Enigma Token Audit

OPENZEPPELIN SECURITY | OCTOBER 13, 2017

Security Audits

The Element Group team asked us to review and audit their Enigma Token contract. We looked at the code and now publish our results.

The audited contract is in the element-group/enigma-erc20-smart-contract repository. The version used for this report is the commit `9b6a6edab5eaf79242cc59d705f8b315657f87b7`.

Here's our assessment and recommendations, in order of importance.

***Update:** The Element Group team has followed most of our recommendations and updated the contracts. The new version is at commit `54e37d2d6b02579305eb0d9e9f8f795d5a0bd23f`.*

Low Severity

No Transfer for initial supply creation

Consider emitting an event `Transfer(0x0, msg.sender, INITIAL_SUPPLY)` in the constructor. This is consistent with the recommended behavior for token creation, and will improve the user experience. For example, the created token amount will be immediately displayed in the Etherscan token interface.

***Update:** Event added in `84412e9`.*

ERC20 compliance

Additionally, the version of OpenZeppelin used for Enigma (1.2.0) included a restriction to the usage of `approve` that has been deemed non-compliant and was removed in the next version. Consider upgrading to OpenZeppelin 1.3.0.

Update: *Fixed in* `a09d58c`.

Notes & Additional Information

- Additionally to the mentioned 1.3.0 changes, the modifications done in the Enigma token to `transfer`, `transferFrom`, and `approve` have all been included in this latest version. If OpenZeppelin were upgraded, these functions could safely be removed and fall back to OpenZeppelin's implementation.
  Update: *Updated to 1.3.0 in* `ac20729`.

Conclusion

No severe issues were found. Some changes were proposed to improve standards compliance, follow best practices and reduce potential attack surface.

Note that as of the date of publishing, the above review reflects the current understanding of known security patterns as they relate to the Enigma Token contract. We have not reviewed the related Enigma project. The above should not be construed as investment advice. For general information about smart contract security, check out our thoughts here.

Related Posts

Beefy
Zap Audit

BRUSHFAM
OpenBrush Contracts Library Security Review

Linea
Bridge Audit

Beefy Zap Audit

BeefyZapRouter serves as a versatile intermediary designed to execute users' orders through routes...

Security Audits

OpenBrush Contracts Library Security Review

OpenBrush is an open-source smart contract library written in the Rust programming language and the...

Security Audits

Linea Bridge Audit

Linea is a ZK-rollup deployed on top of Ethereum. It is designed to be EVM-compatible and aims to...

Security Audits

OpenZeppelin

Defender Platform

- Secure Code & Audit
- Secure Deploy
- Threat Monitoring
- Incident Response
- Operation and Automation

Services

- Smart Contract Security Audit
- Incident Response
- Zero Knowledge Proof Practice

Learn

- Docs
- Ethernaut CTF
- Blog

Company

- About us
- Jobs
- Blog

Contracts Library

Docs

Privacy | Terms of Use