



SAFETIN **AUDIT**

CRYPTOAI

January 31th, 2023



TABLE OF CONTENTS

- I. SUMMARY
- II. OVERVIEW
- III. FINDINGS
 - A. [UINT-1](#) : Unoptimized uint size
 - B. [MSG-1](#) : Unclear error message
 - C. [MSG-3](#) : Too long error message
 - D. [CENT-1](#) : Centralization of major privileges
 - E. [EXT-1](#) : External protocol dependance
- IV. GLOBAL SECURITY WARNINGS
- V. DISCLAIMER

AUDIT SUMMARY

This report was written for [CryptoAI](#) in order to find flaws and vulnerabilities in the [CryptoAI](#) project's source code, as well as any contract dependencies that weren't part of an officially recognized library.

A comprehensive examination has been performed, utilizing Static Analysis, Manual Review, and [CryptoAI](#) Deployment 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
- ❖ Through line-by-line manual review of the entire codebase by industry expert

AUDIT OVERVIEW

PROJECT SUMMARY

Project name	CryptoAI
Description	CryptoAI is an ERC20 token of the Ethereum blockchain. \$CAI is a key component that fuels the CryptoAI ecosystem. It will be primarily used for all the operations within the CryptoAI ecosystem.
Platform	Ethereum
Language	Solidity
Codebase	https://etherscan.io/token/0xf36c5f04127f7470834ed6f98bd4c1be62aba48d#code

FINDINGS SUMMARY

Vulnerability	Total	Resolved
● Critical	0	0
● Major	0	0
● Medium	2	0
● Minor	0	0
● Informational	2	0

EXECUTIVE SUMMARY

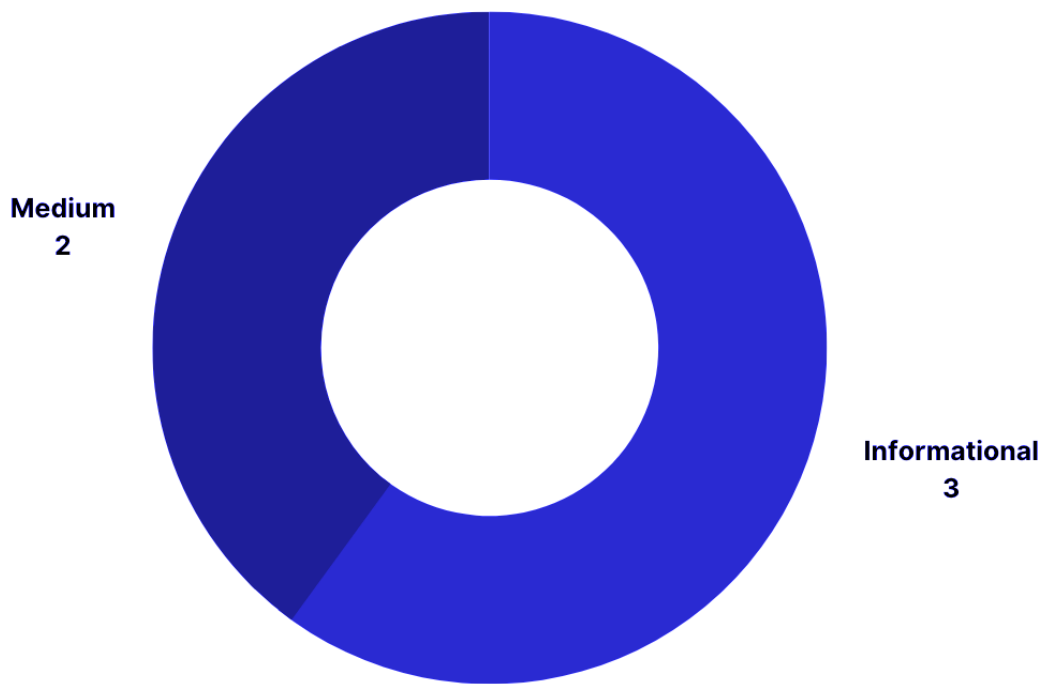
CryptoAI is an ERC20 token on Ethereum. It includes a trading mechanism, taxes (which we will detail later), max transaction and wallet, and an exclusion mechanism.

The taxes are distributed as follows (they are editable, they are the current values of the contract).

Transactions are taxed by 4% for marketing costs and 1% for the contest mechanism, which goes directly into the contest wallet.

There have been no major or critical issues related to the codebase and all findings listed here are minor or informational. The major security problem is the centralization of privileges and the dependance of external protocol.

AUDIT FINDINGS



Code	Title	Severity
CENT-1	Centralization of major privileges	● Medium
EXT-1	Dependence on an external protocol	● Medium
UINT-1	Unoptimized uint size	● Informational
MSG-1	Unclear error message	● Informational
MSG-3	Too long error message	● Informational

UINT-1 | Unoptimized uint size

Description

Some variables in the contract are of type uint, but not of the right size. In order to optimize gas costs when deploying and using the contract, we invite you to assign the right size uint to each variable. Especially since fees are under 100 they don't need to be set as uint256. 8 errors of this type have been found in the smart contract.

Recommendation

```
//L 679-687
uint8 buyTotalFees;
uint8 buyMarketingFee;
uint8 buyContestAIFee;
uint8 buyLiquidityFee;

uint8 sellTotalFees;
uint8 sellMarketingFee;
uint8 sellContestAIFee;
uint8 sellLiquidityFee;

//L 815
function updateFees(uint8 _marketingFeeBuy, uint8
 _liquidityFeeBuy, uint8 _contestAIFeeBuy, uint8
 _marketingFeeSell, uint8 _liquidityFeeSell, uint8
 _contestAIFeeSell) external onlyOwner {
```

MSG-1 | Unclear error message

Description

Some of the error messages in the smart contract are unclear or badly formulated. In order to optimize error handling and project maintenance, we recommend using the clearest possible error messages. 1 error of this type have been found in the smart contract.

Recommendation

We recommend shortening these error messages :

```
//Edited code containing shortened error message  
//line 842 :  
"The pair cannot be removed from marketPair" -> "Must  
keep uniswapV2Pair"
```

MSG-3 | Too long error message

Description

The smart contract has some error messages that are too long. The industry standards specify error messages must have a maximal length of 32 bytes. We recommend having the shortest possible error messages to optimize gas costs (see github.com/ethereum/solidity/issues/4588) and improve error handling. 4 issues of this type have been found in the smart contract.

Recommendation

We recommend shortening these error messages :

```
//Edited code containing shortened error message
//line 798 :
"Cannot set maxTransactionAmounts Lower than 1%" ->
"maxBuyAmount must be higher than 1%"
//line 799 :
"Cannot set maxTransactionAmounts Lower than 1%" ->
"maxSellAmount must be higher than 1%"

//line 877 :
"Buy transfer amount exceeds the maxTransactionAmount."
-> "buy transfer over max amount"
//line 880 :
"Sell transfer amount exceeds the maxTransactionAmount."
-> "Sell transfer over max amount"
```

CENT-1 | Centralization of major privileges

Description

The `onlyOwner` modifier of the smart contract gives major privileges over it (especially, owner can modify taxes and exclude addresses from fees and transaction limits). This can be a problem, in the case of a hack, an attacker who has taken possession of this privileged account could damage the project and the investors.

Recommendation

We recommend at least to use a multi-sig wallet as the owner address, and at best to establish a community governance protocol to avoid such centralization. For more information, see <https://solidity-by-example.org/app/multi-sig-wallet/>

EXT-1 | Dependence to an external protocol

Description

The contract interacts with [Uniswap](#) protocols. The scope of the audit would treat this third party entity as black box and assume it is fully functional. However in the real world, third parties may be compromised and may have led to assets lost or stolen. We fully understand that the business logic of the [CryptoAI](#) token is designed to work with [Uniswap](#).

Recommendation

We encourage the team to constantly monitor the security level of the entire [Uniswap](#) project, as the security of the token is highly dependent on the security of the project.

Global security warnings

These are safety issues for the whole project. They are not necessarily critical problems but they are inherent in the structure of the project itself. Potential attack vectors for these security problems should be monitored.

CENT-1 | Global SPOF (Single Point Of Failure)

The project's smart contracts often have a problem of centralized privileges. The [onlyOwner](#) system in particular can be subject to attack. To address this security issue we recommend using a multi-sig wallet, establishing secure project administration protocols and strengthening the security of project administrators.

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 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 without [Safetin's](#) prior written consent. 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 [Safetin](#) 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

Safetin security assessment 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. Safetin's position is that each company and individual are responsible for their own due diligence and continuous security. Safetin'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 fun.