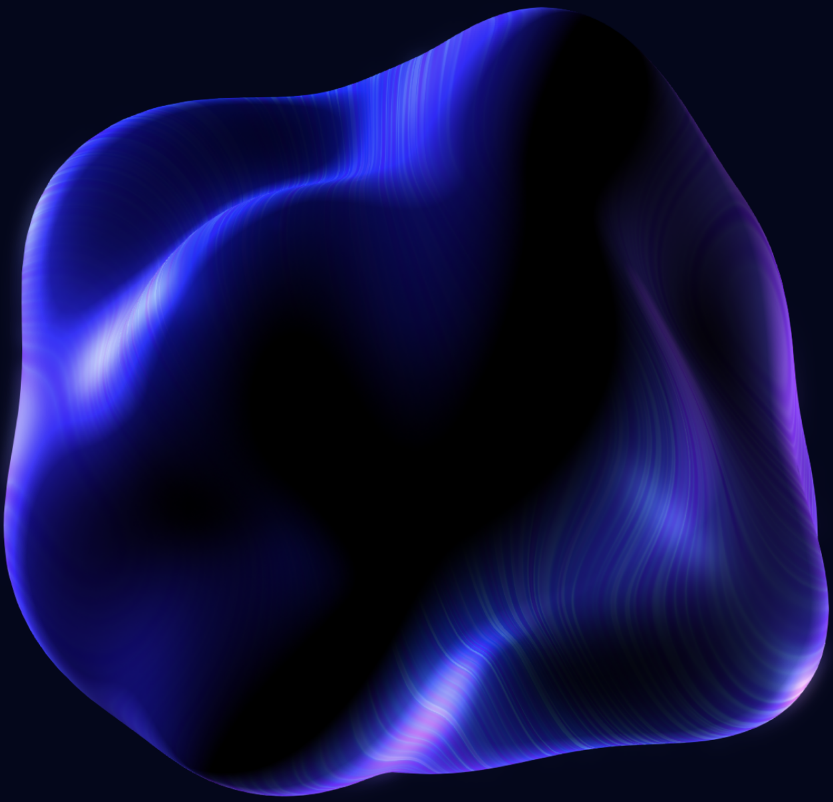
MUNDUS SECURITY

Zerolend Deployment Check on zkSync Era



mundus.dev

@mundus_security

Mundus.dev

MUNDUS
SECURITY

ZeroLend deployment check

This document may contain confidential information about IT systems and the intellectual property of the Customer as well as information about potential vulnerabilities and methods of their exploitation.
The report containing confidential information can be used internally by the Customer, or it can be disclosed publicly after all vulnerabilities are fixed — upon a decision of the Customer.

Project overview

ZeroLend is a lending protocol operating on zkSync Era chain. ZeroLend integrates lending logic by Aave with price oracles by Pyth Network.

Reference information

Name	ZeroLend
Language	Solidity
Chain	zkSync Era
Website	https://zerolend.xyz/
Documentation	https://docs.zerolend.xyz/
Reference repositories	https://github.com/zerolend/core-contracts https://github.com/zerolend/periphery-contracts https://github.com/zerolend/pyth-oracles https://github.com/zerolend/governance https://github.com/zerolend/onez-core

Deployment check summary for Zerolend

Codebase inconsistency issues

Type	Severity
In deployed contracts	**NONE**
Smart contracts vs Git	**NONE**
In dependencies of codebase	**NONE**

Storage analysis

Type	# Issues
Found Total	**29**
Found Critical	**0**
Left as acknowledged after re-checks	**0**
Left as Critical after re-checks	**0**

Deployment check is expert review of the storage and codebase consistency of a deployed project*

Deployed Smart contracts and/or Git repos & Contract storage

Deployment check is especially important for projects with active development and regular updates to ensure that after all incremental updates, the set of contracts and their settings are consistent.

Deployment check protects against

- Errors in CI/CD, especially in large projects
- Potential attacks from people with access to the codebase
- Large number of human errors during updates of the project in the network
- Incorrect cross-references between smart contracts
- Relation to old versions of contracts
- Forgotten role members
- Uncorrected ownership

*Note that security audit is not the part of this review

Scope of work

#	contract	address	verified
1	AaveOracle	0x785765De3E9ac3D8eEb42B4724A7FEA8990142B8	True
2	ACLManager	0x9A60cce3da06d246b492931d2943A8F574e67389	True
3	AToken	0xe8178fF950Ea1B69a51cE961C542a4CC6Cb6e38E	True
4	DelegationAwareAToken	0x102699803F4A2b02046C38C672401759af633510	True
5	EmissionManager	0x72D2aB433526d32e6Ee52c03d1562A9E79bf0F19	True
6	IncentivesV2-Implementation	0x86bd524C09508df7B4B9027464975351B1BC2c92	True
7	IncentivesProxy	0x54AB34aB3C723bD2674c7082aA6fFcdfd3A5BEdc	True
8	Pool-Implementation	0x54d6F91bE4509826559ad12E1Ca6CA3A6C3811e0	True
9	Pool-Proxy	0x4d9429246EA989C9CeE203B43F6d1C7D83e3B8F8	True
10	PoolConfigurator-Implementation	0x8FBC873afD2a23D0bDd79d8a8756a38adda40810	True
11	PoolConfigurator-Proxy	0x9C3058F7bfCA6139ac3013999F57D7aa6a3AB1Ed	True
12	PoolDataProvider	0xB73550bC1393207960A385fC8b34790e5133175E	True
13	ReservesSetupHelper	0xe00d794744e763BeC67BdEdF6e852D4e0d958DFb	True
14	ReserveStrategy-rateStrategy StableOne	0x70cA80C5dE9fC8f080a494453dF1aA9180073031	True
15	ReserveStrategy-rateStrategy StableTwo	0xcaA502e289bFb924732f44f5E70bd08fc052aab8	True
16	ReserveStrategy-rateStrategy VolatileOne	0xEdAc06D73DbdD3460B5728E4bBE9862b04Ac198a	True
17	StableDebtToken	0x3A8ea541597D74ACB33F94533D731940AF516031	True
18	UiIncentive DataProviderV3	0x91ccF57c1E9A7F5A9537eE59306faF8dA3b7e960	True
19	UiPoolDataProviderV3	0x8FE0ac76b634B7D343Bd32282B98E9f271B43367	True
20	VariableDebtToken	0xA48aCc9847Cc1dD2caDA05151C9A78Ba47a305Cb	True

NOTE: Verification process for contracts marked with ⓘ is described below.

#	contract	address	verified
21	WalletBalanceProvider	0xdeEa10da04D867e3303AB6E50FA26C2d8a5e9f70	True
22	WrappedTokenGatewayV3	0x767b4A087c11d7581Ac95eaFfc1FeBFA26bad3d2	True
23	PoolAddressesProvider	0x4f285Ea117eF0067B59853D6d16a5dE8088bA259	True
24	USDC-USD	0x75D018f04f9cb37936530F7e3A909474565A2467	True
25	WETH-USD	0x517F9cd13fE63e698d0466ad854cDba5592eeA73	True
26	USDT-USD	0xCf58E8e67F2BcDd977e61bB6FDC1B0EEd6E1939d	True
27	BorrowLogic	0x81D6b98Beb0A4288dCFab724FDeaE52E5Aa2F7b1	True
28	BridgeLogic	0x6CDe8a8cEE9771A30dE4fEAB8eaccb58cb0d30aF	True
29	ConfiguratorLogic	0x8731d4E5b990025143609F4A40eC80Fb482E46A0	True
30	EModeLogic	0xD84E953a621bb9D81Dc998E0b1482D2916153c23	True
31	FlashLoanLogic	0x424C0995114a614c12506D9A994d3eE140742f12	True
32	LiquidationLogic	0x8855Fd7d577A05d04Cea2E026c5BAa4Bb47feAf9	True
33	PoolAddresses ProviderRegistry	0x78B93fBb35C97b32C7381C81Fa3A620b3fB7787B	False ⓘ
34	PoolLogic	0xA8D16FB0620E3376093cb89e2cD9dEF9fE47Daaa	True
35	SupplyLogic	0x9223dC9205Cf8336CA59bA0bD390647E62D487E5	True
36	Treasury-Controller	0x677C3Cae4F23142c6A8480694554751B462d7326	False ⓘ
37	Treasury-Implementation	0xC59971Ff27806629D9935fbFBBFC2236961f82C8	False ⓘ
38	TreasuryProxy	0xE52540DBD350c611A1B9c51E97e2A6bc16c09133	False ⓘ
39	WETH-AToken	0x9002ecb8a06060e3b56669c6B8F18E1c3b119914	False ⓘ
40	WETH-StableDebtToken	0x9c9158BFF47342A20b7D2Ac09F89e96F3A209b9B	False ⓘ
41	WETH-VariableDebtToken	0x56f58d9BE10929CdA709c4134eF7343D73B080Cf	False ⓘ
42	USDC-AToken	0x016341e6Da8da66b33Fd32189328c102f32Da7CC	False ⓘ
43	USDC-StableDebtToken	0x5faC4FD2e4bCE392d34600d94Aa1114274e54Dff	False ⓘ
44	USDC-VariableDebtToken	0xE60E1953aF56Db378184997cab20731d17c65004	False ⓘ

NOTE: Verification process for contracts marked with ⓘ is described below.

#	contract	address	verified
45	USDT-AToken	0x9ca4806fa54984Bf5dA4E280b7AA8bB821D21505	False ⓘ
46	USDT-StableDebtToken	0x6F977fD05962d67Eb7B16b15684fbEa0462F442d	False ⓘ
47	USDT-VariableDebtToken	0xa333c6FF89525939271E796FbDe2a2D9A970F831	False ⓘ
48	Timelock	0x861cC6724D0aA7Ec7a868887643e682b1c16aeeC	True
49	Multisig	0x7b08d0d9D6f450243500338C39B1c9F01a30d801	True
50	GhoToken	0x90059C32Eeeb1A2aa1351a58860d98855f3655aD	True
51	GhoOracle	0x1E3720185512d22C7352759e79dC3515d752AA50	True
52	cbETH-USD	0x3D5BcB12800A092FC85Ca00837594146F274C273	True
53	BUSD-USD	0x1a963D0C6bF364C1C8AE4F17b6aB773c627cEFB7	True
54	PEPE-USD	0xCd16A63d1960Afe718c4FE62D0b8D8f19Fc29618	True
55	WBTC-USD	0xe99FFA17f20F3f8022862d1BD13519D305eF1377	True
56	uniV2LP-USD	0x071Bf614bc2c50140c1f094346774e529571A9Fb	True
57	TransferStrategy	0xc0fcea0b31c79f70b5453a9c70e361fcaccb43a2	True
58	PEPE-AToken	0x54330D2333AdBF715eB449AAd38153378601cf67	False ⓘ
59	LUSD-AToken	0xd97Ac0ce99329EE19b97d03E099eB42D7Aa19ddB	False ⓘ
60	WBTC-AToken	0x7c65E6eC6fECeb333092e6FE69672a3475C591fB	False ⓘ
61	BUSD-AToken	0xb727F8e11bc417c90D4DcaF82EdA06cf590533B5	False ⓘ
62	GhoAToken Implementation	0x0e1d2c6284144d60dda047c982d5389c5db052c5	True
63	ONEZ-AToken	0x52846A8D972ABbF49F67d83d5509aa4129257F46	False ⓘ
64	GhoVariableDebtToken Implementation	0x6e8667e11bfefe57560f4b29f4d32440d856612b	True
65	ONEZ-VariableDebtToken	0x77dcEd4833E3a91437Ed9891117BD5a61C2AD520	False ⓘ
66	BUSD-VariableDebtToken	0x3E1F1812c2a4f356d1b4FB5Ff7cca5B2ac653b94	False ⓘ
67	GhoDiscountRateStrategy	0x8c58628c4a67906cc09d33f65d34775c1ad9d19a	False ⓘ

Table of contents

Findings summary

Storage findings - 1

#	contract	storage issues initial check	storage issues re-check
1	AaveOracle	found	none
2	ACLManager	found	none
3	AToken	none	--
4	DelegationAwareAToken	none	--
5	EmissionManager	found	none
6	IncentivesV2-Implementation	none	--
7	IncentivesProxy	found	none
12	PoolDataProvider	none	--
13	ReservesSetupHelper	found	none
14	ReserveStrategy-rateStrategyStableOne	found	none
15	ReserveStrategy-rateStrategyStableTwo	found	none
16	ReserveStrategy-rateStrategyVolatileOne	found	none
17	StableDebtToken	none	--
18	UiIncentiveDataProviderV3	none	--
19	UiPoolDataProviderV3	none	--
20	VariableDebtToken	none	--
21	WalletBalanceProvider	none	--
22	WrappedTokenGatewayV3	found	none
23	PoolAddressesProvider	found	none

Storage findings - 2

#	contract	storage issues initial check	storage issues re-check
24	USDC-USD	found	none
25	WETH-USD	found	none
26	USDT-USD	found	none
27	BorrowLogic	none	--
28	BridgeLogic	none	--
30	EModeLogic	none	--
32	LiquidationLogic	none	--
33	PoolAddresses ProviderRegistry	found	none
34	PoolLogic	none	--
35	SupplyLogic	none	--
36	Treasury-Controller	found	none
37	Treasury-Implementation	none	--
38	TreasuryProxy	none	--
39	WETH-AToken	none	--
40	WETH-StableDebtToken	none	--
41	WETH-VariableDebtToken	none	--
42	USDC-AToken	none	--
43	USDC-StableDebtToken	none	--
44	USDC-VariableDebtToken	none	--
45	USDT-AToken	none	--
46	USDT-StableDebtToken	none	--
47	USDT-VariableDebtToken	none	--
48	Timelock	found	none
49	Multisig	found	none

Storage findings - 3

#	contract	storage issues initial check	storage issues re-check
50	GhoToken	found	none
51	GhoOracle	none	--
52	cbETH-USD	none	--
53	BUSD-USD	none	--
54	PEPE-USD	none	--
55	WBTC-USD	none	--
56	uniV2LP-USD	none	--
57	TransferStrategy	found	none
58	PEPE-AToken	none	--
59	LUSD-AToken	none	--
60	WBTC-AToken	none	--
61	BUSD-AToken	none	--
62	GhoAToken Implementation	none	--
63	ONEZ-AToken	none	--
64	GhoVariableDebtToken Implementation	none	--
65	ONEZ-VariableDebtToken	found	none
66	BUSD-VariableDebtToken	none	--
67	GhoDiscountRateStrategy	none	--

Source code findings

The Mundus team has found no issues concerning the consistency of the code base among **verified** contracts.

The forked repositories do not contain any changes to Aave codebase that impose concerns to the security of the protocol.

Addressing unverified contracts

Scope of work contains 21 contracts which are not verified on zkSync Era explorer. The ZeroLend team stated that all verification attempts of these contracts were unsuccessful due to intrinsic problems within the zkSync Era explorer caused by its rapidly evolving nature. The client also stated that redeployment of these contracts was not an option as they already possessed user data. In order to proceed with deployment check of unverified contracts, we have to determine the structures of their storage. This can be done via comparison of deployed contracts' bytecodes to the locally compiled bytecodes.

The table below presents the comparative analysis of deployed and compiled bytecodes for each unverified contract. All of the deployed contracts' bytecode lengths match the lengths of bytecodes compiled from files under the reference_compilation_path column. Furthermore, for each of these contracts, the position of the first discrepancy is less than 32 bytes away from the end of the bytecode, and the difference between the deployed and compiled bytecodes **does not exceed 32 bytes**. These 32 bytes are the bytecode hash which is appended to the end of each contract's bytecode. Since for each contract, the bytecode discrepancy lies within the bytecode hash (last 32 bytes) and the rest of the bytecode matches the compiled version, we conclude that this set of 21 unverified contracts is what it is claimed to be by the ZeroLend team.

The discrepancies in bytecode hashes are likely caused by incorrect links to the external libraries, which have to be set during compilation. Currently, an external library's address is embedded into a contract's bytecode whether or not a contract actually uses this library (see zkSync Era doc for additional info). As described further in the report, neither of the 21 unverified contracts uses any external libraries. Thus, we can assume that the discrepancies in bytecode hashes do not pose a threat to the safety of the protocol.

id	name	lengths equal	length	first diff	diff length	reference compilation path
33	PoolAddresses ProviderRegistry	True	10784	10753	32	PoolAddresses ProviderRegistry.sol
36	Treasury-Controller	True	12064	12033	32	AaveEcosystem ReserveController.sol
37	Treasury-Implementation	True	38624	38593	32	AaveEcosystem ReserveV2.sol

id	name	lengths equal	length	first diff	diff length	reference compilation path
38	TreasuryProxy	True	16096	16065	32	InitializableAdminUpgradeabilityProxy.sol
39	WETH-AToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
40	WETH-StableDebtToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
41	WETH-VariableDebtToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
42	USDC-AToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
43	USDC-StableDebtToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
44	USDC-VariableDebtToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
45	USDT-AToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
46	USDT-StableDebtToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
47	USDT-VariableDebtToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
58	PEPE-AToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol

id	name	lengths equal	length	first diff	diff length	reference compilation path
59	LUSD-AToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
60	WBTC-AToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
61	BUSD-AToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
63	ONEZ-AToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
65	ONEZ-VariableDebtToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
66	BUSD-VariableDebtToken	True	14624	14593	32	InitializableImmutableAdminUpgradeabilityProxy.sol
67	GhoDiscountRateStrategy	True	2144	2145	0	GhoDiscountRateStrategy.sol

Summary

The contents of the contracts in SoW which are unverified by the zkSync Era explorer are identified and safe to use.

Deployment check: source code

This analysis aims to identify any differences or inconsistencies in the source code of the smart contracts. We perform the analysis in three steps:

1. Analyzing for inconsistency between source code files across deployed smart contracts (excluding well-known dependencies such as OpenZeppelin or Uniswap).
2. Looking for the original commit in the client's repository, which represents all source code of deployed smart contracts in the case of providing the client's git
3. Analyzing the dependencies of the contracts

External libraries

The table below lists all external libraries and the contracts that use them. This information is required for identification of incorrect links within the protocol.

library	Pool	PoolConfigurator	FlashLoanLogic	LiquidationLogic
BorrowLogic	X		X	
BridgeLogic	X			
ConfiguratorLogic		X		
EModeLogic	X			X
FlashLoanLogic	X			
LiquidationLogic	X			
PoolLogic	X			
SupplyLogic	X			

Inconsistency between the same project files across contracts (excluding dependencies)

The goal is to check for any differences and inconsistencies in the source code of the same parts of the contracts. We compare each pair of smart contracts in the scope of work (SoW). Files with the same name and relative path included (imported) in both contracts should have the same content.

Summary

The team has found no inconsistencies among **verified** contracts' files.

Searching for the original commit in the client's repository

At this stage, we are looking for the original commit in the client's repository. In the best case, all contracts should be deployed from a single codebase revision to decrease the probability of inconsistency in the contract logic.

core-contracts

contracts	commit	# contracts

AaveOracle
ACLManager
AToken
BorrowLogic
BridgeLogic
ConfiguratorLogic
DelegationAwareAToken
EModeLogic
FlashLoanLogic
IncentivesProxy
LiquidationLogic
PoolAddressesProvider
PoolConfigurator-Implementation
PoolConfigurator-Proxy
PoolDataProvider
Pool-Implementation
PoolLogic
Pool-Proxy
ReserveStrategy-rateStrategyStableOne
ReserveStrategy-rateStrategyStableTwo
ReserveStrategy-rateStrategyVolatileOne
StableDebtToken
SupplyLogic
VariableDebtToken
	latest (2023-08-20T04:47:51+03:00):
	b2a43babe1609e1eb3db219a1789840d6c5802e8
	
	earliest (2023-08-20T04:47:51+03:00):
	b2a43babe1609e1eb3db219a1789840d6c5802e8
		24

contracts	commit	# contracts

ReservesSetupHelper	latest (2023-08-20T04:44:13+03:00):
	b676cc335154b88ec0ae0fd42ef63607e8f8edb5
	earliest (2023-07-15T01:36:18+05:30):
	2448f46b6b472ba0f83a615f68aa8614866a8321
		1

The **ReservesSetupHelper** contract's files do not fall under the same commit with the rest of the core-contracts due to formatting changes in @aave/core-v3/contracts/protocol/pool/PoolConfigurator.sol, which do not introduce any inconsistencies to the logic of the protocol.

periphery-contracts

contracts	commit	# contracts

EmissionManager
IncentivesV2-Implementation
TransferStrategy
UiIncentiveDataProviderV3
UiPoolDataProviderV3
WalletBalanceProvider
WrappedTokenGatewayV3
	latest (2023-07-15T03:13:10+05:30):
	d785e0de52395b7789e0aea9c8a2a14919333af8
	earliest (2023-07-15T03:04:28+05:30):
	841be584a2bae05851da73e3b0984a1c3a804fa9
		7

governance

contracts	commit	# contracts
Timelock **Multisig**	single commit (2023-08-17T03:50:51+03:00): 5dcfce6428e0abcca06efd5f1b075cd0dcf62308	2

The code of the **Multisig** contract corresponds to MultiSigWallet.sol from the gnosis MultiSigWallet repository, commit 90639984c960d281bed3e0a5d56dd4adcb9407c4.

onez-core

contracts	commit	# contracts
GhoAToken-Implementation **GhoOracle** **GhoToken** **GhoVariableDebtToken-Implementation**	latest (2023-08-09T14:27:23+03:00): 9c32834a4620310cc59105d247b0a57bf5e96768 earliest (2023-08-09T14:27:23+03:00): 9c32834a4620310cc59105d247b0a57bf5e96768	4

pyth-oracles

contracts	commit	# contracts
BUSD-USD **uniV2LP-USD** **WBTC-USD**	latest (2023-07-20T00:43:09+05:30): 08c187ac2faf4ee366cf8f4c3ba4ddd60c7ee6cf earliest (2023-07-17T02:11:39+05:30): 08c187ac2faf4ee366cf8f4c3ba4ddd60c7ee6cf	3

contracts	commit	# contracts
USDC-USD **WETH-USD** **USDT-USD**	latest (2023-07-20T00:43:09+05:30): f00726842c0006106739b7da8011367329c9db79 earliest (2023-07-17T02:11:39+05:30): 806d83aa0171dba957652cac521c738289c3441c	3

The **cbETH-USD** and **PEPE-USD** contracts' files do not fall under any commit in the pyth-oracles repository due to uncommitted changes in pyth-oracles/contracts/PythAggregatorV3.sol, which handle ERC20 decimals and, as stated by the client, do not introduce any inconsistencies to the logic of the protocol.

Summary

All **verified** contracts have consistent codebase.

Analyzing the dependencies of the contracts

The goal is to check the consistency of every dependency version and identify any changes across every dependency codebase.

Periphery contracts

contract	@zerolendxyz/core-v3
EmissionManager	b2a43babe1609e1eb3db219a1789840d6c5802e8
IncentivesV2-Implementation	b2a43babe1609e1eb3db219a1789840d6c5802e8
TransferStrategy	b2a43babe1609e1eb3db219a1789840d6c5802e8
UiIncentiveDataProviderV3	b2a43babe1609e1eb3db219a1789840d6c5802e8
UiPoolDataProviderV3	b2a43babe1609e1eb3db219a1789840d6c5802e8
WalletBalanceProvider	b2a43babe1609e1eb3db219a1789840d6c5802e8
WrappedTokenGatewayV3	b2a43babe1609e1eb3db219a1789840d6c5802e8

Governance

contract	@openzeppelin/contracts
Timelock	v4.9.3
Multisig	

ONEZ core

contract	@zerolendxyz/core-v3	@openzeppelin/contracts
GhoAToken-Implementation	b2a43babe1609e1eb3db219a1789840d6c5802e8	v4.8.3
GhoOracle		
GhoToken		v4.8.3
GhoVariableDebtToken-Implementation	b2a43babe1609e1eb3db219a1789840d6c5802e8	

Pyth oracles

contract	@pythnetwork	@openzeppelin/ contracts
USDC-USD	v2.2.0	
WETH-USD	v2.2.0	
USDT-USD	v2.2.0	
BUSD-USD	v2.2.0	
univ2LP-USD	v2.2.0	v4.9.3
WBTC-USD	v2.2.0	
cbETH-USD	v2.2.0	
PEPE-USD	v2.2.0	

Summary

All **verified** contracts use consistent versions of respective dependencies.

Forked code analysis

The following table presents the reference repositories and their corresponding parent repositories.

reference repository	parent repository	pertinent changes
core-contracts	aave-v3-core@v1.19.1	found
periphery-contracts	aave-v3-periphery@2.4.1	none
onez-core	aave/gho-core	none

The non-pertinent changes are comprised of differences in code formatting and updates to solidity pragmas. The only pertinent changes are found in ICreditDelegationToken.sol and DebtTokenBase.sol, which both have delegationWithSig disabled. This change does not pose threat to the protocol security.

Summary

Summary: the ZeroLend codebase contains no changes that undermine security of logic provided by Aave.

Deployment check: storage

We thoroughly examine both public and private storage, as well as immutable and constant variables, to ensure that there are no misconfigurations, especially:

1. Incorrect or outdated addresses to other smart contracts referenced in the scope of work (SoW) - this includes addresses stored in variables, mappings, and other data structures.
2. Any references to other smart contracts or externally owned accounts (EOAs) that may be incorrect or outdated.
3. Any incorrect protocol settings stored in variables or other data structures.
4. Misconfigurations related to the roles and permissions of the contract.
5. Governance issues that may impact the operation and business logic of the smart contract.

Statistics by issue type

type	**comment**	**# found**
EOA	Externally-owned account possesses some kind of privileged access to a contract imposing centralization risks on the protocol.	12
out of scope contract	A contract not in scope of the present report possesses some kind of privileged access to another contract, which undermines the guarantees of the safe protocol setup	5
incorrect value	Any kind of incorrect or unassigned values of contract's storage.	12
total		**29**

ID-1. AaveOracle

issue #	issue type	re-check status
1	incorrect value	dismissed
2	incorrect value	dismissed

1. fallbackOracle = 0x0
    - **comment:** the ZeroLend team stated that this value can be unset since it does not affect the protocol in any way.
2. baseCurrency = 0x0
    - **comment:** the ZeroLend team stated that this value can be unset since it does not affect the protocol in any way.

ID-2. ACLManager

issue #	issue type	re-check status
1	EOA	fixed
2	out of scope contract	fixed
3	incorrect value	fixed

1. EOA 0xb76f765a785eca438e1d95f594490088afaf9acc possesses the following roles:
    - DEFAULT_ADMIN
    - EMERGENCY_ADMIN
    - POOL_ADMIN

- **fix:** EOA 0xb76f765a785eca438e1d95f594490088afaf9acc does not possess any of the roles above.

2. Out of scope contract 0x18F21fE46470F668cE72391Bb870A1822703a4fA possesses the following roles:
    - ASSET_LISTING_ADMIN
    - POOL_ADMIN

- **fix:** out of scope contract 0x18F21fE46470F668cE72391Bb870A1822703a4fA does not possess any of the roles above.

3. No address possesses the following roles
    - RISK_ADMIN
    - FLASH_BORROWER
    - BRIDGE

- **fix:** MultisigWallet contract 0x7b08d0d9D6f450243500338C39B1c9F01a30d801 possesses RISK_ADMIN role. The ZeroLend team indicated, that unset FLASH_BORROWER and BRIDGE roles are not an issue.

At the time of publication of the present report the contract's roles are

role	address	comment
DEFAULT_ADMIN	0x861cc6724d0aa7ec7a86... 0x7b08d0d9d6f450243500...	Timelock Multisig
POOL_ADMIN	0x861cc6724d0aa7ec7a86... 0x7b08d0d9d6f450243500...	Timelock Multisig
EMERGENCY_ADMIN	0x7b08d0d9d6f450243500...	Multisig
RISK_ADMIN	0x7b08d0d9d6f450243500...	Multisig
ASSET_LISTING_ADMIN	0x7b08d0d9d6f450243500...	Multisig
FLASH_BORROWER	none	
BRIDGE	none	

ID-5. EmissionManager

issue #	issue type	re-check status
1	EOA	fixed
2	EOA	fixed

1. _owner = 0xb76F765A785eCa438e1d95f594490088aFAF9acc - EOA
    - **fix:** _owner = 0x7b08d0d9D6f450243500338C39B1c9F01a30d801 -- ZeroLend MultisigWallet
2. _emissionAdmins, mapping(address => address) - EOAs

NOTE: EOAs are marked with ⚠.

reward	admin
0x5AEa5775959fBC2557Cc...	0xb76F765A785eCa438e1d... ⚠
0x9793eac2fECef55248ef...	0xb76F765A785eCa438e1d... ⚠

- **fix:** at the publication date of the present report the admin for each reward is 0x7b08d0d9D6f450243500338C39B1c9F01a30d801 -- ZeroLend MultisigWallet

ID-7. IncentivesProxy

issue #	issue type	re-check status
1	incorrect value	dismissed

1. _authorizedClaimers, mapping(address => address) -- no authorized claimers set
   - **comment:** the ZeroLend team stated that this issue can be ignored.

ID-13. ReservesSetupHelper

issue #	issue type	re-check status
1	EOA	**fixed**
2	incorrect value	dismissed
3	incorrect value	dismissed

1. _owner = 0xb76F765A785eCa438e1d95f594490088aFAF9acc -- EOA
   - **fix:** _owner = 0x7b08d0d9D6f450243500338C39B1c9F01a30d801 -- ZeroLend MultiSigWallet
2. _addressesProvider = 0x0
   - **comment:** the ZeroLend team stated that this value can be unset since it does not affect the protocol in any way.
3. _pool = 0x0
   - **comment:** the ZeroLend team stated that this value can be unset since it does not affect the protocol in any way.

ID-14. ReserveStrategy-rateStrategyStableOne

issue #	issue type	re-check status
1	incorrect value	dismissed

1. `baseVariableBorrowRate = 0`
   - **comment:** the ZeroLend team stated that this value can be unset at the current stage of the protocol.

ID-15. ReserveStrategy-rateStrategyStableTwo

issue #	issue type	re-check status
1	incorrect value	dismissed

1. `baseVariableBorrowRate = 0`
   - **comment:** the ZeroLend team stated that this value can be unset at the current stage of the protocol.

ID-16. ReserveStrategy-rateStrategyVolatileOne

issue #	issue type	re-check status
1	incorrect value	dismissed

1. `baseVariableBorrowRate = 0`
   - **comment:** the ZeroLend team stated that this value can be unset at the current stage of the protocol.

ID-22. WrappedTokenGatewayV3

issue #	issue type	re-check status
1	EOA	fixed

1. _owner = 0xb76F765A785eCa438e1d95f594490088aFAF9acc -- EOA
   - **fix:** _owner = 0x7b08d0d9D6f450243500338C39B1c9F01a30d801 -- ZeroLend MultiSigWallet

ID-23. PoolAddressesProvider

issue #	issue type	re-check status
1	EOA	fixed
2	incorrect value	dismissed

1. _owner = 0xb76F765A785eCa438e1d95f594490088aFAF9acc -- EOA
   - **fix:** _owner = 0x7b08d0d9D6f450243500338C39B1c9F01a30d801 -- ZeroLend MultiSigWallet
2. _addresses[PRICE_ORACLE_SENTINEL] = 0x0 -- unset priceOracleSentinel contract
   - **comment:** the ZeroLend team stated that this issue can be ignored.

ID-24. USDC-USD

issue #	issue type	re-check status
1	out of scope contract	dismissed

1. pyth = f087c864aeccfb6a2bf1af6a0382b0d0f6c5d834 -- out of scope contract
   - **comment:** the ZeroLend team stated that this is an official Pyth Network address on zkSync Era, which can be verified here.

ID-25. WETH-USD

issue #	issue type	re-check status
1	out of scope contract	dismissed

1. pyth = f087c864aeccfb6a2bf1af6a0382b0d0f6c5d834 -- out of scope contract
   - **comment:** the ZeroLend team stated that this is an official Pyth Network address on zkSync Era, which can be verified here.

ID-26. USDT-USD

issue #	issue type	re-check status
1	out of scope contract	dismissed

1. pyth = f087c864aeccfb6a2bf1af6a0382b0d0f6c5d834 -- out of scope contract
   - **comment:** the ZeroLend team stated that this is an official Pyth Network address on zkSync Era, which can be verified here.

ID-33. PoolAddressesProviderRegistry

issue #	issue type	re-check status
1	EOA	**fixed**

1. _owner = 0xb76F765A785eCa438e1d95f594490088aFAF9acc -- EOA
   - **fix:** _owner = 0x7b08d0d9D6f450243500338C39B1c9F01a30d801 -- ZeroLend MultiSigWallet

ID-36. Treasury-Controller

issue #	issue type	re-check status
1	EOA	**fixed**

1. _owner = 0xb76F765A785eCa438e1d95f594490088aFAF9acc -- EOA
   - **fix:** _owner = 0x7b08d0d9D6f450243500338C39B1c9F01a30d801 -- ZeroLend MultiSigWallet

ID-48. Timelock

issue #	issue type	re-check status
1	EOA	fixed

1. EOA 0xb76f765a785eca438e1d95f594490088afaf9acc possesses the following roles:
    - PROPOSER
    - EXECUTOR
    - CANCELLER

- **fix:** EOA 0xb76f765a785eca438e1d95f594490088afaf9acc does not possess any of the roles above.

At the time of publication of the present report the contract's roles are

role	address	comment
DEFAULT_ADMIN	0x7b08d0d9d6f450243500... 0x861cc6724d0aa7ec7a86...	Multisig Timelock
PROPOSER	0x7b08d0d9d6f450243500...	Multisig
EXECUTOR	0x7b08d0d9d6f450243500...	Multisig
CANCELLER	0x7b08d0d9d6f450243500...	Multisig

ID-49. Multisig

issue #	issue type	re-check status
1	incorrect value	fixed

1. The ratio of required / owners.length = 2 / 2. Multisig contract is considered to protect from private key compromise when the ratio of required / owners.length = 2 / 3.
    - **fix:** at the date of publication of the present report the ratio of required / owners.length = 2 / 3

ID-50. GhoToken

issue #	issue type	re-check status
1	EOA	fixed
2	out of scope contract	fixed

1. EOA 0xb76f765a785eca438e1d95f594490088afaf9acc possesses the following roles:
    - DEFAULT_ADMIN
    - FACILITATOR_MANAGER
    - BUCKET_MANAGER

- **fix:** EOA 0xb76f765a785eca438e1d95f594490088afaf9acc does not possess any of the roles above.

2. Out of scope contract 0x18f21fe46470f668ce72391bb870a1822703a4fa possesses the the following roles:
    - DEFAULT_ADMIN
    - FACILITATOR_MANAGER

- **fix:** out of scope contract 0x18F21fE46470F668cE72391Bb870A1822703a4fA does not possess any of the roles above.

At the time of publication of the present report the contract's roles are

role	address	comment
DEFAULT_ADMIN	0x7b08d0d9d6f450243500...	Multisig
FACILITATOR_MANAGER	none	
BUCKET_MANAGER	none	

The ZeroLend team indicated, that unset FACILITATOR_MANAGER and BUCKET_MANAGER roles are not an issue.

ID-57. TransferStrategy

issue #	issue type	re-check status
1	EOA	dismissed
2	EOA	dismissed

1. rewardsAdmin = 0xb76f765a785eca438e1d95f594490088afaf9acc -- EOA
   - **comment:** the ZeroLend team stated the EOA as an admin of this particular contract is not an issue at this stage of the protocol.
2. rewardsVault = 0xb76f765a785eca438e1d95f594490088afaf9acc -- EOA
   - **comment:** the ZeroLend team stated the EOA as a rewards vault of this particular contract is not an issue at this stage of the protocol.

ID-65. ONEZ-VariableDebtToken

issue #	issue type	re-check status
1	incorrect value	dismissed

1. _discountToken = 0x0
   - **comment:** the ZeroLend team stated that this issue can be ignored as there will be no discount token for the ONEZ stablecoin.

Disclaimers

Mundus disclaimer

The smart contracts given for audit have been analyzed in accordance with the best industry practices at the date of this report, in relation to cybersecurity vulnerabilities and issues in smart contract source code, the details of which are disclosed in this report (Source Code); the Source Code compilation, deployment, and functionality (performing the intended functions).

The audit makes no statements or warranties on the security of the code. It also cannot be considered as a sufficient assessment regarding the utility and safety of the code, bug-free status, or any other statements of the contract. While we have done our best in conducting the analysis and producing this report, it is important to note that you should not rely on this report only — we recommend proceeding with several independent audits and a public bug bounty program to ensure the security of smart contracts.

Technical disclaimers

Smart contracts are deployed and executed on a blockchain platform. The platform, its programming language, and other software related to the smart contract can have vulnerabilities that can lead to hacks. Thus, the audit can't guarantee the explicit security of the audited smart contracts.