PROTOFIRE

October 2025

Deployment Verification

Tokos

Content

Executive Summary

This report provides a verification of the deployed smart contracts for the Tokos on the Somnia network. The analysis was conducted by cross-referencing the provided list of 43 deployed contract addresses against the standard Aave V3 protocol architecture.
The deployment is assessed to be architecturally complete and correct. All core protocol contracts, essential periphery contracts and required helper contracts are present. The deployment correctly utilizes the proxy pattern for upgradeability and includes the necessary logic implementations for all core functionalities.
One temporary deployment helper contract (ReservesSetupHelper) was identified, which is expected for initial setup but is not part of the long-term operational protocol. This is not a security concern but a normal part of the deployment process. The naming conventions are consistent and adapted for the Somnia network.

Scope of Verification

The scope of this verification was to analyze the provided list of 43 deployed smart contract addresses and their corresponding names. The objective was to:
Verify Completeness: Ensure that no critical core or periphery contracts from the standard Aave V3 deployment are missing.
Verify Correctness: Confirm that the roles of the deployed contracts align with their intended function within the Aave V3 architecture.
Identify Anomalies: Identify any contracts that are not part of a standard Aave V3 deployment or note any deviations in naming and structure.

This report does not constitute a full security audit of the smart contract code or an on-chain verification of contract state and parameters.

Overall Findings

The deployment includes all necessary components. The core logic (Pool, PoolConfigurator), governance (ACLManager), data feeds (AaveOracle), and token implementations (AToken, DebtTokens) are all present. Key periphery contracts that enable a functional user experience (DataProvider, Gateway, etc.) have also been deployed.
The architecture correctly follows the Aave V3 standard of using modular logic libraries and separating proxy contracts from their implementations. This is a best practice that allows for future upgrades without disrupting the protocol's core addresses.

Detailed Contract Analysis

Contract Name	Deployed Address	Role & Purpose	Verification Status
Core Registries & Proxies			
PoolAddressesProviderRegistry	0xac5ba04B233A8Dfe0d013c705Ce6B7B36179bCB7	A top-level registry to list multiple Aave markets.	Verified - Core Component
PoolAddressesProvider	0x1C13Fea2A9a3Ae9962f12B6afAC1AFcd8205f752	The central "address book" for this specific Aave market.	Verified - Core Component
Pool - Proxy	0xEC6758e6324c167DB39B6908036240460a2b0168	The main user-facing contract for supply, borrow, and repay actions.	Verified - Core Component
PoolConfigurator - Proxy	0x9B67C48e6E8FE1c4e59B83358bf66C82168be7E8	The administrative contract for configuring reserves and market parameters.	Verified - Core Component
Treasury - Proxy	0x2BaC9d4D33B6AbBDD29ac050e03Be0a1032E6b1D	The contract that collects protocol-generated fees.	Verified - Core Component
Incentives - Proxy	0xdA28C6035EA2026E51a697cBEe1FaFaAbA495acC	The main contract for managing and distributing rewards to users.	Verified - Core Component
Logic Implementations & Libraries			
PoolLogic	0x054d3103c6fDbF091B2E42BdaF46b0BA24A60938	Contains the master logic for the main Pool contract.	Verified - Core Component
SupplyLogic	0x0b4f7C890FBAEA3c25bA1eEB9567eEfd07D774Bb	Library containing the specific logic for supply and withdraw operations.	Verified - Core Component

BorrowLogic	0x06A5726959776C4c8f592D440EB8A35FB5C407E2	Library containing the specific logic for borrow and repay operations.	Verified - Core Component
LiquidationLogic	0x68CF59B862b77123403633D372E505388aa47500	Library with logic for handling the liquidation of undercollateralized loans.	Verified - Core Component
EModeLogic	0x42f591A98B240B0FAca1430d2cDB3a94Be271513	Library with logic for Aave's High-Efficiency Mode (E-Mode).	Verified - Core Component
FlashLoanLogic	0x6c6586f29a4D387557D7881C10dE7061B8716bE4	Library with logic for executing capital-efficient flash loans.	Verified - Core Component
BridgeLogic	0xD9517c72807a85abd7B9baCC602428793D65e7fF	Library with logic related to Aave's cross-chain features.	Verified - Core Component
PoolConfigurator - Implementation	0xCA4E8E357077e1F9d05eABBfAB096e44b59F16EE	Contains the master administrative logic for the PoolConfigurator.	Verified - Core Component
ConfiguratorLogic	0x008AEA7ec3db936bd2fC76dC5BBb4F108b8Ee7DC	A library of functions used by the PoolConfigurator-Implementation.	Verified - Core Component
Treasury - Implementation	0x91D5aB4458112D7910cE55fA9FbdB58A4d661B63	Contains the logic for the Treasury contract's operations.	Verified - Core Component
IncentivesV2 - Implementation	0x839934166b2ECd942bE97Da7f4DF051530eD0601	Contains the logic for the Incentives Controller.	Verified - Core Component
Governance & Oracles			

ACLManager	0xCf974cC0C04c49Fb1fD52Bd538Fb366544B98C32	The Access Control List manager, defining permissions for admin roles.	Verified - Core Component
AaveOracle	0x9b7B5d69a08071ce74C2f3D655038E3b782eAaFc	The primary price oracle that provides asset prices to the protocol.	Verified - Core Component
Treasury - Controller	0x342b677edfB5198E62dC8d4aCd03751A4262B4d9	The controller contract that owns and manages the Treasury's funds.	Verified - Core Component
EmissionManager	0x3eD88b09ff521e17C00c847826bb551C1b3ca5bC	Manages the rate of reward token distribution by the incentives system.	Verified - Core Component
Token Implementations (Templates)			
AToken	0xdE2a0De646d659fe3a9a591db75AEfC8Aa110883	The base implementation (template) for standard interest-bearing aTokens.	Verified - Core Component
DelegationAwareAToken	0xA44158Ae450ff1546a665c4d8b676ecdD9C7ce1F	The base implementation for aTokens that support credit delegation.	Verified - Core Component
StableDebtToken	0xcc1D8278Af3E152c3148d1E21E8D8AE09fcE3D35	The base implementation (template) for stable rate debt tokens.	Verified - Core Component
VariableDebtToken	0xC044d88C2c3513d571390d934B9d4918d6426bc6	The base implementation (template) for variable rate debt tokens.	Verified - Core Component

Interest Rate Strategies			
ReserveStrategy - rateStrategyVolatileOne	0xDd928DF5A5e19591cCfa8e46857480F745F038b7	Contract that defines a volatile interest rate model for assets.	Verified - Core Component
ReserveStrategy - rateStrategyStableOne	0xd82de360f3E15BbeBe85b4726aF92E1A0351ab9b	Contract that defines a stable interest rate model for assets.	Verified - Core Component
ReserveStrategy - rateStrategyStableTwo	0xB51f0ec3b33fD49EF82622b0343d07eb264Ac730	Contract that defines an alternative stable interest rate model.	Verified - Core Component
Periphery & Helper Contracts			
PoolDataProvider	0x6A8c1d9ff923B75D662Ee839E4AD8949279bAF10	The AaveProtocolDataProvider for fetching aggregated protocol data.	Verified - Essential Helper
UiPoolDataProvider V3	0x5ef828E2C7C55eea505dA3310b0831eD01189e3E	Helper contract with data formatted specifically for user interface consumption.	Verified - Essential Helper
UiIncentiveDataProviderV3	0x732771d89b2D15a17Ed66aAD43b7bc37116719b5	Helper contract for fetching formatted incentive data for the UI.	Verified - Essential Helper
WalletBalanceProvider	0xcEd68D9Acd2663ECCc875996690b1380A754da8A	Helper contract to fetch multiple user token balances in a single on-chain call.	Verified - Essential Helper
WrappedTokenGatewayV3	0xc97d0602b501B5123a0558dDFEb2A28fD6C78dB9	Gateway contract to allow seamless interaction using the network's native asset.	Verified - Essential Helper

Deployment-Specific Contracts			
ReservesSetupHelper	0x543a0437cED7794aee010FcCf3eAbFf201B2181B	A temporary helper contract used only during initial deployment to configure reserves.	Verified - Deployment Utility

In addition to the deployed contracts, generated tokens were also verified.

Generated Asset-Specific Tokens			
WSOMI-AToken	0xFe171d9d2679c4544ADD1a20d565C251cEd9FF4A	The unique, interest-bearing aToken for the WSOMI asset.	Verified
WSOMI-VariableDebtToken	0xe8138d52b0F6Ca180D4d86260688456091fbBAA4	The unique token represents a variable rate WSOMI debt position.	Verified
WSOMI-StableDebtToken	0x1B3509b886EF5868a8D83Deccd7646bacFF2393E	The unique token represents a stable rate WSOMI debt position.	Verified
USDC.E-AToken	0x024c0D0f79FAa994453CC955Df986A361E97f1a9	The unique, interest-bearing aToken for the USDC.E asset.	Verified
USDC.E-VariableDebtToken	0x418763671a385aeb7629e00EEcE6Adb9bF104cbC	The unique token represents a variable rate USDC.E debt position.	Verified
USDC.E-StableDebtToken	0xF2F84725209581D63007Cb843b97b0302C115900	The unique token represents a stable rate USDC.E debt position.	Verified
WETH-AToken	0x27b52440EE1730bE92E2f17395DC6699Ed25211b	The unique, interest-bearing aToken for the WETH asset.	Verified

WETH-VariableDebtToken	0x3bb29cF53e13dc38Fdb330C8422dfe977a1c0ee6	The unique token represents a variable rate WETH debt position.	Verified
WETH-StableDebtToken	0xDa80c646D37f291A385b93759f7fD45373c400B8	The unique token represents a stable rate WETH debt position.	Verified

Observations and Notes

The ReservesSetupHelper contract is a temporary utility used during the initial setup to batch-configure multiple reserves in a single transaction. It is not part of the operational protocol and has no function after deployment. Its presence in the deployed list is normal and does not pose a risk.

The list includes nine token contracts for WSOMI, USDC.E, and WETH (one aToken, one stable debt token, and one variable debt token for each). These are not extraneous contracts; they are the expected result of successfully initializing these three assets in the protocol via the PoolConfigurator.

Additionally, the owner of the deployed contracts is a multisig wallet with the address **0xe61f29aEb3BBc4269839adF0B44530d025bBd85c**, which provides an added layer of security control over administrative actions.

Conclusion

The provided list of 43 deployed contracts represents a complete deployment of a Aave V3 fork. All required core and periphery components have been successfully deployed, and the structure aligns with established best practices for creating a functional and upgradeable lending market. No core components appear to be missing from the deployment.

Disclaimer

This report is a verification of the completeness of the provided contract list against the standard Aave V3 architecture. It is not a security audit of the smart contract source code, nor does it verify the correctness of on-chain risk parameters, configurations, or administrative permissions.