CERTIK

Security Assessment

Panther ZKP Token

Nov 16th, 2021

Table of Contents

Summary

This report has been prepared for Panther Protocol to discover issues and vulnerabilities in the source code of the Panther ZKP Token project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Static Analysis and Manual Review 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:

- 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.

Overview

Project Summary

Project Name	Panther ZKP Token
Platform	other
Language	Solidity
Codebase	https://github.com/pantherprotocol/zkp-token/tree/master/contracts
Commit	• f1e9d857dbd1660d90f1f029511f93417896d792 • ed7262b28e35f561cf35c66b4ac1bf60690d87a4

Audit Summary

Delivery Date	Nov 16, 2021
Audit Methodology	Static Analysis, Manual Review
Key Components	

Vulnerability Summary

Vulnerability Level	Total	Pending	Declined	Acknowledged	Partially Resolved	Resolved
Critical	0	0	0	0	0	0
Major	1	0	0	1	0	0
Medium	0	0	0	0	0	0
Minor	0	0	0	0	0	0
Informational	1	0	0	0	0	1
Discussion	0	0	0	0	0	0

Audit Scope

ID	File	SHA256 Checksum
CCK	interfaces/Constants.sol	262cddb01adf7bff3c3f582e0fb1ef33d8989786cd5c4ad60688bb2a8bb93e8a
ZKP	ZKPToken.sol	dae1edcf593ba4e946cce0d860bbd2663b6b65b12865bf6a22af670693bbac89

Understandings

Overview

The Panther Protocol is a blockchain network with a focus on privacy while also providing compliance tools through zero-knowledge proofs. In this report, we looked at the Panther Protocol's ZKP token as well as their implementation of vesting pools. This includes how stakeholders interact with the vesting pool and the implementation of a vesting pool's wallet.

Dependencies

We assume the contracts `PoolStakes`, `VestingPools`, `ZKPToken`, `Constants`, `Claimable`, `DefaultOwnable`, `TokenAddress`, `VestingPoolsAddress`, `DefaultOwnerAddress`, `ProxyFactory`, and `SafeUints` are deployed successfully and triggered correctly within the protocol.

There are a few depending injection contracts or addresses in the current project:

- `DefaultOwnerAddress`, `TokenAddress`, and `VestingPoolsAddress` for the contract `PoolStakes`;
- `TokenAddress` for the contract `VestingPools`;
- `_minter` for the contract `ZKPToken`.

We assume these contracts or addresses are valid and non-vulnerable actors and implementing proper logic to collaborate with the current project.

Privileged Functions

In the contract `PoolStakes`, the roles `_owner` and `_defaultOwner` have the authority over the following functions:

- `PoolStakes.addStakes()`, which adds stakeholders along with their allocations to a proxy;
- `PoolStakes.massWithdraw()`, which sends tokens to stakeholders;
- `PoolStakes.claimErc20()`, which sends the contract's extra tokens to an address;
- `PoolStakes.removeContract()`, which destroys a proxy version of the contract under the conditions that all stakes have been paid and the contract does not contain any vested Tokens;
- `DefaultOwnable.transferOwnership()`, which transfers the `_owner` role to a designated address.

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

- `VestingPools.addVestingPools()`, which adds a vesting pool and its associated wallet;
- `VestingPools.updatePoolTime()`, which changes the start time and vesting duration of a vesting pool;
- `VestingPools.claimErc20()`, which sends ERC20 tokens or unvested tokens to an address;

- `VestingPools.removeContract()`, which destroys the `VestingPools` contract under the condition that all allocated tokens have been vested;
- `Ownable.renounceOwnership()`, which disables all functions with the `onlyOwner` modifier;
- `Ownable.transferOwnership()`, which transfers the `_owner` role to a different address.

In addition, the wallet associated to the vesting pool has the authority over the following functions:

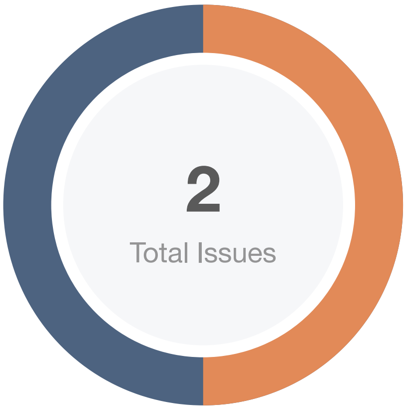
- `VestingPools.release()`, which sends tokens allocated in the vesting pool to the wallet;
- `VestingPools.releaseTo()`, which sends tokens allocated in the vesting pool to a chosen address;
- `VestingPools.updatePoolWallet()`, which changes the address of the wallet for that vesting pool.

In the contract `ZKPToken`, the role `minter` has the authority over the following functions:

- `ZKPToken.mint()`, which mints new ZKP tokens;
- `ZKPToken.setMinter()`, which sets the address for the `minter` role.

To improve the trustworthiness of the project, dynamic runtime updates in the project should be notified to the community. Any plan to invoke the aforementioned functions should be also considered to move to the execution queue of the `Timelock` contract.

Findings



- **Critical** **0** (0.00%)
- **Major** **1** (50.00%)
- **Medium** **0** (0.00%)
- **Minor** **0** (0.00%)
- **Informational** **1** (50.00%)
- **Discussion** **0** (0.00%)

ID	Title	Category	Severity	Status
GLOBAL-01	Unlocked Compiler Version	Language Specific	● Informational	✓ Resolved
ZKP-01	Centralization Risk	**Centralization / Privilege**	● **Major**	ⓘ Acknowledged

GLOBAL-01 | Unlocked Compiler Version

Category	Severity	Location	Status
Language Specific	Informational	Global	Resolved

Description

The contract has unlocked compiler version. An unlocked compiler version in the source code of the contract permits the user to compile it at or above a particular version. This, in turn, leads to differences in the generated bytecode between compilations due to differing compiler version numbers. This can lead to an ambiguity when debugging as compiler specific bugs may occur in the codebase that would be hard to identify over a span of multiple compiler versions rather than a specific one.

Recommendation

We advise that the compiler version is instead locked at the lowest version possible that the contract can be compiled at. For example, for version `v0.6.2` the contract should contain the following line:

```
pragma solidity 0.6.2;
```

Alleviation

[Panther Team]: Exact compiler version (8.4) is fixed in the hardhat.config.ts.

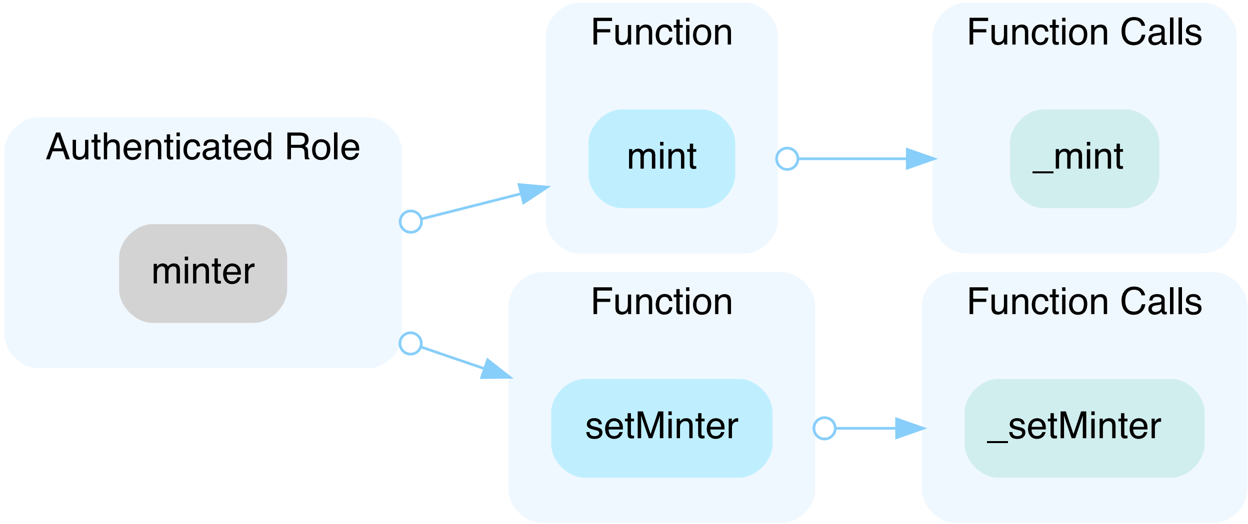
ZKP-01 | Centralization Risk

Category	Severity	Location	Status
Centralization / Privilege	Major	projects/panther/contracts/ZKPToken.sol (9a05001): 24, 33	Acknowledged

Description

In the contract `ZKPToken`, the role `minter` has the authority over the following functions:

- `ZKPToken.mint()`, the minter can mints arbitrary amount of ZKP tokens;
- `ZKPToken.setMinter()`, the minter can assign arbitrary the address as the `minter` role.



Any compromise to the `minter` account may allow the hacker to take advantage of this and arbitrarily mint new tokens or prevent new tokens from being minted.

Recommendation

We advise the client to carefully manage the `minter` account's private key to avoid any potential risks of being hacked. In general, we strongly recommend centralized privileges or roles in the protocol to be improved via a decentralized mechanism or smart-contract-based accounts with enhanced security practices, e.g., Multisignature wallets.

Here are some feasible suggestions that would also mitigate this risk in the short-term and long-term:

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

- Introduction of a DAO/governance/voting module to increase transparency and user involvement.

Alleviation

[Panther Team]: Contracts logic (ZKPToken has a single minter, VestingPools contract mints $ZKP), the Deployment Plan (./deploymentPlan.README.md), and the Contracts Hierarchy diagram (docs/ZKP-contracts-hierarchy.png) provides for the VestingPools contract being the only ZPToken.minter.

[Certik]: The auditors agree that, if the `minter` is the `VestingPool` contract, there will not be risks on the `minter` account's private key. However, considering the auditors do not know if the deployment will proceed correctly, the status of this issue will be updated after contract deployment upon request.

Appendix

Finding Categories

Centralization / Privilege

Centralization / Privilege findings refer to either feature logic or implementation of components that act against the nature of decentralization, such as explicit ownership or specialized access roles in combination with a mechanism to relocate funds.

Language Specific

Language Specific findings are issues that would only arise within Solidity, i.e. incorrect usage of private or delete.

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

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.

About

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.