



Security Review For Atleta Network



Collaborative Audit Prepared For:
Lead Security Expert(s):

Atleta Network
hildingr
vinica_boy

Date Audited:

January 17 - January 19, 2026

Introduction

VestingAtlaImmutable is a fully immutable, Merkle-based ATLA vesting smart contract designed to distribute tokens according to predefined cliff and vesting schedules without any administrative control after deployment.

The contract is initialized once at deployment time, with all vesting parameters, beneficiary allocations, and the vesting start timestamp permanently fixed in the constructor. After deployment, no party retains the ability to modify vesting parameters, change beneficiaries, withdraw funds, pause execution, or otherwise intervene in the vesting process.

Key features and design choices:

- Merkle-proof allocations
- Immutable vesting configuration
- Cliff, TGE, and periodic unlocks
- Calendar-aware time calculations
- Deterministic start time
- Per-beneficiary accounting
- Single and batch claiming
- Strict one-time initialization
- No administrative privileges

In summary, VestingAtlaImmutable provides a trust-minimized, code-level immutable ATLA vesting mechanism with Merkle-verified allocations, calendar-accurate schedules, and fully permissionless claiming.

Scope

Repository: [potemkinViktor/AtletaVesting](#)

Audited Commit: [b433710b96214124ecd0d3cde81742834bb2a6e1](#)

Final Commit: [b84c660447ee4b99cf91b0bb2e5cc75150e282c0](#)

Files:

- [contracts/libs/BokkyPooBahsDateTimeLibrary.sol](#)
- [contracts/VestingAtlaImmutable.sol](#)

Final Commit Hash

[b84c660447ee4b99cf91b0bb2e5cc75150e282c0](#)

Findings

Each issue has an assigned severity:

- High issues are directly exploitable security vulnerabilities that need to be fixed.
- Medium issues are security vulnerabilities that may not be directly exploitable or may require certain conditions in order to be exploited. All major issues should be addressed.
- Low/Info issues are non-exploitable, informational findings that do not pose a security risk or impact the system's integrity. These issues are typically cosmetic or related to compliance requirements, and are not considered a priority for remediation.

Issues Found

High	Medium	Low/Info
0	0	2

Issues Not Fixed and Not Acknowledged

High	Medium	Low/Info
0	0	0

Issue L-1: Consider additional sanity checks in constructor [RESOLVED]

Source: <https://github.com/sherlock-audit/2026-01-atleta-immutable-contract-jan-17th/issues/3>

Summary

Given that the vesting data is now immutable we could add some additional sanity checks on construction to catch incorrect parameter inputs.

Something like $\text{totalPeriods} * \text{durationPeriodInMonths} + \text{cliffDuration} < 10 * 12$, max 10 year vesting or something is wrong.

We could set a max value on VESTING_START_TIME based on the current timestamp. Limit how far in the future it can be set.

Could also pass a leaf and merkel proof and make sure the root in the vestingData is viable.

Recommendation

These sanity checks are not as important if we send the funds in a separate transaction since we can performed the off-chain before the funds are sent.

Issue L-2: ClaimTge bool is redundant [RESOLVED]

Source: <https://github.com/sherlock-audit/2026-01-atleta-immutable-contract-jan-17th/issues/4>

Summary

The `claimedTge` in the `vestingInfo` struct is redundant since `amountPaidOut` must be `>0` if the `tge` has been claimed so the information is implicitly there.

We can remove this bool and below code to save on gas.

<https://github.com/sherlock-audit/2026-01-atleta-immutable-contract-jan-17th/blob/058ed194c87163781a5ae15f602c5c0ba7a6d4f9/AtletaVesting/contracts/VestingAtlaImmutable.sol#L181-L183>

Recommendation

Remove the bool and the code that sets it to `true`

Disclaimers

Sherlock does not provide guarantees nor warranties relating to the security of the project.

Usage of all smart contract software is at the respective users' sole risk and is the users' responsibility.