Audit & Bug Bounty Programs

Security of the platform is our highest priority. All contract code and balances are publicly verifiable, and security researchers are eligible for a bug bounty for reporting undiscovered vulnerabilities.

Audits

OpenZeppelin has performed the following audits on UMA contracts:

- Common and oracle directory contracts: April 28, 2020
- Financial-templates directory contracts: May 12, 2020
- Updates to the Expiring Multiparty contracts and flash loan mitigations for the voting contracts: September 9, 2020
- Perpetual Multiparty template contracts: February 2, 2021
- Insured Bridge contracts: December 1, 2021
- Governance, cross-chain oracle, and optimistic rewarder contracts: January 7, 2022
- UMA Optimistic Governor Audit: July 21, 2022
- Across Token and Token Distributor Audit: July 21, 2022
-

Additionally, OpenZeppelin audits incremental upgrades to UMA's contracts on a continuous basis. The continuous audit report can be foundhere .

Bug Bounty Rewards

UMA encourages the community to audit our contracts and security; we also encourage the responsible disclosure of any issues. The bug bounty program is intended to recognize the value of working with the community of independent security researchers and sets out our definition of good faith in the context of finding and reporting vulnerabilities, as well as what you can expect from us in return.

UMA offers substantial rewards for discoveries that can prevent the loss of assets, the freezing of assets, or harm to users.

All rewards will be paid in UMA, and the amount of compensation will vary depending on bug severity. Reward amounts typically correspond to severity in the following manner.

Severity Reward amount in USD Low 250 Medium 3,000 High 10,000 Critical up to 1,000,000 Severity is calculated according to theOWASP risk rating model based on Impact and Likelihood.

?

Scope

The scope of our bug bounty program includes any and all of UMA's production smart contracts. It does not include known issues with the intended behavior.

In scope

- All UMA, Oval, or Across smart contracts that are deployed to mainnet or are otherwise noted as being applicable.
- Bot or other offchain code to support deployed smart contracts.
-

Examples of what’s in scope:

- Being able to steal funds
- Being able to freeze funds or render them inaccessible by their owners
-

Out of scope:

- Issues that have already been submitted by another user or are already known to the UMA team
-
    - Note: this includes bugs known to the UMA team, but have not been disclosed due to active mitigation efforts.
- *
- Vulnerabilities in contracts built on top of the protocol by third-party developers (such as smart contract wallets)
- Vulnerabilities that require ownership of an admin key
- Any files, modules or libraries other than the ones mentioned above
- More efficient gas solutions (although these suggestions are appreciated)
- Any points listed as an already known weaknesses
- Any points listed in an audit report
-

Submissions

Please email your submissions tobugs@umaproject.org .

The submission must include clear and concise steps to reproduce the discovered vulnerability.

Terms & Conditions

If you comply with the policies below when reporting a security issue to us, we will not initiate a lawsuit or law enforcement investigation against you in response to your report.

We ask that you:

- Report any vulnerability you've discovered promptly.
- Avoid violating the privacy of others, disrupting our systems, destroying data, or harming user experience.
- Use onlybugs@umaproject.org
- to discuss vulnerabilities with us.
- Keep the details of any discovered vulnerabilities confidential until they are publicly announced by Risk Labs.
- Perform testing only on in-scope systems, and respect systems and activities which are out-of-scope.
- Not engage in blackmail, extortion, or any other unlawful conduct.
- Not be a current or former UMA Foundation employee, vendor, contractor, or the employee of an UMA vendor or contractor.
-

Public disclosure of the bug or the indication of an intention to exploit it on Mainnet will make the report ineligible for a bounty. If in doubt about other aspects of the bounty, most of theEthereum Foundation bug bounty program rules will apply.

Any questions? Reach us via email (bugs@umaproject.org). For more information on the UMA platform, check out ourwebsite andGithub .

All reward determinations, including eligibility and payment amount, are made at UMA's sole discretion. UMA reserves the right to reject submissions and alter the terms and conditions of this program without notice.

Previous New Network Requests Next Approved Price Identifiers Last updated1 month ago On this page *Audits * Bug Bounty Rewards * Scope * Submissions * Terms & Conditions

Was this helpful? Edit on GitHub