Code Security Assessment

Ambire

Feb 3rd, 2022

Table of Contents

Summary

This report has been prepared for Ambire to discover issues and vulnerabilities in the source code of the Ambire 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	Ambire
Platform	ethereum
Language	Solidity
Codebase	https://github.com/AmbireTech/wallet/
Commit	09c5da5f5b5572092289b3c1cf8371b62ad87cee

Audit Summary

Delivery Date	Feb 03, 2022
Audit Methodology	Static Analysis, Manual Review

Vulnerability Summary

Vulnerability Level	Total	Pending	Declined	Acknowledged	Partially Resolved	Mitigated	Resolved
Critical	0	0	0	0	0	0	0
Major	1	0	0	0	0	1	0
Medium	1	0	0	0	0	0	1
Minor	3	0	0	0	0	0	3
Informational	1	0	0	0	0	0	1
Discussion	0	0	0	0	0	0	0

Audit Scope

ID	File	SHA256 Checksum
WAE	WALLET.sol	0fc8dc2c61493795ede7e57154e0b997c434ae07d2bd03fb3617015e91e65f0c

Findings

Critical	**0** (0.00%)
Major	**1** (16.67%)
Medium	**1** (16.67%)
Minor	**3** (50.00%)
Informational	**1** (16.67%)
Discussion	**0** (0.00%)

ID	Title	Category	Severity	Status
WAE-01	Missing Emit Events	Coding Style	Informational	Resolved
WAE-02	Unused `internal` Function	Volatile Code	Minor	Resolved
WAE-03	Incorrect Inequality	Logical Issue, Mathematical Operations	Minor	Resolved
WAE-04	Unclear Use of `enum.Mint`	Gas Optimization, Inconsistency	Minor	Resolved
WAE-05	Centralization Risk in WALLET.sol	**Centralization / Privilege**	**Major**	Mitigated
WAE-06	Potential Change In `SupplyController` Address	Control Flow	Medium	Resolved

WAE-01 | Missing Emit Events

Category	Severity	Location	Status
Coding Style	Informational	WALLET.sol: 68~71	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. We recommend adding an event that notifies users when the `supplyController` address changes.

Alleviation

[Certik] - The `Ambire` team have resolved the issue by following our recommendation. The changes can be seen in the following commit,

https://github.com/AmbireTech/wallet/commit/1d2451df2396488ba99b587372adc4111d2e20c2

WAE-02 | Unused `internal` Function

Category	Severity	Location	Status
Volatile Code	Minor	WALLET.sol: 110~114	Resolved

Description

The lined function is an `internal` function that is never called. In other words there is no use of the function.

Recommendation

We advise the client to review the functionality of the function `innerMint()` within the `WALLETSupplyController` contract and remove it if unnecessary.

Alleviation

[Certik] - The `Ambire` team included extra functionality into the `Wallet.sol` contract that calls the linked `internal` function in this finding. These changes do not introduce any other vulnerabilities hence the issue is resolved. The changes can be seen in this commit

https://github.com/AmbireTech/wallet/commit/8b150b77d1c2e717955a3e367938e7abe5eba34a#diff-bacc0336483daf107a214580c08ff4409be2d01c5712087774c24e73ee870931

WAE-03 | Incorrect Inequality

Category	Severity	Location	Status
Logical Issue, Mathematical Operations	Minor	WALLET.sol: 103~104	Resolved

Description

The comment on line 103 states that an address should not receive an incentive of more than 10 `WALLET` tokens. However, the following statement actually restricts an address from receiving more than 9 `WALLET` tokens.

```
require(amountPerSecond < 10e18, "AMOUNT_TOO_LARGE");
```

Recommendation

We recommend replacing the line above with the following line:

```
require(amountPerSecond <= 10e18, "AMOUNT_TOO_LARGE");
```

Remark For clarity, our recommendation would allow an incentive of 10 `WALLET` tokens but no more than 10 while the original code only allowed an incentive of strictly less than 10 `WALLET` tokens.

Alleviation

[Certik] - The `Ambire` team has resolved the issue by including the line written in the recommendation. The changes can be seen in the commit below

https://github.com/AmbireTech/wallet/commit/db93ef70df183745829387f7609b0087126d91b3#diff-bacc0336483daf107a214580c08ff4409be2d01c5712087774c24e73ee870931

WAE-04 | Unclear Use Of `enum.Mint`

Category	Severity	Location	Status
Gas Optimization, Inconsistency	Minor	WALLET.sol: 75	Resolved

Description

In this contract, address are assigned a governance role. Certain functions can be called depending on a users governance role. The role `Mint` is one of the governance roles defined however there are no privileges for that role.

Recommendation

We advise the client to review the source code for a need of the `Mint` role. If there is no need, we recommend to remove the `Mint` role from enum declaration on line 75.

Alleviation

[Certik] - Previously a user can be set a governance level of None, Mint, or All. The Ambire team removed the `enum` list and included a mapping that either states if an address has governing power. So in other words, either a user has governing power or it doesn't. The changes made by the `Ambire` team does resolve the issue and can be seen in this commit,

https://github.com/AmbireTech/wallet/commit/1421114d63b4611a96f7b88d96c4d57702c10a26#diff-bacc0336483daf107a214580c08ff4409be2d01c5712087774c24e73ee870931

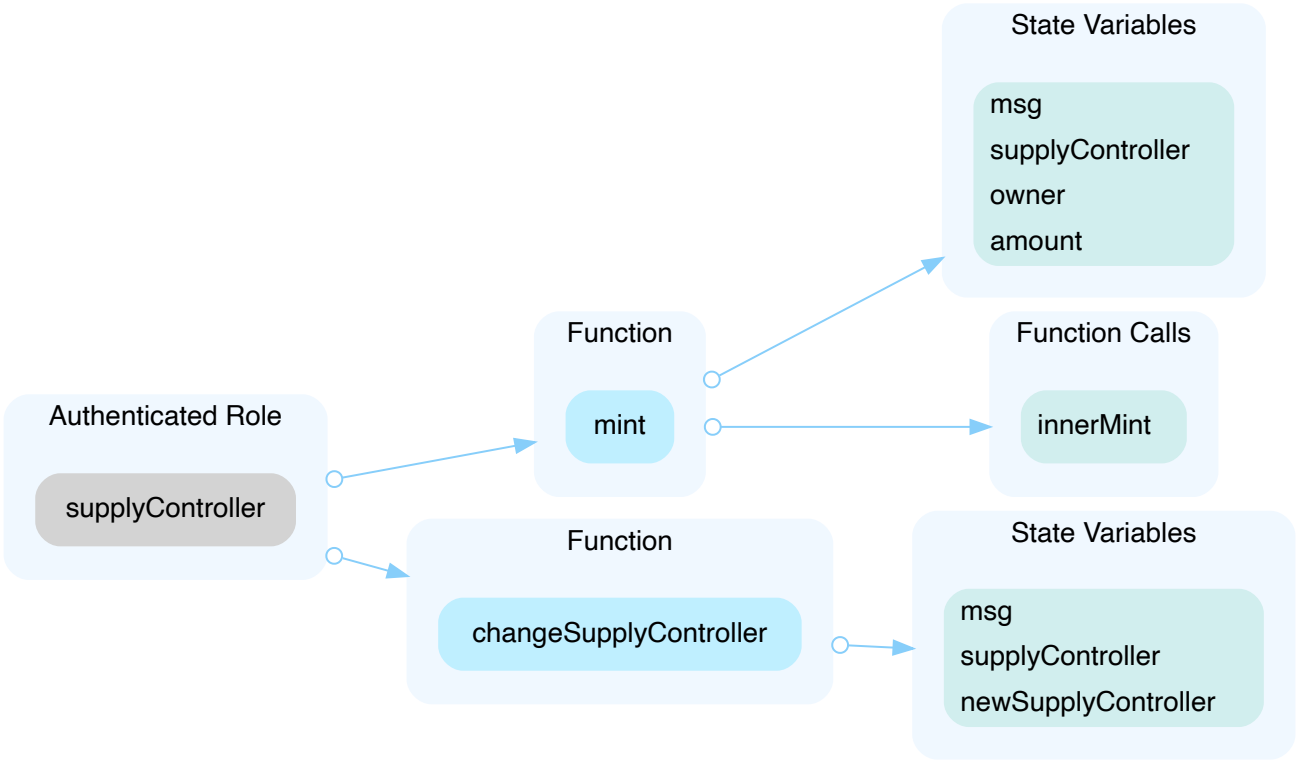
WAE-05 | Centralization Risk In WALLET.sol

Category	Severity	Location	Status
Centralization / Privilege	Major	WALLET.sol: 63~66, 68~71	Mitigated

Description

In the contract, `WALLETToken`, the role, `supplyController`, has authority over the functions shown in the diagram below.

Any compromise to the `supplyController` account may allow the hacker to take advantage of this authority and mint as many tokens to any address they wish.



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

[Ambire Team] - This is intended behavior, and `supplycontroller` will be controlled by a classic gnosis(pre-safe) multi-signature wallet. Later on, a time-lock will be added, either as a separate contract between supplycontroller and the multi-sig, or directly implemented in the `supplycontroller`

[Certik] - A multi-sig wallet signature wallet including and a time-lock contract would greatly reduce the risk of this finding. However, since the multi-signature wallet address have not been provided to us we cannot update the status of this finding accordingly.

[Ambire Team] - To further mitigate this we implemented a supply cap in the token https://github.com/AmbireTech/wallet/commit/8b0f6ca9be34bd92e146c8f18091402fedcb8a7c and we are providing a multi-sig wallet: <0x23c2c34f38ce66ccc10e71e9bb2a06532d52c5e9>

[Ambire Team]: The token is getting launched on Ethereum

The address is: 0x88800092ff476844f74dc2fc427974bbee2794ae

0x23c2c34f38ce66ccc10e71e9bb2a06532d52c5e9 is a gnosis (not gnosis safe) multisig on Ethereum as you can see here https://etherscan.io/address/0x23c2c34f38ce66ccc10e71e9bb2a06532d52c5e9#code. It's only presence on BSC is because of an airdrop apparently influenced by the Ethereum assets on the same address.

The deployment plan is as follows:

- deploy the token with an Ambire account as a supply controller (done)
- change the supply controller to the multisig
- change the supply controller to the final version of the contract, and the multisig will have governance rights to this

WAE-06 | Potential Change In `SupplyController` Address

Category	Severity	Location	Status
Control Flow	Medium	WALLET.sol: 63, 68, 91, 96	Resolved

Description

The deployer of the contract has their governance level set to `ALL`. That privilege allows the deployer of the contract to set the governance of any user to any level they want. We now describe the potential risk with this privilege. Suppose the deployer of the contract sets the governance level of Oscar to `ALL`. Then Oscar can set his address as the `supplyController` address and mint himself as many tokens as he likes.

Recommendation

Consider setting a bound to how many tokens can be minted and consider setting a time limit to how often tokens can be minted.

Alleviation

[Ambire Team] - It is an intended functionality, as we want to allow SupplyController to be upgradeable for two reasons

1. adding staking incentives to it, similarly to how the ADX supllyController has staking incentives: https://etherscan.io/address/0x9d47f1c6ba4d66d8aa5e19226191a8968bc9094e upgrade it to a version that **does not** allow changing the supplyController anymore, thereby removing most of the centralization risk

Alternatively, we can introduce a cap in `Wallet.sol` itself

[Certik] - This risk is related `WAL-05` and would be mitigated if the gnosis multi-signature wallet is implemented.

[Ambire Team] - we are introducing a supply cap https://github.com/AmbireTech/wallet/commit/8b0f6ca9be34bd92e146c8f18091402fedcb8a7c#diff-bacc0336483daf107a214580c08ff4409be2d01c5712087774c24e73ee870931

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.

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.

Mathematical Operations

Mathematical Operation findings relate to mishandling of math formulas, such as overflows, incorrect operations etc.

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.

Control Flow

Control Flow findings concern the access control imposed on functions, such as owner-only functions being invoke-able by anyone under certain circumstances.

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

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.