INVI TOKEN - audit

Security Assessment

CertiK Assessed on Apr 22nd, 2025

CertiK Assessed on Apr 22nd, 2025

INVI TOKEN - audit

The security assessment was prepared by CertiK, the leader in Web3.0 security.

Executive Summary

TYPES	ECOSYSTEM	METHODS
ERC-20, Vesting	EVM Compatible	Formal Verification, Manual Review, Static Analysis

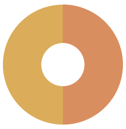
LANGUAGE	TIMELINE	KEY COMPONENTS
Solidity	Delivered on 04/22/2025	N/A

CODEBASE	COMMITS
d3101960a23dbaf9674fef5597d8940392b6462a 22477495c7aced18b875e33ee997b2523d82fa23 View All in Codebase Page	d3101960a23dbaf9674fef5597d8940392b6462a 22477495c7aced18b875e33ee997b2523d82fa23 View All in Codebase Page

Highlighted Centralization Risks

ⓘ Initial owner token share is 100%

Vulnerability Summary



4	2	0	2	0
Total Findings	Resolved	Partially Resolved	Acknowledged	Declined

🟧	1	Centralization	1 Acknowledged	Centralization findings highlight privileged roles & functions and their capabilities, or instances where the project takes custody of users' assets.
🟥	0	Critical		Critical risks are those that impact the safe functioning of a platform and must be addressed before launch. Users should not invest in any project with outstanding critical risks.
🟧	1	Major	1 Acknowledged	Major risks may include logical errors that, under specific circumstances, could result in fund losses or loss of project control.
🟨	2	Medium	2 Resolved	Medium risks may not pose a direct risk to users' funds, but they can affect the overall functioning of a platform.
🟨	0	Minor		Minor risks can be any of the above, but on a smaller scale. They generally do not compromise the overall integrity of the project, but they may be less efficient than other solutions.

■	0	Informational	Informational errors are often recommendations to improve the style of the code or certain operations to fall within industry best practices. They usually do not affect the overall functioning of the code.

TABLE OF CONTENTS | INVI TOKEN - AUDIT

CODEBASE | INVI TOKEN - AUDIT

Repository

d3101960a23dbaf9674fef5597d8940392b6462a

22477495c7aced18b875e33ee997b2523d82fa23

Commit

d3101960a23dbaf9674fef5597d8940392b6462a 22477495c7aced18b875e33ee997b2523d82fa23

AUDIT SCOPE | INVI TOKEN - AUDIT

2 files audited ● 2 files with Acknowledged findings

ID	Repo	File	SHA256 Checksum
● ITI	ryuk6911/INVI_TOKEN	InvincibleToken.sol	efab43b4cdc59a26dc3aad878b0d6915ea3978c29138f6f8066eca57c6cff594
● IVI	ryuk6911/INVI_TOKEN	InvincibleVesting.sol	0f86d5095ba02f05d498d309d1a81e86de702c89fc2f5ed28c6f3ffc7ab32778

APPROACH & METHODS | INVI TOKEN - AUDIT

This report has been prepared for INVI TOKEN to discover issues and vulnerabilities in the source code of the INVI TOKEN - audit project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Formal Verification, Manual Review, and Static Analysis techniques.

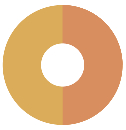
The auditing process pays special attention to the following considerations:

- Testing the smart contracts against both common and uncommon attack vectors.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

The security assessment resulted in findings that ranged from critical to informational. We recommend addressing these findings to ensure a high level of security standards and industry practices. We suggest recommendations that could better serve the project from the security perspective:

- Testing the smart contracts against both common and uncommon attack vectors;
- Enhance general coding practices for better structures of source codes;
- Add enough unit tests to cover the possible use cases;
- Provide more comments per each function for readability, especially contracts that are verified in public;
- Provide more transparency on privileged activities once the protocol is live.

FINDINGS | INVI TOKEN - AUDIT



4	0	1	1	2	0	0
Total Findings	Critical	Centralization	Major	Medium	Minor	Informational

This report has been prepared to discover issues and vulnerabilities for INVI TOKEN - audit. Through this audit, we have uncovered 4 issues ranging from different severity levels. Utilizing the techniques of Formal Verification, Manual Review & Static Analysis to complement rigorous manual code reviews, we discovered the following findings:

ID	Title	Category	Severity	Status
ITA-04	**Centralization Related Risks**	**Centralization**	**Centralization**	● **Acknowledged**
ITA-03	**Initial Token Distribution**	**Centralization**	**Major**	● **Acknowledged**
ITA-05	Infinite Unlock Loop And Incorrect Percentage In `InvincibleVesting`	Design Issue	Medium	● Resolved
ITA-06	Compilation Error In `updateOracle()` Function	Coding Issue	Medium	● Resolved

ITA-04 | CENTRALIZATION RELATED RISKS

Category	Severity	Location	Status
Centralization	● **Centralization**	**InvincibleToken.sol (pre): 50; InvincibleVesting.sol (pre): 45, 53**	● **Acknowledged**

Description

In the contract `Ownable` , the role `_owner` has authority over the following functions:

- `transferOwnership()`
- `renounceOwnership()`

Any compromise to the `_owner` account may allow the hacker to take advantage of this authority and transfer/renounce the ownership.

In the contract `InvincibleVesting` , the role `_owner` has authority over the following functions:

- `setTokenAddress()`

Any compromise to the `_owner` account may allow the hacker to take advantage of this authority and initialize the token address.

Recommendation

The risk describes the current project design and potentially makes iterations to improve in the security operation and level of decentralization, which in most cases cannot be resolved entirely at the present stage. We advise the client to carefully manage the privileged account's private key to avoid any potential risks of being hacked. In general, we strongly recommend centralized privileges or roles in the protocol be improved via a decentralized mechanism or smart-contract-based accounts with enhanced security practices, e.g., multisignature wallets. Indicatively, here are some feasible suggestions that would also mitigate the potential risk at a different level in terms of short-term, long-term and permanent:

Short Term:

Timelock and Multi sign (⅔, ⅗) combination *mitigate* by delaying the sensitive operation and avoiding a single point of key management failure.

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;
  AND
- Assignment of privileged roles to multi-signature wallets to prevent a single point of failure due to the private key compromised;
  AND

- A medium/blog link for sharing the timelock contract and multi-signers addresses information with the public audience.

Long Term:

Timelock and DAO, the combination, *mitigate* by applying decentralization and transparency.

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;
  AND
- Introduction of a DAO/governance/voting module to increase transparency and user involvement.
  AND
- A medium/blog link for sharing the timelock contract, multi-signers addresses, and DAO information with the public audience.

Permanent:

Renouncing the ownership or removing the function can be considered *fully resolved*.

- Renounce the ownership and never claim back the privileged roles.
  OR
- Remove the risky functionality.

Alleviation

[INVI TOKEN Team, 04/24/2025]: The team acknowledged this issue.

[CertiK, 04/24/2025]: It is suggested to implement the aforementioned methods to avoid centralized failure. Also, CertiK strongly encourages the project team to periodically revisit the private key security management of all addresses related to centralized roles.

ITA-03 | INITIAL TOKEN DISTRIBUTION

Category	Severity	Location	Status
Centralization	● **Major**	**InvincibleToken.sol (pre): 44~45**	● **Acknowledged**

Description

All of the INVI tokens are sent to the `vestingContract` address. This is a centralization risk because the address can distribute tokens without obtaining the consensus of the community. Any compromise to the address may allow a hacker to steal and sell tokens on the market, resulting in severe damage to the project.

Recommendation

It is recommended that the team be transparent regarding the initial token distribution process. The token distribution plan should be published in a public location that the community can access. The team should make efforts to restrict access to the private keys of the deployer account or EOAs. A multi-signature (⅔, ⅗) wallet can be used to prevent a single point of failure due to a private key compromise. Additionally, the team can lock up a portion of tokens, release them with a vesting schedule for long-term success, and deanonymize the project team with a third-party KYC provider to create greater accountability.

Alleviation

[INVI TOKEN, 04/22/2025]: In practice, the vestingContract refers to the InvincibleVesting contract in the audit scope, which follows specific rules to release tokens.

[CertiK, 04/22/2025]: It is suggested to implement the aforementioned methods to avoid centralized failure. Also, CertiK strongly encourages the project team to periodically revisit the private key security management of all addresses related to centralized roles.

ITA-05 | INFINITE UNLOCK LOOP AND INCORRECT PERCENTAGE IN `InvincibleVesting`

Category	Severity	Location	Status
Design Issue	● Medium	InvincibleVesting.sol (pre): 79	● Resolved

Description

The `checkUnlock()` function in the `InvincibleVesting` contract is responsible for unlocking and transferring tokens to a designated `beneficiary` based on price and time conditions. The design implements a two-phase release strategy:

- **First unlock**: Transfers **10% of the current contract balance**.
- **Subsequent unlocks**: Each release transfers **5% of the remaining balance** and increases the `currentPriceTarget` by 30%.

However, this approach has several issues:

1. **Never Fully Released**
   Each release after the initial unlock transfers only a fixed percentage of the remaining balance. Since the balance never reaches zero with such logic, **the contract will asymptotically approach zero but never fully unlock all tokens**. This is a common geometric decay behavior and may not match the expectation of full vesting completion.

2. **Incorrect Use of `unlockedPercent`**
   The `unlockedPercent` variable increases by a flat 5 on every unlock after the first. Since the transferred amount is always calculated from the **current** balance (not the original allocation), `unlockedPercent` **does not represent the actual total percentage of the originally vested tokens released**. As a result, `unlockedPercent` can **exceed 100%**, which may mislead users or downstream systems relying on it for accounting.

3. **Potential Accounting Inconsistency**
   Without tracking the original total vesting amount, it is impossible to determine how much of the vesting has been completed or remains. This limits transparency and may cause confusion or integration issues.

```
    function checkUnlock() external nonReentrant {
        require(address(token) != address(0), "Token address not set");

        (, int256 price, , ,) = priceFeed.latestRoundData();
        require(price >= int256(currentPriceTarget), "Price below target");
        require(lastUnlockTime == 0 || block.timestamp >= lastUnlockTime + sustainDuration, "Sustain duration not met");

        uint256 balance = token.balanceOf(address(this));
        require(balance > 0, "No tokens left");

        uint256 toUnlock;
        if (unlockedPercent == 0) {
            // First unlock: 10%
            toUnlock = (balance * 10) / 100;
            unlockedPercent = 10;
        } else {
            // Subsequent unlocks: 5% each time and increase target by 30%
            toUnlock = (balance * 5) / 100;
            unlockedPercent += 5;
            currentPriceTarget = (currentPriceTarget * 130) / 100;
        }

        lastUnlockTime = block.timestamp;
        require(token.transfer(beneficiary, toUnlock), "Transfer failed");
        emit TokensUnlocked(beneficiary, toUnlock, unlockedPercent, currentPriceTarget);
    }
```

Recommendation

We recommend reviewing and potentially redesigning the vesting logic.

Alleviation

[INVI TOKEN Team, 04/22/2025]: The team heeded the advice and resolved the issue in commit: 22477495c7aced18b875e33ee997b2523d82fa23.

ITA-06 | COMPILATION ERROR IN `updateOracle()` FUNCTION

Category	Severity	Location	Status
Coding Issue	● Medium	InvincibleVesting.sol (commit:224774): 85	● Resolved

Description

The `updateOracle()` function in the `InvincibleVesting` contract contains a compilation error due to the use of an undeclared identifier `OracleUpdated`.

```
function updateOracle(address _newOracle) external onlyOwner {
    require(_newOracle != address(0), "Zero oracle address");
    emit OracleUpdated(address(priceFeed), _newOracle); // ← Compiler Error:
Undeclared identifier
    // can't change immutable, so this function is illustrative only
    // for a real upgrade you'd use a proxy or a new contract
}
```

Recommendation

It is recommended to revise the code.

Alleviation

[INVI TOKEN Team, 04/24/2025]: The team heeded the advice and resolved the issue in commit: 22292b2b21098b2afe44dddcf62c75a963e8dcb0.

OPTIMIZATIONS | INVI TOKEN - AUDIT

ID	Title	Category	Severity	Status
ITA-01	Variables That Could Be Declared As Immutable	Gas Optimization	Optimization	● Resolved

ITA-01 | VARIABLES THAT COULD BE DECLARED AS IMMUTABLE

Category	Severity	Location	Status
Gas Optimization	● Optimization	InvincibleToken.sol (pre): 12	● Resolved

Description

The linked variables assigned in the constructor can be declared as `immutable`. Immutable state variables can be assigned during contract creation but will remain constant throughout the lifetime of a deployed contract. A big advantage of immutable variables is that reading them is significantly cheaper than reading from regular state variables since they will not be stored in storage.

Recommendation

We recommend declaring these variables as immutable.

Alleviation

[INVI TOKEN Team, 04/22/2025]: The team heeded the advice and resolved the issue in commit: 22477495c7aced18b875e33ee997b2523d82fa23.

APPENDIX | INVI TOKEN - AUDIT

Finding Categories

Categories	Description
Gas Optimization	Gas Optimization findings do not affect the functionality of the code but generate different, more optimal EVM opcodes resulting in a reduction on the total gas cost of a transaction.
Coding Issue	Coding Issue findings are about general code quality including, but not limited to, coding mistakes, compile errors, and performance issues.
Centralization	Centralization findings detail the design choices of designating privileged roles or other centralized controls over the code.
Design Issue	Design Issue findings indicate general issues at the design level beyond program logic that are not covered by other finding categories.

Checksum Calculation Method

The "Checksum" field in the "Audit Scope" section is calculated as the SHA-256 (Secure Hash Algorithm 2 with digest size of 256 bits) digest of the content of each file hosted in the listed source repository under the specified commit.

The result is hexadecimal encoded and is the same as the output of the Linux "sha256sum" command against the target file.

DISCLAIMER | CERTIK

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to you ("Customer" or the "Company") in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes, nor may copies be delivered to any other person other than the Company, without CertiK's prior written consent in each instance.

This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts CertiK to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. CertiK's position is that each company and individual are responsible for their own due diligence and continuous security. CertiK's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

The assessment services provided by CertiK is subject to dependencies and under continuing development. You agree that your access and/or use, including but not limited to any services, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third-parties.

ALL SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF ARE PROVIDED "AS IS" AND "AS AVAILABLE" AND WITH ALL FAULTS AND DEFECTS WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, CERTIK HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, CERTIK SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM COURSE OF DEALING, USAGE, OR TRADE PRACTICE. WITHOUT LIMITING THE FOREGOING, CERTIK MAKES NO WARRANTY OF ANY KIND THAT THE SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET CUSTOMER'S OR ANY OTHER PERSON'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULT, BE COMPATIBLE OR WORK WITH ANY SOFTWARE, SYSTEM, OR OTHER SERVICES, OR BE SECURE, ACCURATE, COMPLETE, FREE OF HARMFUL CODE, OR ERROR-FREE. WITHOUT LIMITATION TO THE FOREGOING, CERTIK PROVIDES NO WARRANTY OR

UNDERTAKING, AND MAKES NO REPRESENTATION OF ANY KIND THAT THE SERVICE WILL MEET CUSTOMER'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULTS, BE COMPATIBLE OR WORK WITH ANY OTHER SOFTWARE, APPLICATIONS, SYSTEMS OR SERVICES, OPERATE WITHOUT INTERRUPTION, MEET ANY PERFORMANCE OR RELIABILITY STANDARDS OR BE ERROR FREE OR THAT ANY ERRORS OR DEFECTS CAN OR WILL BE CORRECTED.

WITHOUT LIMITING THE FOREGOING, NEITHER CERTIK NOR ANY OF CERTIK'S AGENTS MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND, EXPRESS OR IMPLIED AS TO THE ACCURACY, RELIABILITY, OR CURRENCY OF ANY INFORMATION OR CONTENT PROVIDED THROUGH THE SERVICE. CERTIK WILL ASSUME NO LIABILITY OR RESPONSIBILITY FOR (I) ANY ERRORS, MISTAKES, OR INACCURACIES OF CONTENT AND MATERIALS OR FOR ANY LOSS OR DAMAGE OF ANY KIND INCURRED AS A RESULT OF THE USE OF ANY CONTENT, OR (II) ANY PERSONAL INJURY OR PROPERTY DAMAGE, OF ANY NATURE WHATSOEVER, RESULTING FROM CUSTOMER'S ACCESS TO OR USE OF THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS.

ALL THIRD-PARTY MATERIALS ARE PROVIDED "AS IS" AND ANY REPRESENTATION OR WARRANTY OF OR CONCERNING ANY THIRD-PARTY MATERIALS IS STRICTLY BETWEEN CUSTOMER AND THE THIRD-PARTY OWNER OR DISTRIBUTOR OF THE THIRD-PARTY MATERIALS.

THE SERVICES, ASSESSMENT REPORT, AND ANY OTHER MATERIALS HEREUNDER ARE SOLELY PROVIDED TO CUSTOMER AND MAY NOT BE RELIED ON BY ANY OTHER PERSON OR FOR ANY PURPOSE NOT SPECIFICALLY IDENTIFIED IN THIS AGREEMENT, NOR MAY COPIES BE DELIVERED TO, ANY OTHER PERSON WITHOUT CERTIK'S PRIOR WRITTEN CONSENT IN EACH INSTANCE.

NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST CERTIK WITH RESPECT TO SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS.

THE REPRESENTATIONS AND WARRANTIES OF CERTIK CONTAINED IN THIS AGREEMENT ARE SOLELY FOR THE BENEFIT OF CUSTOMER. ACCORDINGLY, NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH REPRESENTATIONS AND WARRANTIES AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST CERTIK WITH RESPECT TO SUCH REPRESENTATIONS OR WARRANTIES OR ANY MATTER SUBJECT TO OR RESULTING IN INDEMNIFICATION UNDER THIS AGREEMENT OR OTHERWISE.

FOR AVOIDANCE OF DOUBT, THE SERVICES, INCLUDING ANY ASSOCIATED ASSESSMENT REPORTS OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL, REGULATORY, OR OTHER ADVICE.

Elevating Your Entire Web3 Journey

Founded in 2017 by leading academics in the field of Computer Science from both Yale and Columbia University, CertiK is a leading blockchain security company that serves to verify the security and correctness of smart contracts and blockchain-based protocols. Through the utilization of our world-class technical expertise, alongside our proprietary, innovative tech, we’re able to support the success of our clients with best-in-class security, all whilst realizing our overarching vision; provable trust for all throughout all facets of blockchain.