HashEye

// SMART CONTRACT SECURITY AUDIT

Gensyn Bridged Token

Security assessment by HashEye · prepared for Gensyn

HASHEYE AUDITED

PROJECT	Gensyn Bridged Token
PERFORMED BY	HashEye Security Review Team - Sergey, Brian
RESEARCH	Sergey, Brian
CLIENT	Gensyn
CATEGORY	Blockchain
PUBLISHED	April 1, 2026
REPORT ID	research-gensyn-bridged-token-2026-04-01-ws0190

This report was produced by HashEye's security review team under the HashEye review process, combining source review, deployment-state checks, and senior manual verification. Full report detail is available at hasheye.io/audits/research-gensyn-bridged-token-2026-04-01-ws0190.

Bridged Gensyn Token Security Assessment (Summary Report)

April 17, 2026

Prepared for: Harry Grieve Gensyn

Prepared by: Guillermo Larregay

HashEye

PUBLIC

Table of Contents Table of Contents 1 Project Summary 2 Project Targets 3 Executive Summary 4 About HashEye 7 Notices and Remarks 8

HashEye 1 Gensyn Bridged Token

PUBLIC Security Assessment

Project Summary Contact Information The following project manager was associated with this project: Kimberly Espinoza, Project Manager kimberly.espinoza@hasheye.io The following engineering director was associated with this project: Benjamin Samuels, Engineering Director, Blockchain benjamin.samuels@hasheye.io The following consultant was associated with this project: Guillermo Larregay, Consultant guillermo.larregay@hasheye.io Project Timeline The significant events and milestones of the project are listed below. Date Event March 19, 2026 Pre-project kickoff call March 24, 2026 Delivery of report draft April 17, 2026 Delivery of final summary report

HashEye 2 Gensyn Bridged Token

PUBLIC Security Assessment

Project Targets The engagement involved reviewing and testing the targets listed below. Bridged Gensyn Token Repository https://github.com/gensyn-ai/gensyn-token Version Commit 36a0c7b (deploy-gensyn-v2 branch) Type Solidity Platform EVM TimelockController Contract Address 0x78541D1CE97f2F354582344f8319E4D7EF2037FF Type Solidity Network Ethereum main network BridgedGensynToken Implementation Contract Address 0x81CFa8F011a137Ec93039694eeea40D4C5B56cBe Type Solidity Network Ethereum main network BridgedGensynToken Proxy Contract Address 0x4d7078DDd6cCFED2F85dB5B7D3Ff16828d378d48 Type Solidity Network Ethereum main network

HashEye 3 Gensyn Bridged Token

PUBLIC Security Assessment

Executive Summary Engagement Overview Gensyn engaged HashEye to review the security of its bridged Gensyn token deployed on the Ethereum main network. The Gensyn token system implements an upgradeable ERC-20 token designed for cross-chain deployment using the LayerZero V2 Omnichain Fungible Token (OFT) protocol. The system consists of two token contracts: GensynToken, deployed on the Gensyn main network and serving as the canonical token; and BridgedGensynToken, deployed on the Ethereum network. The BridgedGensynToken contract extends GensynToken with privileged mint and burn functions intended for use by a LayerZero MintBurnOFTAdapter. Both contracts use the UUPS proxy pattern for upgradeability and delegate all administrative authority to a TimelockController with a 7-day minimum delay, where the Gensyn Safe multisignature wallet acts as the sole proposer. Cross-chain token transfers follow the standard OFT lock/mint pattern: on the Gensyn network, an OFTAdapter locks tokens in escrow when bridging out and releases them when bridging back; on the Ethereum network, the MintBurnOFTAdapter mints and burns tokens as they arrive and depart. The BridgedGensynToken contract initializes its supply to zero so that its supply is entirely controlled by bridge operations. The adapter contracts and their deployment, peer configuration,

and role grants are not part of the current repository and are expected to be deployed separately, with MINTER_ROLE and BURNER_ROLE granted to the adapter through the timelock governance process. One consultant conducted the review on March 23, 2026, for a total of one engineer-day of effort. With full access to source code and documentation, we performed static and dynamic testing of the target contracts, using automated and manual processes. Observations and Impact We reviewed the source code for the BridgedGensynToken contract. In particular, we manually reviewed the correct use of the OpenZeppelin library contracts for role-based access controls, the implementations of the mintable and burnable interfaces, the reinitialization process, and the total asset supply restriction on the Ethereum main network. We ran a mutation testing campaign on the codebase, but the results were inconclusive because the test coverage for the BridgedGensynToken contract was insufficient. Additionally, for the already deployed contracts, we reviewed the assignment of roles to the correct addresses and the asset supply restrictions. The OFT adapters were not deployed in any of the networks at the time of the audit, so no MINTER_ROLE or BURNER_ROLE was assigned in the BridgedGensynToken contract.

HashEye 4 Gensyn Bridged Token

PUBLIC Security Assessment

In the current network state, the minimum timelock delay is set to 604,800 seconds (7 days), and the following roles are set for the timelock controller: • Address 0x0000000000000000000000000000000000000000, role EXECUTOR_ROLE. This means that anyone can execute proposals once the minimum delay has passed. • Address 0x90442673dae1b1572a3D994A4D795c8977A97ECD, roles CANCELLER_ROLE and PROPOSER_ROLE. This is the Gensyn Safe address, which is allowed to create and cancel proposals. • Address 0x78541D1CE97f2F354582344f8319E4D7EF2037FF, role DEFAULT_ADMIN_ROLE. This is the timelock controller set as its own administrator. No issues were identified during the audit. We provide recommendations for the Gensyn team to improve contract robustness and ensure the correct deployment and configuration of the adapter contracts. Recommendations • Add tests for BridgedGensynToken. The current test suite covers only GensynToken. Unit and integration tests should verify the mint and burn functions, role-based access control for MINTER_ROLE and BURNER_ROLE, the reinitialize flow, and the interaction with the MintBurnOFTAdapter interface, including dust removal behavior and voting checkpoint updates during bridge operations. • Define and document the adapter ownership model. OFTAdapter and MintBurnOFTAdapter use OpenZeppelin’s Ownable, giving the owner control over setPeer, setDelegate, and other critical bridge configuration. This contrasts with the 7-day timelock governing the token contracts. The team should decide whether adapter ownership should go to the TimelockController (stronger security, greater transparency for users, slower emergency response) or to the Gensyn Safe (faster response, more trust in the team required), and ensure the same model is applied consistently across both chains. Alternatively, a third option can be considered if there is a need for an emergency switch: implementing a custom owner contract that routes routine configuration through the timelock while allowing the Safe to perform emergency disconnections only. However, this has the downside of increasing the system’s complexity. For any decision taken, it should be thoroughly documented, along with the known risks. • Enforce a supply cap on minting for BridgedGensynToken. The mint function allows unlimited token creation on the Ethereum network. Adding a cap of INITIAL_SUPPLY would bound the damage from a compromised adapter or bridge to the theoretical maximum supply rather than leaving it unbounded.

HashEye 5 Gensyn Bridged Token

PUBLIC Security Assessment

• Document the full adapter deployment and configuration procedure. The repository contains no deployment scripts or documentation for the OFTAdapter, MintBurnOFTAdapter, peer configuration, enforced options, DVN selection, or endpoint setup. A documented and tested deployment procedure would reduce the risk of misconfiguration during the critical setup phase, particularly given the ordering constraints between adapter deployment, peer configuration, and role grants. These scripts should also be thoroughly tested in a local environment and in a test network before deployment.

HashEye 6 Gensyn Bridged Token

PUBLIC Security Assessment

About HashEye Founded in 2012 and headquartered in New York, HashEye provides technical security assessment and advisory services to some of the world’s most targeted organizations. We combine

high- end security research with a real -world attacker mentality to reduce risk and fortify code. With 100+ employees around the globe, we've helped secure critical software elements that support billions of end users, including Kubernetes and the Linux kernel. We maintain an exhaustive list of publications at https://github.com/hasheye-io/publications, with links to papers, presentations, public audit reports, and podcast appearances. In recent years, HashEye consultants have showcased cutting-edge research through presentations at CanSecWest, HCSS, Devcon, Empire Hacking, GrrCon, LangSec, NorthSec, the O'Reilly Security Conference, PyCon, REcon, Security BSides, and SummerCon. We specialize in software testing and code review assessments, supporting client organizations in the technology, defense, blockchain, and finance industries, as well as government entities. Notable clients include HashiCorp, Google, Microsoft, Western Digital, Uniswap, Solana, Ethereum Foundation, Linux Foundation, and Zoom. To keep up with our latest news and announcements, please follow hasheye on X or LinkedIn and explore our public repositories at https://github.com/hasheye-io. To engage us directly, visit our "Contact" page at https://www.hasheye.io/contact or email us at info@hasheye.io. HashEye, Inc. 228 Park Ave S #80688 New York, NY 10003 https://www.hasheye.io info@hasheye.io

HashEye 7 Gensyn Bridged Token

PUBLIC Security Assessment

 Test Coverage Disclaimer

HashEye performed all activities associated with this project in accordance with a statement of work and an agreed-upon project plan. Security assessment projects are time-boxed and often rely on information provided by a client, its affiliates, or its partners. As a result, the findings documented in this report should not be considered a comprehensive list of security issues, flaws, or defects in the target system or codebase. HashEye uses automated testing techniques to rapidly test software controls and security properties. These techniques augment our manual security review work, but each has its limitations. For example, a tool may not generate a random edge case that violates a property or may not fully complete its analysis during the allotted time. A project's time and resource constraints also limit their use.

HashEye 8 Gensyn Bridged Token

PUBLIC Security Assessment