CERTIK

Preliminary Comments

Bloxxy

Dec 21st, 2021

Table of Contents

Summary

This report has been prepared for Bloxxy to discover issues and vulnerabilities in the source code of the Bloxxy 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	Bloxxy
Platform	Ethereum
Language	Solidity
Codebase	https://github.com/la-cucina/lac-token/
Commit	f660281ea4fece462e8dad47be7e61627f863e9f 741e10ac2f8a86ac3cbfddcac019171a98dc95c8

Audit Summary

Delivery Date	Dec 21, 2021
Audit Methodology	Static Analysis, Manual Review
Key Components	

Vulnerability Summary

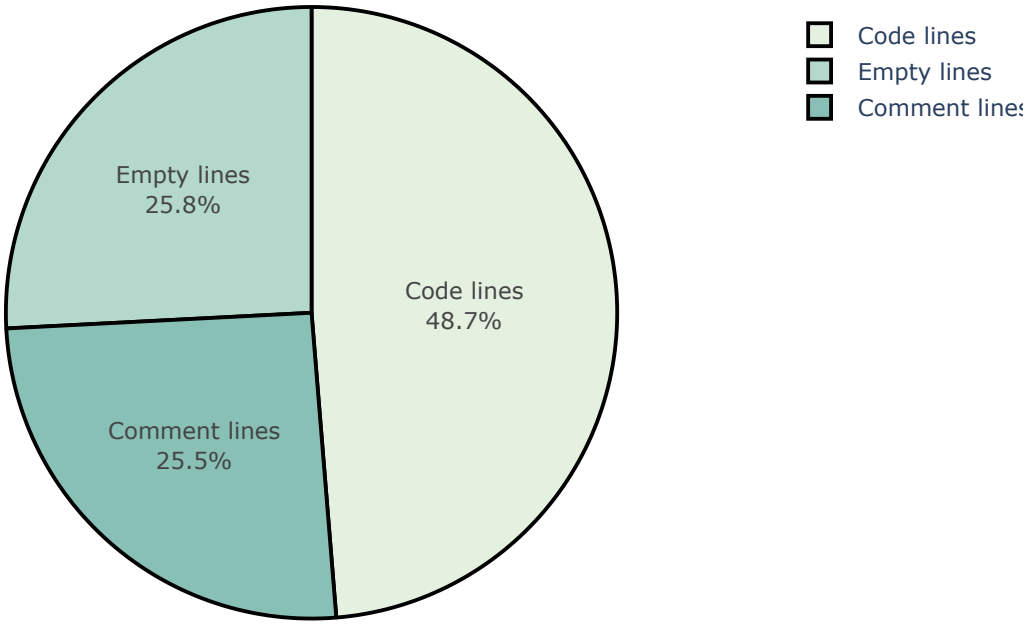
Vulnerability Level	Total	Pending	Declined	Acknowledged	Partially Resolved	Resolved
Critical	0	0	0	0	0	0
Major	2	0	0	1	0	1
Medium	1	0	0	0	0	1
Minor	0	0	0	0	0	0
Informational	2	0	0	0	0	2
Discussion	0	0	0	0	0	0

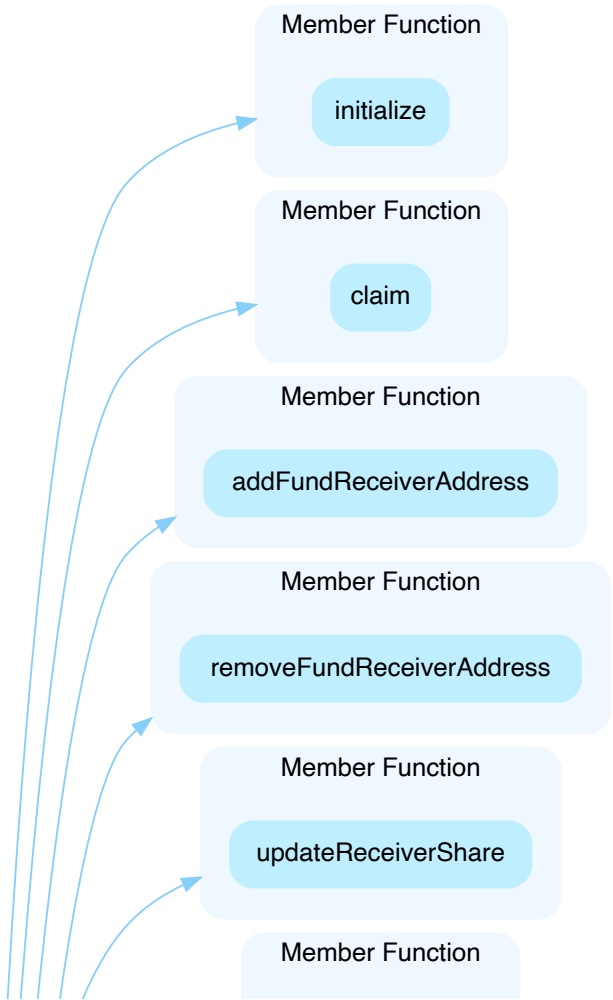
Audit Scope

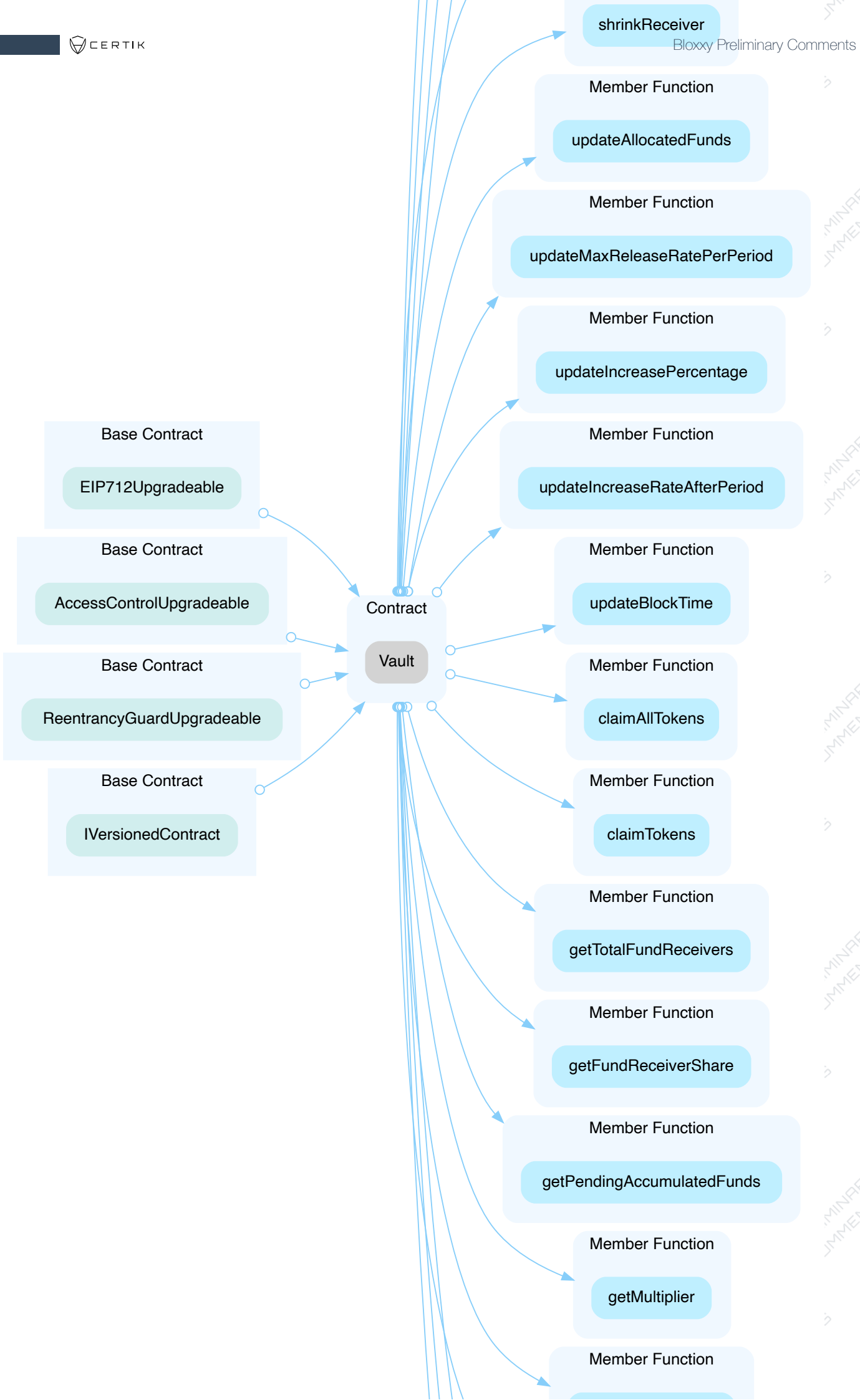
ID	File	SHA256 Checksum
IVC	interfaces/IVersionedContract.sol	348ad8746a4954f46a23e198a275d5825284adc85137bcb4b09265b30de5502d
LTU	library/LacTokenUtils.sol	a1aae63dadcfc9b3ff7de67649a73db46430b3497a3edd517d9948acfef438bc
LTB	LacToken.sol	83ff81799c85d586137c976bc3f88f32c925e6de267a83aace787111d97aecb8
VBC	Vault.sol	579c16159e201eedfa32c526a3842c07177f83545f8595e605f96d73521ad6a0

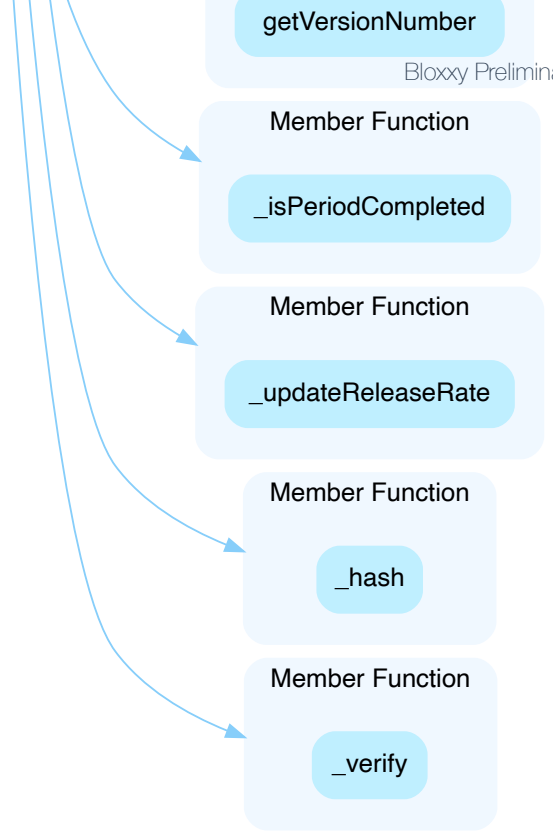
Diagrams

Source Line Chart






shrinkReceiver
Member Function
updateAllocatedFunds
Member Function
updateMaxReleaseRatePerPeriod
Member Function
updateIncreasePercentage
Base Contract
EIP712Upgradeable
Member Function
updateIncreaseRateAfterPeriod
Base Contract
AccessControlUpgradeable
Member Function
updateBlockTime
Contract
Base Contract
Vault
ReentrancyGuardUpgradeable
Member Function
claimAllTokens
Base Contract
Member Function
IVersionedContract
claimTokens
Member Function
getTotalFundReceivers
Member Function
getFundReceiverShare
Member Function
getPendingAccumulatedFunds
Member Function
getMultiplier
Member Function


getVersionNumber
Member Function
_isPeriodCompleted
Member Function
_updateReleaseRate
Member Function
_hash
Member Function
_verify

Findings

Critical	0 (0.00%)
Major	2 (40.00%)
Medium	1 (20.00%)
Minor	0 (0.00%)
Informational	2 (40.00%)
Discussion	0 (0.00%)

ID	Title	Category	Severity	Status
LTB-01	Initial Token Distribution	**Centralization / Privilege**	**Major**	Resolved
VBC-01	Centralization Risk	**Centralization / Privilege**	**Major**	Acknowledged
VBC-02	Missing Emit Events	Coding Style	Informational	Resolved
VBC-03	Unnecessary API Exposure	Coding Style	Informational	Resolved
VBC-04	Logical Flaw In `getPendingAccumulatedFunds()`	Logical Issue	Medium	Resolved

LTB-01 | Initial Token Distribution

Category	Severity	Location	Status
Centralization / Privilege	Major	projects/Bloxxy/contracts/LacToken.sol (eaca569): 26	Resolved

Description

All of the Lac tokens are sent to the contract deployer when deploying the contract. This could be a centralization risk as the deployer can distribute Lac tokens without obtaining the consensus of the community.

Recommendation

We recommend the team to be transparent regarding the initial token distribution process, and the team shall make enough efforts to restrict the access of the private key.

Alleviation

`[Bloxxy team]`: We have addressed the risk of LTB-01 | Initial Token Distribution in the following ways:

- The LAC token has been minted with a final, hard capped supply of 500 million. The final balance was sent to the issuer/deployer until the initial allocation, which will take place within the next 14 days (24th December, 2021). With 18% the LACs being allocated to the Team, 17% for our referral program and future business development and 5% as seed liquidity on our liquidity pool(s).
- 60% of all LAC tokens are destined to be locked in the Vault smart contract as rewards for our community members. The Vault will continuously release rewards over a period of 10 years. The below graph and table describes our current concept for the release schedule. However, the exact schedule will be set as soon as the LaCucina Pilot is over (by January 31st, 2022). Until then, it will be released at a fixed rate of ±8 LAC tokens per block. The final release schedule will be announced throughout our community and social media channels. Once the final schedule has been determined, our aim is to have the schedule coded into the smart contract, such that we (or anyone else in this respect) will not be able to change it again. As for transparency, the proposed allocation will be published on our Whitepaper, documentation and via our blog on Medium.
- With regards to the private key and access to it, the private key of the deployer is saved on a cold wallet device in a physical vault, in a secure location. Only one person has access to this.

For more information on LaCucina's LAC Tokenomics, learn more via our blog on Medium: https://medium.com/@lacucina/lac-tokenomics-part-i-928ca266e689.

VBC-01 | Centralization Risk

Category	Severity	Location	Status
Centralization / Privilege	Major	projects/Bloxxy/contracts/Vault.sol (eaca569): 176, 188, 204, 224, 271, 276, 281, 286, 294, 309	Acknowledged

Description

In the contract `Vault.sol`, the role `admin` has the authority over the following function:

- `addFundReceiverAddress()`
- `removeFundReceiverAddress()`
- `updateReceiverShare()`
- `shrinkReceiver()`
- `updateMaxReleaseRatePerPeriod()`
- `updateIncreasePercentage()`
- `updateIncreaseRateAfterPeriod()`
- `updateBlockTime()`
- `claimAllTokens()`
- `claimTokens()`

Any compromise to the `admin` account may allow the hacker to take advantage of this and manipulate the sensitive functionalities of the system, like claim tokens/all tokens from the contract.

Recommendation

We advise the client to carefully manage the `admin` account's private key to avoid any potential risks of being hacked. In general, we strongly recommend centralized privileges or roles in the protocol to be improved via a decentralized mechanism or smart-contract-based accounts with enhanced security practices, e.g., Multisignature wallets.

Indicatively, here is some feasible suggestions that would also mitigate the potential risk at the different level in term of short-term and long-term:

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;
- Assignment of privileged roles to multi-signature wallets to prevent a single point of failure due to the private key;
- Introduction of a DAO/governance/voting module to increase transparency and user involvement.

Alleviation

`[Bloxxy team]`: We have addressed the centralization risk of VBC-01 | Centralization Risk in the following ways:

- The privileged roles will be assigned to a multi-signature wallet, as suggested.
- Governance voting - Upon launch we are introducing a non-binding voting mechanic allowing our users to submit propositions, and vote on other user's proposals regarding future changes to the platform. In the future, this module might be upgraded to a full governance model, hence handing over the governance of some aspects of the platform, to the community.

VBC-02 | Missing Emit Events

Category	Severity	Location	Status
Coding Style	Informational	projects/Bloxxy/contracts/Vault.sol (eaca569): 176, 188, 204, 224, 271, 276, 281, 286, 294, 309	Resolved

Description

The function that affects the status of sensitive variables should be able to emit events as notifications.

- `addFundReceiverAddress()`
- `removeFundReceiverAddress()`
- `updateReceiverShare()`
- `shrinkReceiver()`
- `updateMaxReleaseRatePerPeriod()`
- `updateIncreasePercentage()`
- `updateIncreaseRateAfterPeriod()`
- `updateBlockTime()`
- `claimAllTokens()`
- `claimTokens()`

Recommendation

We advise the client to add events for sensitive actions, and emit them in the function.

Alleviation

`[Bloxxy team]`: We have fixed the VBC-02 | Missing Emit Events issue by adding the events in mentioned methods.

VBC-03 | Unnecessary API Exposure

Category	Severity	Location	Status
Coding Style	Informational	projects/Bloxxy/contracts/Vault.sol (eaca569): 244	Resolved

Description

If a function is not needed to be exposed to external user, then it visibility should be set to internal/private

Recommendation

We advise the client to update the visibility of `updateAllocatedFunds()` to `internal` as there's no need for external user to call it.

Alleviation

`[Bloxxy team]`: We advise the client to update the visibility of updateAllocatedFunds() to internal as there's no need for external user to call it.

VBC-04 | Logical Flaw In `getPendingAccumulatedFunds()`

Category	Severity	Location	Status
Logical Issue	Medium	projects/Bloxxy/contracts/Vault.sol (eaca569): 358~362	Resolved

Description

In the function `getPendingAccumulatedFunds()`, the `totalPeriodsCompleted` is used to calculated `totalBlocks` when `totalPeriodsCompleted` is greater than 0. However, this calculation is not accurate and the remainder of the division in L358 will be ignored, which in turn that the `totalBlocks` value is always less than or equal to the real block number appended to the blockchain.

```
358 uint256 totalPeriodsCompleted = (block.timestamp - (periodEndTime)) /
359     increaseRateAfterPeriods;
360
361 if (totalPeriodsCompleted > 0) {
362     totalBlocks = (totalPeriodsCompleted * 1 weeks) / blockTime;
```

For example, if `(block.timestamp – (periodEndTime)) / increaseRateAfterPeriods` equals to `2.5` arithmetically, the value that is assigned to `totalPeriodsCompleted` would be just 2, which turns out the value of `totalBlocks` is 2 * 1 weeks / blockTime, which is inaccurate.

Recommendation

We advise the client to revise the function and improve the calculation implementation.

Alleviation

`[Bloxxy team]`: We have addressed the risk of VBC-04 | Logical Flaw in the following ways:

- We have removed the times tamp based calculation of blocks and are using the block.number for more precision.
- The updated getPendingAccumulatedFunds() method calculates the accumulatedTokens for receiver according to the currentReleaseRatePerBlock of the respective period.
- The getPendingAccumulatedFunds method calculates the accumulated funds considering the following conditions :
- One period is completed, and the time passed after period completion is greater than the one period duration:
    - One period is completed and a certain number of blocks are then passed in current period.

- Total blocks passed in current perSome number of blocks are passed in current period.
- This updated method ensures accumulated funds are calculated up to the current block.

In this logic, if totalPeriodsCompleted gets the fractional value like 2.5, then the do-while loop will run only for 2 times, and it will calculate the accumulated funds according to the rates of the respective period.

This logic ensures that the remaining blocks of current period (up to the current block) are also considered for calculating the accumulatedFunds.

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.

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.