CERTIK

Security Assessment

Feminist Coin

Jul 4th, 2022

Table of Contents

Summary

This report has been prepared for Feminist Coin to discover issues and vulnerabilities in the source code of the Feminist Coin 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	Feminist Coin
Platform	BSC
Language	Solidity
Codebase	https://bscscan.com/address/0x8b12759cDBa5649eD067775b9026DbAd34B4Ba50

Audit Summary

Delivery Date	Jul 04, 2022 UTC
Audit Methodology	Static Analysis, Manual Review

Vulnerability Summary

Vulnerability Level	Total	Pending	Declined	Acknowledged	Mitigated	Partially Resolved	Resolved
Critical	0	0	0	0	0	0	0
Major	2	0	0	1	0	0	1
Medium	0	0	0	0	0	0	0
Minor	1	0	0	1	0	0	0
Optimization	2	0	0	2	0	0	0
Informational	3	0	0	3	0	0	0
Discussion	0	0	0	0	0	0	0

Audit Scope

ID	File	SHA256 Checksum
OWN	@openzeppelin/contracts/access/Ownable.sol	75e3c97011e75627ffb36f4a2799a4e887e1a3e27ed427490e82d7b6f51cc5c9
IER	@openzeppelin/contracts/token/ERC20/extensions/IERC20Metadata.sol	af5c8a77965cc82c33b7ff844deb9826166689e55dc037a7f2f790d057811990
ERE	@openzeppelin/contracts/token/ERC20/ERC20.sol	3cd9bf87ad804088f574a5266771f038a2c44b53d85f355aadb35645e497d1c2
IEC	@openzeppelin/contracts/token/ERC20/IERC20.sol	94f23e4af51a18c2269b355b8c7cf4db8003d075c9c541019eb8dcf4122864d5
COT	@openzeppelin/contracts/utils/Context.sol	1458c260d010a08e4c20a4a517882259a23a4baa0b5bd9add9fb6d6a1549814a
FTC	contracts/FmToken.sol	664fa500cac4fb31fb5c99c8252b93caa7f7fb040bc8bcaca51ad6b37736a202

Findings

Critical	**0** (0.00%)
Major	**2** (33.33%)
Medium	**0** (0.00%)
Minor	**1** (16.67%)
Informational	**3** (50.00%)
Discussion	**0** (0.00%)

ID	Title	Category	Severity	Status
CON-03	Redundant Code Components	Volatile Code	Informational	Acknowledged
ERE-01	Potential Risk On `approve()` / `transferFrom()` Methods	Volatile Code	Minor	Acknowledged
FTC-01	Initial Token Distribution	**Centralization / Privilege**	**Major**	Acknowledged
FTC-02	Declaration Naming Convention	Coding Style	Informational	Acknowledged
OWN-01	Centralization Risks In Ownable.sol	**Centralization / Privilege**	**Major**	Resolved
OWN-02	Missing Emit Events	Coding Style	Informational	Acknowledged

CON-03 | Redundant Code Components

Category	Severity	Location	Status
Volatile Code	Informational	@openzeppelin/contracts/token/ERC20/ERC20.sol: 280; @openzeppelin/contracts/utils/Context.sol: 21	Acknowledged

Description

The linked statements do not affect the functionality of the codebase and appear to be either leftovers from test code or older functionality.

Recommendation

We advise to remove the redundant statements for production environments.

ERE-01 | Potential Risk On `approve()` / `transferFrom()` Methods

Category	Severity	Location	Status
Volatile Code	Minor	@openzeppelin/contracts/token/ERC20/ERC20.sol: 312	Acknowledged

Description

Function _**approve** [*link*] from current project triggered expert rules derived from function _**approve** [*link*] from project **Monsters Adventures**.

Here is the description text of a finding from that project that might be relevant.

Current `MonsterAdventuresToken` implementation is vulnerable to a known ERC20 race condition issue, which could lead to token theft. When a user calls `approve()` for a second time on a spender that has already been allowed, the spender could call `transferFrom()` to transfer the previous value and still receive the authorization to transfer the new value.

Exploit scenario:

1. Alice calls `approve(Bob, 100)` to allow Bob to spend 100 tokens.
2. Alice changes her mind and calls `approve(Bob, 50)`.
3. Bob observes the second `approve(Bob, 50)` function call and calls `transferFrom(Alice, Bob, 100)` before the second `approve(Bob, 50)` call.
4. The above scenario can be achieved by front-running. In this case, Bob can transfer another 50 tokens from Alice and in total, he transferred 150 tokens from Alice.

Recommendation

We would advise to use OpenZeppelin ERC20 implementation as it includes `increaseApproval` and `decreaseApproval`. These functions only change the allowance by a certain value, instead of setting the new one. It is commonly used protection against FrontRunning of ERC20's approve issue.

FTC-01 | Initial Token Distribution

Category	Severity	Location	Status
Centralization / Privilege	**Major**	contracts/FmToken.sol: 14~17	Acknowledged

Description

All of the `Feminist` tokens are sent to the contract deployer when deploying the contract. This could be a centralization risk as the deployer can distribute `Feminist` 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.

FTC-02 | Declaration Naming Convention

Category	Severity	Location	Status
Coding Style	Informational	contracts/FmToken.sol: 8, 9, 10, 11	Acknowledged

Description

One or more declarations do not conform to the Solidity style guide with regards to its naming convention.

Particularly:

- `camelCase`: Should be applied to function names, argument names, local and state variable names, modifiers
- `UPPER_CASE`: Should be applied to `constant` variables
- `CapWords`: Should be applied to contract names, struct names, event names and enums

```
10      address public constant feministAgency =
0x87840DcBB724F3b1E714beAb8Db173b61d630C26;
```

- Constant variable `feministAgency` is not in `UPPER_CASE`.

```
11      address public constant dao = 0xDF3c793fD50175527F192B8D82B21165D3572beE;
```

- Constant variable `dao` is not in `UPPER_CASE`.

```
8       address public constant teamer = 0x05417a7B3755D77F7003368476853f1a8C532EC1;
```

- Constant variable `teamer` is not in `UPPER_CASE`.

```
9       address public constant investor = 0x5A2448504b4D83aeCa6510B59279311C2a8d51eD;
```

- Constant variable `investor` is not in `UPPER_CASE`.

Recommendation

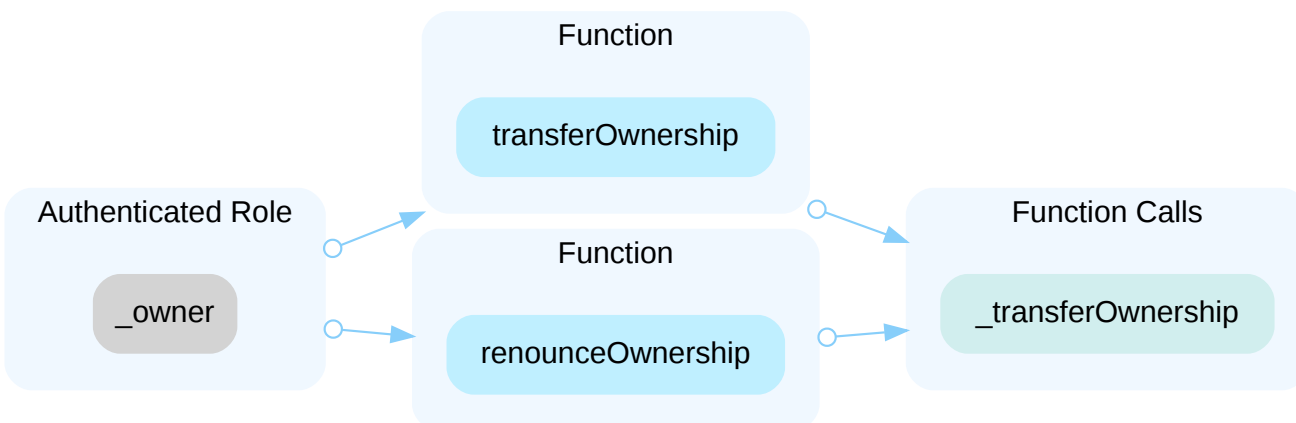
We recommend adjusting those variable and function names to properly conform to Solidity's naming convention.

OWN-01 | Centralization Risks In Ownable.sol

Category	Severity	Location	Status
Centralization / Privilege	**Major**	@openzeppelin/contracts/access/Ownable.sol: 54, 62	Resolved

Description

In the contract `Ownable` the role `_owner` has authority over the functions **transferOwnership** and **renounceOwnership** shown in the diagram below. Any compromise to the `_owner` account may allow the hacker to take advantage of this authority and transfer the ownership of the contract to any other desired address/account and also renounce ownership to leave the contract without an owner and remove any functionality that is only available to the owner.



Recommendation

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

Short Term:

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

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

Long Term:

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

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

Permanent:

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

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

Alleviation

`[CertiK]`: The team heeded the advice and resolved the finding by renouncing the `owner` address to `address(0)` in the deployment
https://bscscan.com/address/0x8b12759cDBa5649eD067775b9026DbAd34B4Ba50

OWN-02 | Missing Emit Events

Category	Severity	Location	Status
Coding Style	Informational	@openzeppelin/contracts/access/Ownable.sol: 54, 62	Acknowledged

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.

Optimizations

ID	Title	Category	Severity	Status
CON-01	Function Should Be Declared External	Gas Optimization	Optimization	Acknowledged
CON-02	Improper Usage Of `public` And `external` Type	Gas Optimization	Optimization	Acknowledged

CON-01 | Function Should Be Declared External

Category	Severity	Location	Status
Gas Optimization	Optimization	@openzeppelin/contracts/access/Ownable.sol: 54, 62; @openzeppelin/contracts/token/ERC20/ERC20.sol: 62, 70, 87, 94, 101, 113, 136, 158, 181, 201	Acknowledged

Description

The functions which are never called internally within the contract should have external visibility for gas optimization.

```
101     function balanceOf(address account) public view virtual override returns (uint256) {
```

```
113     function transfer(address to, uint256 amount) public virtual override returns (bool) {
```

```
136     function approve(address spender, uint256 amount) public virtual override returns (bool) {
```

```
158     function transferFrom(
```

```
181     function increaseAllowance(address spender, uint256 addedValue) public virtual returns (bool) {
```

```
201     function decreaseAllowance(address spender, uint256 subtractedValue) public virtual returns (bool) {
```

```
54     function renounceOwnership() public virtual onlyOwner {
```

```
62     function name() public view virtual override returns (string memory) {
```

```
62    function transferOwnership(address newOwner) public virtual onlyOwner {
```

```
70    function symbol() public view virtual override returns (string memory) {
```

```
87    function decimals() public view virtual override returns (uint8) {
```

```
94    function totalSupply() public view virtual override returns (uint256) {
```

Recommendation

We advise to change the visibility of the aforementioned functions to `external`.

CON-02 | Improper Usage Of `public` And `external` Type

Category	Severity	Location	Status
Gas Optimization	Optimization	@openzeppelin/contracts/access/Ownable.sol: 62; @openzeppelin/contracts/token/ERC20/ERC20.sol: 101, 113, 136, 158, 181, 201	Acknowledged

Description

`public` functions that are never called by the contract could be declared as `external`. `external` functions are more efficient than `public` functions.

Recommendation

Consider using the external attribute for public functions that are never called within the contract.

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.

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.