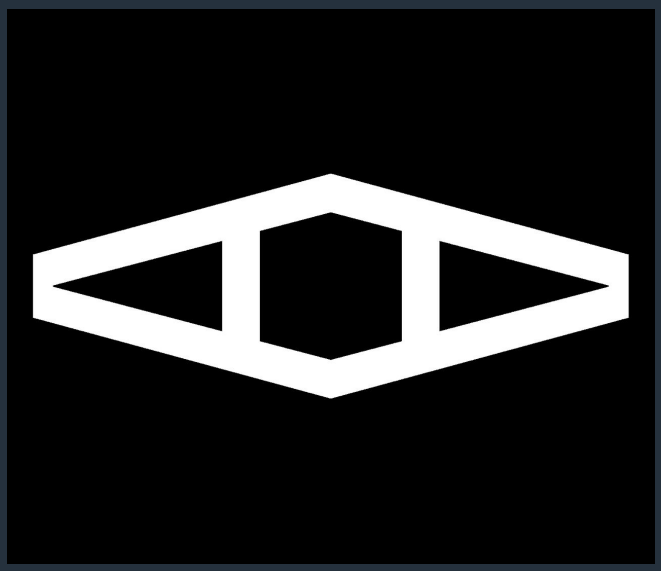


SMART CONTRACT AUDIT

ZOKYO.

June, 24th 2022 | v. 1.0

PASS

Zokyo's Security Team has concluded that this smart contract passes security qualifications to be listed on digital asset exchanges.

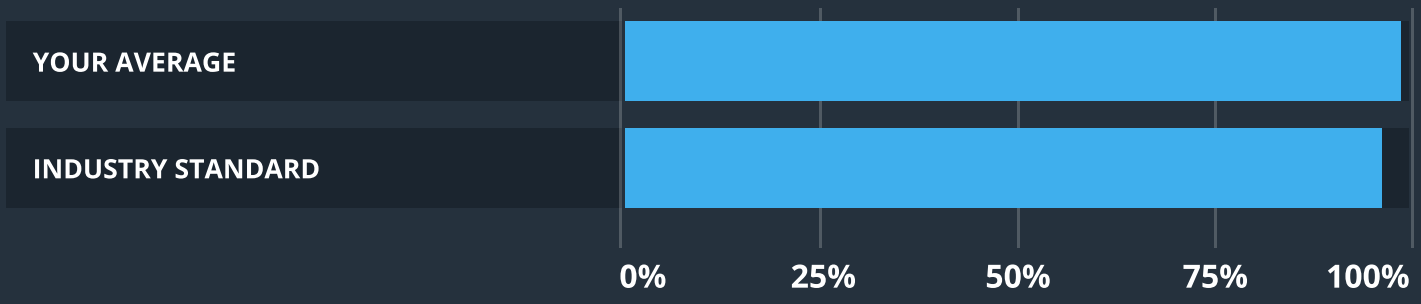
TECHNICAL SUMMARY

This document outlines the overall security of the ArtMeta smart contracts, evaluated by Zokyo's Blockchain Security team.

The scope of this audit was to analyze and document the ArtMeta smart contract codebase for quality, security, and correctness.

Contract Status

Testable Code



The testable code is 100%, which is above the industry standard of 95%.

It should be noted that this audit is not an endorsement of the reliability or effectiveness of the contract, rather limited to an assessment of the logic and implementation. In order to ensure a secure contract that's able to withstand the Ethereum network's fast-paced and rapidly changing environment, we at Zokyo recommend that the ArtMeta team put in place a bug bounty program to encourage further and active analysis of the smart contract.

TABLE OF CONTENTS

AUDITING STRATEGY AND TECHNIQUES APPLIED

The Smart contract's source code was taken from the ArtMeta deployed token:
https://polygonscan.com/address/0xdcff29b7bd211aaef6e4a3989e4d3f732cf5b4b6#code

Within the scope of this audit Zokyo auditors have reviewed the following contract(s):

- Token.sol

Throughout the review process, care was taken to ensure that the contract:

- Implements and adheres to existing standards appropriately and effectively;
- Documentation and code comments match logic and behavior;
- Distributes tokens in a manner that matches calculations;
- Follows best practices in efficient use of resources, without unnecessary waste;
- Uses methods safe from reentrance attacks;
- Is not affected by the latest vulnerabilities;
- Whether the code meets best practices in code readability, etc.

Zokyo's Security Team has followed best practices and industry-standard techniques to verify the implementation of ArtMeta smart contracts. To do so, the code is reviewed line-by-line by our smart contract developers, documenting any issues as they are discovered. Part of this work includes writing a unit test suite using the Truffle testing framework. In summary, our strategies consist largely of manual collaboration between multiple team members at each stage of the review:

1	Due diligence in assessing the overall code quality of the codebase.	3	Testing contract logic against common and uncommon attack vectors.
2	Cross-comparison with other, similar smart contracts by industry leaders.	4	Thorough, manual review of the codebase, line-by-line.

EXECUTIVE SUMMARY

Auditor's team has reviewed the ArtMeta token implementation of ERC20 standard.
Token uses standard OpenZeppelin implementation of ERC20 with no additional or suspicious functionality. Token's supply is 100M tokens which are minted during the deployment to the deployer address (0x04Bb64aBf19733607f6b6A90Cc584c2574D9cFe8).

Auditors team has confirmed the correctness of the implementation.

STRUCTURE AND ORGANIZATION OF DOCUMENT

For ease of navigation, sections are arranged from most critical to least critical. Issues are tagged “Resolved” or “Unresolved” depending on whether they have been fixed or addressed. Issues tagged “Verified” contain unclear or suspicious functionality that either needs explanation from the Customer’s side or it is an issue that the Customer disregards as an issue. Furthermore, the severity of each issue is written as assessed by the risk of exploitation or other unexpected or otherwise unsafe behavior:

Critical

The issue affects the contract in such a way that funds may be lost, allocated incorrectly, or otherwise result in a significant loss.

High

The issue affects the ability of the contract to compile or operate in a significant way.

Medium

The issue affects the ability of the contract to operate in a way that doesn’t significantly hinder its behavior.

Low

The issue has minimal impact on the contract’s ability to operate.

Informational

The issue has no impact on the contract’s ability to operate.

COMPLETE ANALYSIS

INFO | VERIFIED

ArtMeta token implements ERC20 standard

During the audit it was verified, that ArtMeta token implements ERC20 standard completely, has no additional functionality which can interfere with standard token business logic and has no internal loopholes, thus it is safe for the usage by users.

	Token.sol
Re-entrancy	Pass
Access Management Hierarchy	Pass
Arithmetic Over/Under Flows	Pass
Unexpected Ether	Pass
Delegatecall	Pass
Default Public Visibility	Pass
Hidden Malicious Code	Pass
Entropy Illusion (Lack of Randomness)	Pass
External Contract Referencing	Pass
Short Address/ Parameter Attack	Pass
Unchecked CALL Return Values	Pass
Race Conditions / Front Running	Pass
General Denial Of Service (DOS)	Pass
Uninitialized Storage Pointers	Pass
Floating Points and Precision	Pass
Tx.Origin Authentication	Pass
Signatures Replay	Pass
Pool Asset Security (backdoors in the underlying ERC-20)	Pass

CODE COVERAGE AND TEST RESULTS FOR ALL FILES

Tests written by Zokyo Secured team

As part of our work assisting ArtMeta in verifying the correctness of their contract code, our team was responsible for writing integration tests using the Truffle testing framework.

Tests were based on the functionality of the code, as well as a review of the ArtMeta contract requirements for details about issuance amounts and how the system handles these.

Contract: ArtMeta Token

- ✓ Initialized with correct total supply (50ms)
- ✓ Initialized with correct name (55ms)
- ✓ Initialized with correct symbol (52ms)
- ✓ Initialized with correct decimals
- ✓ Total supply was minted to correct address (38ms)

5 passing (1s)

FILE	% STMTS	% BRANCH	% FUNCS
Token.sol	100	100	100
All files	**100**	**100**	**100**

We are grateful to have been given the opportunity to work with the ArtMeta team.

The statements made in this document should not be interpreted as investment or legal advice, nor should its authors be held accountable for decisions made based on them.

Zokyo's Security Team recommends that the ArtMeta team put in place a bug bounty program to encourage further analysis of the smart contract by third parties.

ZOKYO.