ENIGMA DARK

Securing the Shadows

Security Review

Aave 3.2 Upgrade

September, 2024

Contents

Summary

Enigma Dark
Enigma Dark is a web3 security firm leveraging the best talent in the space to secure all kinds of blockchain protocols and decentralized apps. Our team comprises experts who have honed their skills at some of the best auditing companies in the industry. With a proven track record as highly skilled white-hats, they bring a wealth of experience and a deep understanding of the technology and the ecosystem.

Learn more about us at enigmadark.com

BGS Labs
BGD Labs is a Web3 development initiative and one of the main Aave contributors, responsible for the protocol's ongoing maintenance and the introduction of new, innovative features. Comprised of a team of expert developers with deep experience in Ethereum, more precisely in Decentralized Finance (DeFi), governance token modeling/economics.

AAVE 3.2
AAVE is one of the leading lending an borrowing platforms in DeFi, allowing users to supply and borrow assets with variable interest rates. The Aave 3.2 upgrade removes the previously disabled stable debt from the codebase and introduces liquid eModes. This upgrade is designed to reduce dead / unused code and improve the user experience, making Aave's eMode functionality more granular, flexible, and efficient.

For a comprehensive overview of the upgrade, please refer to BGS Labs' official proposal on the AAVE governance forum here.

Engagement Overview

Over the course of 2 weeks starting September 2nd 2024, the Enigma Dark team conducted a security review of the AAVE 3.2 upgrade including stable debt removal and liquid eMode. The review was performed by two Lead Security Researchers, vnmrtz.eth & Lambda.

The following repositories were reviewed at the specified commits:

Repository	Commit
AAVE v3 Origin	`a4849111a0ce57e3af1ca5cd9a9b8c6a8cdad1e0` & `dd0bbecb90a53628fe15c076217eac3a7275182f`

Risk Classification

Severity	Description
Critical	Vulnerabilities that lead to a loss of a significant portion of funds of the system.
High	Exploitable, causing loss or manipulation of assets or data.
Medium	Risk of future exploits that may or may not impact the smart contract execution.
Low	Minor code errors that may or may not impact the smart contract execution.

Vulnerability Summary

Severity	Count	Fixed	Acknowledged
Critical	0	0	0
High	0	0	0
Medium	0	0	0
Low	0	0	0
Informational	5	3	2

Findings

Index	Issue Title	Status
I-01	Remaining variables referencing stable rate	Fixed
I-02	Early return on `executeSetUserEMode` affects revert behaviour	Acknowledged
I-03	Constant `KEEP_CURRENT_ADDRESS` not longer used	Acknowledged
I-04	Miscellaneous	Fixed
I-05	Typos	Fixed

Detailed Findings

High Risk

No issues found.

Medium Risk

No issues found.

Low Risk

No issues found.

Informational

I-01 - Remaining variables referencing stable rate

Technical Details: There are a few remaining variables which reference the stable rate and could be safely removed:

- src/contracts/helpers/interfaces/IUiIncentiveDataProviderV3.sol: AggregatedReserveIncentiveData.sIncentiveData
- src/contracts/helpers/interfaces/IUiIncentiveDataProviderV3.sol: UserReserveIncentiveData.sTokenIncentivesUserData

- src/contracts/helpers/interfaces/IUiPoolDataProviderV3.sol: AggregatedReserveData.stableDebtTokenAddress

Developer Response: Fixed at commit dbdb739.

I-02 - Early return on `executeSetUserEMode` affects revert behaviour

Technical Details: After a user enters an eMode and borrows an asset, if that asset later becomes non-borrowable within that specific eMode, the previous version of the function (without the early return) would revert when attempting to set the same eMode again, as the validation checks would fail.

However, with the new early return mechanism, the function no longer reverts, giving the impression that the eMode can still be activated, even though it shouldn't be. This change in revert behaviour should be documented to prevent any inconsistencies or confusion for integrators.

Recommendation: Consider performing validations first or documenting the early revert on no-op behaviour.

Developer Response: Acknowledged. If some smart contract logic if by mistake tries to enter a state it already is in, it's valid to skip everything as the user already is in the desired end state - no matter if the current rules would or would not allow entering.

I-03 - Constant `KEEP_CURRENT_ADDRESS` not longer used

Technical Details: Since the eMode changes, the constant KEEP_CURRENT_ADDRESS is no longer used and could in theory be removed.

Developer Response: Acknowledged, we think we would keep as it doesn't hurt and might be reused.

I-04 - Miscellaneous

Technical Details:

- Very small typo was introduced in src/contracts/interfaces/IPool.sol:295, should be "variable debt" instead of "variabledebt".
- src/contracts/extensions/v3-config-engine/IAaveV3ConfigEngine.sol:326: Wrong comment, this should be EModeBorrowableUpdate instead of EModeCollateralUpdate.
- src/contracts/protocol/libraries/logic/ConfiguratorLogic.sol:43 NatSpec still references stable debt token.

Developer Response: Fixed 1) & 3) at commit 4fcf2d6. On 2) the config engine was updated quite a bit after audit started so this no longer applies.

I-05 - Typos

Technical Details: Wrong comment was introduced in src/contracts/extensions/v3-config-engine/IAaveV3ConfigEngine.sol:295 and IAaveV3ConfigEngine.sol:190, should reference "AssetEModeUpdate" instead of "EModeCollateralUpdate".

Developer Response: Fixed at commit a1240df

Disclaimer

This report does not endorse or critique any specific project or team. It does not assess the economic value or viability of any product or asset developed by parties engaging Enigma Dark for security assessments. We do not provide warranties regarding the bug-free nature of analyzed technology or make judgments on its business model, proprietors, or legal compliance.

This report is not intended for investment decisions or project participation guidance. Enigma Dark aims to improve code quality and mitigate risks associated with blockchain technology and cryptographic tokens through rigorous assessments.

Blockchain technology and cryptographic assets inherently involve significant risks. Each entity is responsible for conducting their own due diligence and maintaining security measures. Our assessments aim to reduce vulnerabilities but do not guarantee the security or functionality of the technologies analyzed.

This security engagement does not guarantee against a hack. It is a review of the codebase at a during a specific period of time. Enigma Dark makes no warranties regarding the security of the code and does not warrant that the code is free from defects. By deploying or using the code, the project and users of the contracts agree to use the code at their own risk. Any modifications to the code will require a new security review.