CERTIK

Security Assessment

Kyoko III-g2g

Mar 31st, 2022

Table of Contents

Summary

This report has been prepared for Kyoko III-g2g to discover issues and vulnerabilities in the source code of the Kyoko III-g2g project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Static Analysis and Manual Review 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:

- 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.

Overview

Project Summary

Project Name	Kyoko III-g2g
Platform	Other
Language	Solidity
Codebase	https://github.com/kyoko-finance/kyoko-g2g-contract
Commit	1.099a95063313599a10fba84ea88bb1a118a7db3a 2.9dc77897c081bbe0bbf183fd6f1857e77429ba4c

Audit Summary

Delivery Date	Mar 31, 2022 UTC
Audit Methodology	Static Analysis, Manual Review

Vulnerability Summary

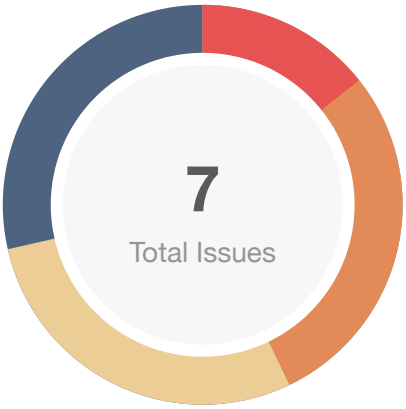
Vulnerability Level	Total	Pending	Declined	Acknowledged	Mitigated	Partially Resolved	Resolved
Critical	1	0	0	0	0	0	1
Major	2	0	0	2	0	0	0
Medium	0	0	0	0	0	0	0
Minor	2	0	0	0	0	0	2
Informational	2	0	0	0	0	0	2
Discussion	0	0	0	0	0	0	0

Audit Scope

ID	File	SHA256 Checksum
CSG	credit/CreditSystem.sol	a435c5d8b037102b13bf4f34370a191dabb52f29255a9aa558dc929d17d64473
CGC	lendingpool/Collector.sol	b58f46d036b282206ac9d5b475be132d6f4a0a3175c4dc0a3232e14bdc417c73
DTG	lendingpool/DataTypes.sol	7cc02614361aaf0788696b90ad5abad6ca7fa7ccd0694774e47b603a6c34c864
KLP	lendingpool/KyokoLendingPool.sol	ceae38f2513af0374fb822a033b008891ddd74c0ce18a485926a49667f11be70
LPS	lendingpool/LendingPoolStorage.sol	6595134c2057d7d9fdae1268d63a11cfd3a6f020457afc6e116ab3c6088959f6
GLG	libraries/GenericLogic.sol	c69b2622363ad37a3e74c612809525d29fc30e4638b2ec4478262e80009546b0
KMG	libraries/KyokoMath.sol	62d337ffb27bc18cee15709767829877560878dab02b1b7f09a15fb29005f381
MUG	libraries/MathUtils.sol	bde8a50a35337003cf9e3e9aa08a285baf3bc19fc938f6084564d6e9ad1c7a37
PMG	libraries/PercentageMath.sol	a2c18db81769301d48d28c770928236ffa8bd98fd9f50e7f5c4fdd8b737795b7
RLG	libraries/ReserveLogic.sol	3424c12afe738f1e1761995f9292b2157a957b61070493a7130b17bf5683e29c
VLG	libraries/ValidationLogic.sol	2d6c6b90690468fc2aaeab40a578f278040f403e383a3c54c2eda3040cea99f3
DTB	token/DebtTokenBase.sol	3e36df758c614351dd858191c1fda9fb5696a1ef7e60e6f3d8fa94249a54fe58
DRI	token/DefaultReserveInterestRateStrategy.sol	cb12ef2dd2af73e24f751f599847e857a9f910bd9aa140c1ecfe1ad74fc95888
IER	token/IncentivizedERC20.sol	718c094df26e98e8d2bdaacafb33a8917609e4ce0eb2bbf790fecb18869e616d

ID	File	SHA256 Checksum
KTG	token/KToken.sol	9672ed9a22319615fc0d2d9e400a28ad106eb5f609a05646bd92232f22c9c808
VDT	token/VariableDebtToken.sol	61d893b08ce9e40dd1f26e6ce7ab6f5d3184052874f86022124db1165abd2836

Findings



Critical	**1** (14.29%)
Major	**2** (28.57%)
Medium	**0** (0.00%)
Minor	**2** (28.57%)
Informational	**2** (28.57%)
Discussion	**0** (0.00%)

ID	Title	Category	Severity	Status
Kyoko-01	Centralization Related Risks	**Centralization / Privilege**	**Major**	Acknowledged
Kyoko-02	Financial Models	Volatile Code	Major	Acknowledged
GLG-01	Confusing Logic Of `balanceDecreaseAllowed()`	Logical Issue	Minor	Resolved
IER-01	Variable Never Used	Coding Style	Informational	Resolved
KLP-01	Lack of debit line check	Logical Issue	Critical	Resolved
KLP-02	Missing Emit Events	Coding Style	Minor	Resolved
VLG-01	Redundant Parameter	Coding Style	Informational	Resolved

Kyoko-01 | Centralization Related Risks

Category	Severity	Location	Status
Centralization / Privilege	Major	Global	Acknowledged

Description

In the contract `CreditSystem.sol`, the role `ROLE_CREDIT_MANAGER` has authority over the following functions:

- setG2GCreditLine(): set the debit limit of the `G2G` user.
- setG2GActive(): put the user into the `G2G` whitelist and set the `G2G` flag.
- setCCALCreditLine(): set the debit limit of the `CCAL` user.
- setCCALActive(): put the user into the `CCAL` whitelist and set the `CCAL` flag.
- removeG2GCredit(): remove the user from the `G2G` whitelist and set the flag.
- removeCCALCredit(): remove the user from the `CCAL` whitelist and set the flag.

Any compromise to the `ROLE_CREDIT_MANAGER` account may allow a hacker to take advantage of this authority and modify the debit limit of the specified user.

In the contract `KyokoLendingPool.sol`, the role `LENDING_POOL_ADMIN` has authority over the following functions:

- initReserve(): init the information of the reserve.
- setPause(): pause/unpause the protocol.
- setReserveInterestRateStrategyAddress(): set the address of the rate calculate strategy contract.
- setReserveFactor(): set the reserve factor which can decide the interest the depositors can get.
- setActive(): activate/inactivate the reserve.
- setCreditStrategy(): set the address of the `CreditSystem` which store the information that the user can borrow how many tokens.

Any compromise to the `LENDING_POOL_ADMIN` account may allow a hacker to take advantage of this authority and set the `credit system` to give some users A higher debt limit.

In the contract `KToken.sol`, the role `_pool` has authority over the following functions:

- burn(): burn tokens and transfer the corresponding amount to the user.
- mint(): mint tokens.
- mintToTreasury(): mint tokens to the treasury address.

- transferUnderlyingTo(): transfer the underlying assets to the specified address.

Any compromise to the `_pool` account may allow a hacker to take advantage of this authority and transfer the assets to other addresses.

In the contract `VariableDebtToken.sol`, the role `_pool` has authority over the following functions:

- burn(): burn tokens.
- mint(): mint tokens.

Any compromise to the `_pool` account may allow a hacker to take advantage of this authority and clear off some user debts.

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., multi-signature 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.

Noted: Recommend considering the long-term solution or the permanent solution. The project team shall make a decision based on the current state of their project, timeline, and project resources.

Alleviation

The Kyoko team acknowledged this finding, and team would use multi-sig wallet to manage the contracts `CreditSystem` and `KyokoLendingPool`.

Kyoko-02 | Financial Models

Category	Severity	Location	Status
Volatile Code	Major	Global	Acknowledged

Description

The main functions of the Kyoko D2D protocol can be described as follows:

1. Users can deposit some Stable Coins.
2. Some DAOs that are already registered in the credit system can borrow some tokens.
3. If the tokens in `KyokoLendingPool` are borrowed, the depositor can get some interest. If no DAO borrows tokens, there is no interest.
4. The borrower repays the tokens and pays some interest.

There are some issues to confirm.

1. DAOs can borrow tokens without any collateral. How can Kyoko prevent the borrower from not repaying their assets?
2. `creditLine` only restricts how many tokens DAO can borrow, but not the value of tokens DAO can borrow. Are all the tokens that can be borrowed Stable Coins? If not, there are problems that lower price tokens will never be borrowed.
3. From `KyokoLendingPool` we can see that if the user wants to withdraw more than the amount held by `KToken`, then the withdrawal will fail.
4. In `GenericLogic.calculateUserAccountData()`, it assumes that all tokens have decimals less than 18 (1 ether). Is it possible that some tokens have a decimal point greater than 18?

Recommendation

Financial models of blockchain protocols need to be resilient to attacks. They need to pass simulations and verifications to guarantee the security of the overall protocol.

The financial model of this protocol is not in the scope of this audit.

Alleviation

[Koyoko]: Currently, the whitelist has a strict entry and vetting mechanism, which is similar to a kind of "guarantor" mechanism through official partners and investment institutions. And then a community vote is conducted, and only DAOs or guilds with potential investment value or strong trust will pass the vetting

process. At this stage, only credit loans for small stable coins will be considered, other coins and prophecy machines will not be considered for now.

At the beginning of the business, the project will provide an initial pool of funds. The higher the utilization of funds, the higher the borrowing rate and deposit rate, and this model will motivate borrowers to repay and attract users to deposit.

In the current version, the protocol only supports tokens with a decimal point less than or equal to 18 decimal places.

GLG-01 | Confusing Logic Of `balanceDecreaseAllowed()`

Category	Severity	Location	Status
Logical Issue	Minor	libraries/GenericLogic.sol: 32~49	Resolved

Description

The function `balanceDecreaseAllowed()` will always return `true`. And the name of this function seems to be inconsistent with the function logic. What is this function used for?

Recommendation

We recommend returning the correct value and ensuring that the function name matches the logic.

Alleviation

The Kyoko team removed the function `balanceDecreaseAllowed()` and struct `balanceDecreaseAllowedLocalVars` in commit `8acec0ab9db6637b97cec3bf4431b312a46b6d74`.

IER-01 | Variable Never Used

Category	Severity	Location	Status
Coding Style	Informational	token/IncentivizedERC20.sol: 208	Resolved

Description

The `oldRecipientBalance` has been declared but never accessed.

Recommendation

Consider deleting it.

Alleviation

The Kyoko team removed the variable never used in commit `b0af176448abb87b8b52d19b4a67b142d406e68c`.

KLP-01 | Lack Of Debit Line Check

Category	Severity	Location	Status
Logical Issue	Critical	lendingpool/KyokoLendingPool.sol: 324~327	Resolved

Description

```
324 (
325   uint256 totalDebtInWEI,
326   uint256 availableBorrowsInWEI
327 ) = getUserAccountData(vars.user);
```

```
337 IVariableDebtToken(reserve.variableDebtTokenAddress).mint(
338   vars.user,
339   vars.onBehalfOf,
340   vars.amount,
341   reserve.variableBorrowIndex
342 );
```

The function `_executeBorrow()` is called by the function `borrow()` and the input parameter `vars.user` is `msg.sender`. In line 327, the code will get the available borrow amount for `msg.sender`. But in line 337, the debt tokens will be minted into the `vars.onBehalfOf` account. This means that an invested user can set an allowance to some other users to borrow any asset tokens without any restrictions.

Recommendation

We recommend reviewing this function and the other `ValidationLogic` functions to ensure that each validation logic is correct.

Alleviation

The Kyoko team heeded our advice and removed the function `approveDelegation()` in commit `9dc77897c081bbe0bbf183fd6f1857e77429ba4c`.

KLP-02 | Missing Emit Events

Category	Severity	Location	Status
Coding Style	Minor	lendingpool/KyokoLendingPool.sol: 412~417	Resolved

Description

There should always be events emitted in the sensitive functions that are controlled by centralization roles.

Recommendation

It is recommended emitting events for the sensitive functions that are controlled by centralization roles.

Alleviation

The Kyoko team emitted an event for the `initReserve()` function in commit `b0af176448abb87b8b52d19b4a67b142d406e68c`.

VLG-01 | Redundant Parameter

Category	Severity	Location	Status
Coding Style	Informational	libraries/ValidationLogic.sol: 50~51	Resolved

Description

The parameters `reserves` and `reservesCount` didn't use in the function `validateWithdraw()`.

Recommendation

Consider deleting them.

Alleviation

The Kyoko team removed the parameters never used in commit `b0af176448abb87b8b52d19b4a67b142d406e68c`.

Appendix

Finding Categories

Centralization / Privilege

Centralization / Privilege findings refer to either feature logic or implementation of components that act against the nature of decentralization, such as explicit ownership or specialized access roles in combination with a mechanism to relocate funds.

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.

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

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.

About

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.