CERTIK

Security Assessment

Metaracers

Dec 9th, 2021

Table of Contents

Summary

This report has been prepared for Metaracers to discover issues and vulnerabilities in the source code of the Metaracers 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.

Additionally, this audit is based on a premise that all external contracts were implemented safety.

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	Metaracers
Platform	bsc
Language	Solidity
Codebase	https://github.com/MetaRacers/MRS/commit/6b66a9e9c8b6b3fcd5c0557c8fb262f9543f91fd
Commit	6b66a9e9c8b6b3fcd5c0557c8fb262f9543f91fd

Audit Summary

Delivery Date	Dec 09, 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	2	0	0
● Medium	0	0	0	0	0	0
● Minor	1	0	0	0	0	1
● Informational	8	0	0	0	0	8
● Discussion	0	0	0	0	0	0

Audit Scope

ID	File	SHA256 Checksum
MRS	MRS.sol	a1b8cb0825e241d11cb5738fee85c9fb3dff6f1776ec6844e836d1de29d4df14

Understandings

Overview

The `Metaracers` Protocol is an ERC20 token deployed on the Binance smart chain.

There is no transaction fee. Those who are in blacklist can not send or receive tokens. Those who are in whitelist can exclude from `antiWhale`. The owner can update the blacklist and whitelist. When `antiWhale` is enabled by owner, transaction amount is limited and only one sell transaction is allowed in a period time set by owner.

The owner can withdraw ERC20 tokens of contract address in case of emergency.

Privileged Functions

The contract contains the following privileged functions that are restricted by some modifiers. They are used to modify the contract configurations and address attributes. We grouped these functions below:

The `onlyOwner` modifier:

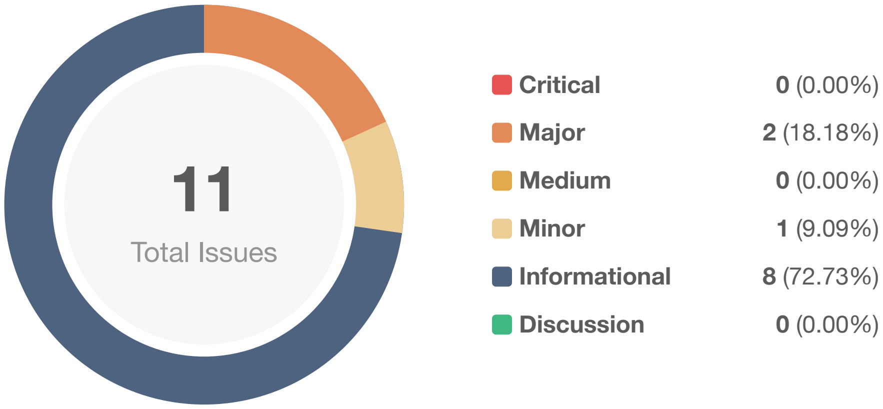
Contract `Ownable`:

- renounceOwnership()
- transferOwnership()

Contract `MRS`:

- addWhitelist()
- multiBlacklist()
- multiRemoveFromBlacklist()
- setAntiWhale()
- setMaxSell()
- setAntiWhaleEnd()
- rescueStuckErc20()

Findings



ID	Title	Category	Severity	Status
GLOBAL-01	Centralization Risk	**Centralization / Privilege**	● **Major**	ⓘ Acknowledged
GLOBAL-02	Missing Emit Events	Coding Style	● Informational	✓ Resolved
MRS-01	Unlocked Compiler Version	Language Specific	● Informational	✓ Resolved
MRS-02	Too Many Digits	Coding Style	● Informational	✓ Resolved
MRS-03	Useless Variable	Language Specific, Gas Optimization	● Informational	✓ Resolved
MRS-04	Token Minted To Centralized Address	**Centralization / Privilege**	● **Major**	ⓘ Acknowledged
MRS-05	Ambiguous Function Name	Coding Style	● Informational	✓ Resolved
MRS-06	Lack of Input Validation	Volatile Code	● Minor	✓ Resolved
MRS-07	Missing Error Messages	Coding Style	● Informational	✓ Resolved
MRS-08	Unreachable `else-clause` in Function `antiWhale()`	Coding Style	● Informational	✓ Resolved
MRS-09	Function `burnFrom()` Available for Everyone	Inconsistency	● Informational	✓ Resolved

GLOBAL-01 | Centralization Risk

Category	Severity	Location	Status
Centralization / Privilege	● Major	Global	ⓘ Acknowledged

Description

In the contract `Ownable`, the role `owner` has the authority over the following function:

- renounceOwnership()
- transferOwnership()

In the contract `MRS`, the role `owner` has the authority over the following function:

- addWhitelist()
- multiBlacklist()
- multiRemoveFromBlacklist()
- setAntiWhale()
- setMaxSell()
- setAntiWhaleEnd()
- rescueStuckErc20()

Any compromise to the `owner` account may allow the hacker to take advantage of this.

Recommendation

We advise the client to carefully manage the `owner` 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

The team acknowledged.

GLOBAL-02 | Missing Emit Events

Category	Severity	Location	Status
Coding Style	● Informational	Global	✓ Resolved

Description

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

- setAntiWhale()
- setMaxSell()
- setAntiWhaleEnd()

Recommendation

Consider adding events for sensitive actions, and emit them in the function.

Alleviation

The team heeded our advice and changed related codes. Code change was applied in commit 5f2e1a008d8c6e445de26886a59b19a0102d23f8.

MRS-01 | Unlocked Compiler Version

Category	Severity	Location	Status
Language Specific	● Informational	MRS.sol: 2	✓ Resolved

Description

The contract has unlocked compiler version. An unlocked compiler version in the source code of the contract permits the user to compile it at or above a particular version. This, in turn, leads to differences in the generated bytecode between compilations due to differing compiler version numbers. This can lead to an ambiguity when debugging as compiler specific bugs may occur in the codebase that would be hard to identify over a span of multiple compiler versions rather than a specific one.

Recommendation

We advise that the compiler version is instead locked at the lowest version possible that the contract can be compiled at. For example, for version `v0.8.2` the contract should contain the following line:

```
pragma solidity 0.8.2;
```

Alleviation

The team heeded our advice and changed related codes. Code change was applied in commit 5f2e1a008d8c6e445de26886a59b19a0102d23f8.

MRS-02 | Too Many Digits

Category	Severity	Location	Status
Coding Style	● Informational	MRS.sol: 1023, 1028	✓ Resolved

Description

Literals with many digits are difficult to read and review.

Recommendation

We recommend modifying as below:

```
1023  uint256 private _totalSupply = 3 * 10**8 * 10**uint256(_decimals);
```

```
1028  uint256 public maxSell = 3000 * 10**uint256(_decimals);
```

Alleviation

The team heeded our advice and changed related codes. Code change was applied in commit 5f2e1a008d8c6e445de26886a59b19a0102d23f8.

MRS-03 | Useless Variable

Category	Severity	Location	Status
Language Specific, Gas Optimization	● Informational	MRS.sol: 1024	✓ Resolved

Description

The variable `_tFeeTotal` is never used in this contract.

Recommendation

We advise to remove the useless variables.

Alleviation

The team heeded our advice and changed related codes. Code change was applied in commit 5f2e1a008d8c6e445de26886a59b19a0102d23f8.

MRS-04 | Token Minted To Centralized Address

Category	Severity	Location	Status
Centralization / Privilege	● **Major**	MRS.sol: 1040	ⓘ Acknowledged

Description

The amount of `_totalSupply` tokens that are minted to the centralized address `msg.sender` who is `owner`, may raise the community's concerns about the centralization issue.

Recommendation

We advise the client to carefully manage the `owner` account's private key and avoid any potential risks of being hacked. We also advise the client to adopt Multisig, Timelock, and/or DAO in the project to manage this specific account in this case.

Alleviation

The team acknowledged.

MRS-05 | Ambiguous Function Name

Category	Severity	Location	Status
Coding Style	● Informational	MRS.sol: 1044	✓ Resolved

Description

The function name `addWhitelist` is ambiguous, since it also can remove accounts from whitelist.

Recommendation

We recommend changing the name `addWhitelist` to `setWhitelist`.

Alleviation

The team heeded our advice and changed related codes. Code change was applied in commit 5f2e1a008d8c6e445de26886a59b19a0102d23f8.

MRS-06 | Lack of Input Validation

Category	Severity	Location	Status
Volatile Code	● Minor	MRS.sol: 1064~1070	✓ Resolved

Description

The length of array `receivers[]` and `amounts[]` should be the same. Function `multiTransfer()` misses the validation.

Recommendation

We recommend adding the validation of array's length.

Alleviation

The team heeded our advice and changed related codes. Code change was applied in commit 5f2e1a008d8c6e445de26886a59b19a0102d23f8.

MRS-07 | Missing Error Messages

Category	Severity	Location	Status
Coding Style	● Informational	MRS.sol: 1173, 1190~1191	✓ Resolved

Description

The **require** can be used to check for conditions and throw an exception if the condition is not met. It is better to provide a string message containing details about the error that will be passed back to the caller.

Recommendation

We advise providing a string message containing details about the error.

Alleviation

The team heeded our advice and changed related codes. Code change was applied in commit 5f2e1a008d8c6e445de26886a59b19a0102d23f8.

MRS-08 | Unreachable `else-clause` in Function `antiWhale()`

Category	Severity	Location	Status
Coding Style	● Informational	MRS.sol: 1214~1240	✓ Resolved

Description

The `traders[_sender]["SELL"].lastTrade` is either 0 or bigger than 0. As a result, the last `else-clause` at Line 1233~1238 is unreachable.

```
1221  if (_amount > maxSell) {
1222    revert("Anti whale sell");
1223  }
1224  else if (traders[_sender]["SELL"].lastTrade == 0) {
1225      traders[_sender]["SELL"] = TraderInfo({
1226          lastTrade: curTime,
1227          amount: _amount
1228      });
1229  } else if (
1230      traders[_sender]["SELL"].lastTrade  > 0
1231  ) {
1232      revert("Wait for next trade");
1233  } else {
1234      traders[_sender]["SELL"] = TraderInfo({
1235          lastTrade: curTime,
1236          amount: _amount
1237      });
1238  }
```

Recommendation

We recommend changing as blew:

```
if (_amount > maxSell) {
  revert("Anti whale sell");
} else if (traders[_sender]["SELL"].lastTrade == 0) {
    traders[_sender]["SELL"] = TraderInfo({
        lastTrade: curTime,
        amount: _amount
    });
} else {
    revert("Wait for next trade");
}
```

Alleviation

The team heeded our advice and changed related codes. Code change was applied in commit 5f2e1a008d8c6e445de26886a59b19a0102d23f8.

MRS-09 | Function `burnFrom()` Available for Everyone

Category	Severity	Location	Status
Inconsistency	● Informational	MRS.sol: 498, 1185	✓ Resolved

Description

The function `burn()` in abstract contract `ERC20Burnable` is `override` in contract `MRS`, so that only owner can call the `burn()` successfully. It seems that you do not want users burn their tokens on their own. However, the public function `burnFrom()` is not `override` in contract `MRS` and is still available for everyone. This means users still can burn tokens from other accounts with approval. Please make sure whether you allow users to burn tokens or not.

Alleviation

The team removed all the related codes about `burn`. Code change was applied in commit 5f2e1a008d8c6e445de26886a59b19a0102d23f8.

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.

Language Specific

Language Specific findings are issues that would only arise within Solidity, i.e. incorrect usage of private or delete.

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.

CERTIK