not as clear as expected and maybe suffer from several unreliabilities.

1. In fact, the value of `lastHeight` is the slot number of the current block, not the height of the block. So the block threshold check here is actually to determine whether the difference from the slot of `txres` to the slot of the current block has exceeded the set slot threshold, not the block height. We know that the slot number of a block is not closely related to the height of the block. The same size slot may correspond to a different number of blocks. That is to say, 1000 slots may have 10 blocks, 5, or 3, etc. But the Cardano rolls back by counting blocks. Therefore, it is not very reasonable to use the slot gap as the threshold check of a block.
2. The value of `b.GetChainConfig().Confirmations` is set in the base configuration. We cannot be sure that it is effective in reducing the impact of rollback.
3. This check is skipped when the entry `allowUnstable` is true, so the setting of this parameter needs to be carefully considered.

An incorrect handle of rollback can allow the following exploits to occur:

Double spending UTxO on Cardano 1. A user wants to bridge some tokens from the Carano chain to another chain. 2. The user transfers some tokens to the `Mpc` address in the Cardano chain. 3. The `Mpc` address in the other chain will transfer tokens to the address user specified. 4. Unfortunately, the Cardano chain rolls back a few blocks. 5. The tokens that the `Mpc` address should have received are returned to the user.

Recommendation

The likelihood of a deep rollback is uncertain; however, the usage of a higher block threshold check will decrease the probability of a rollback affecting the multiChain.

According to the IOHK, rollbacks of 20 blocks or higher are categorized as **very deep** and are the most unlikely to occur. Therefore, we advise changing the block threshold check to ensure 20 blocks have passed before proceeding with a transaction.

We also recommend the team constantly monitor potential rollbacks affecting their protocol.

Alleviation

The client heeded the advice and resolved this issue in commit `4dcc6237a65ddb2a48485fe23b7025e83bc16ae0`. And the configuration is under the control of the MPC validators.

GLOBAL-01 | SWAP ROLLBACK

Category	Severity	Location	Status
Logical Issue	Informational		Acknowledged

Description

After the swap, the Cardano rollback may lead to loss to the user. Here's the flow:

1. The user swaps tokens cross-chain via the router.
2. The user transfers the tokens to the MPC address in the source chain.
3. The MPC address in the Cardano chain transfers tokens to the address specified by the user.
4. The Cardano chain rolls back.
5. The user didn’t get the tokens in the Cardano chain but lost tokens in the source chain.

Recommendation

Please review the design and decide whether to change the code.

1. There should be a retry mechanism in the router.
2. If rollback happens, it means the swap/bridge has failed, the tokens in the source chain should be returned to the user.

Alleviation

[Multichain]: The Multichain MPC validators do not mark the transaction sending user assets complete until the stable block height is reached on destination chains. If the Cardano chain rolls back before the stable block height is reached, the validators will send the transaction again.

If the Cardano chain rollback after the stable block height is reached, users can send ticket at ***help center*** and the operation team can deal with the case off-chain.

GLOBAL-02 | PROCESS REVIEW

Category	Severity	Location	Status
Volatile Code	Informational		Acknowledged

Description

Looking over the whole process, the `cardano` module we audited is just a cross-chain relay, which is similar to other chains like ripple and eth. All we can do is audit the implementation of these methods of the bridge interface. So it is hard for us to confirm that the whole swap process is complete and solid.

Recommendation

We encourage the team to focus on code that is out of audit scope to ensure they are safe and secure.

Alleviation

[Multichain Foundation]: Thanks for mentions. We will.

BRI-02 | MISLEADING FUNCTION NAME

Category	Severity	Location	Status
Coding Style	Informational	tokens/cardano/bridge.go: 72, 82	Acknowledged

Description

The result of the function `GetLatestBlockNumber()` is the slot number of the current block, not the height of the block. The discrepancy with the naming is easily misunderstood.

Recommendation

Consider using a proper function name.

Alleviation

The team acknowledged the finding, and decided to retain the code base unchanged.

CAR-02 | WRONG COMMENTS

Category	Severity	Location	Status
Inconsistency	Informational	tokens/cardano/bridge.go: 81; tokens/cardano/init.go: 41, 45; tokens/cardano/verifytx.go: 52	Resolved

Description

The linked comments are incorrect since there are not fit with the `cardano` module.

Recommendation

We recommend updating the mentioned comment.

Alleviation

The team heeded the advice and resolved this issue in commit `e5c47fcafc966e09ccb63e22d81c20031df7fe3f`.

CAR-03 | LACK OF INPUT VALIDATION

Category	Severity	Location	Status
Volatile Code	Informational	tokens/cardano/bridge.go: 96, 108; tokens/cardano/init.go: 42	Partially Resolved

Description

The input parameter of these functions should be verified as non empty value to prevent occurring unexpected errors and can save gas effectively.

Recommendation

It's recommended to check them before using them.

Alleviation

The team heeded the advice and partially resolved this issue in commit `ecb4196616ac9825d65b26d8e02abde486c6222a`.

OPTIMIZATIONS | MULTICHAIN FOUNDATION - CARDANO

ID	Title	Category	Severity	Status
ADD-01	Unused Function `PublicKeyToAddress()`	Coding Style	Optimization	Acknowledged
CAR-04	Codes For Test Only	Volatile Code	Optimization	Acknowledged

ADD-01 | UNUSED FUNCTION `PublicKeyToAddress()`

Category	Severity	Location	Status
Coding Style	Optimization	tokens/cardano/address.go: 19~21	Acknowledged

Description

The function `PublicKeyToAddress()` is not used and implemented.

Recommendation

It is recommended to remove the methods that are not used.

Alleviation

[Multichain Foundation]: This will be used in cli. so it's ok

CAR-04 | CODES FOR TEST ONLY

Category	Severity	Location	Status
Volatile Code	Optimization	tokens/cardano/cardanoCmd.go: 13, 14; tokens/cardano/verifytx.go: 13	Acknowledged

Description

The linked constants may be used for testing purposes only.

Recommendation

We recommend the usage of the `#[cfg(test)]` annotation to exclude test code from the production build and save storage in the deployment phase.

Alleviation

The team acknowledged the finding, and decided to retain the code base unchanged.

APPENDIX | MULTICHAIN FOUNDATION - CARDANO

Finding Categories

Categories	Description
Logical Issue	Logical Issue findings detail a fault in the logic of the linked code, such as an incorrect notion on how block.timestamp works.
Volatile Code	Volatile Code findings refer to segments of code that behave unexpectedly on certain edge cases that may result in a vulnerability.
Coding Style	Coding Style findings usually do not affect the generated byte-code but rather comment on how to make the codebase more legible and, as a result, easily maintainable.
Inconsistency	Inconsistency findings refer to functions that should seemingly behave similarly yet contain different code, such as a constructor assignment imposing different require statements on the input variables than a setter function.

Checksum Calculation Method

The "Checksum" field in the "Audit Scope" section is calculated as the SHA-256 (Secure Hash Algorithm 2 with digest size of 256 bits) digest of the content of each file hosted in the listed source repository under the specified commit.

The result is hexadecimal encoded and is the same as the output of the Linux "sha256sum" command against the target file.

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.

CertiK | Securing the Web3 World

Founded in 2017 by leading academics in the field of Computer Science from both Yale and Columbia University, CertiK is a leading 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.