RAPID LABS

SMART CONTRACT SECURITY AUDIT

Final report **Plan: Simple**

A51 Finance

January 2024

RL@RAPIDLABS.FINANCE

RAPIDLABS.FINANCE

TABLE OF CONTENTS

INTRODUCTION

The report has been prepared for A51 Finance.

Pilot is an ERC20 token with additional functionality.

The token is burnable.

The token is mintable. Accounts with a minter role can mint new tokens.

The contract has permit extension.

Name	A51 Finance
Audit date	2024-01-12 - 2024-01-12
Language	Solidity
Platform	Ethereum

ANALYZED CONTRACTS

Name	Address
Pilot	0x37c997b35c619c21323f3518b9357914e8b99525

AUDIT PROCESS

Our audit structure consists of two stages:

Auto-analysis

- Our automated tools allow us to scan smart contract code and find potential issues
- We hand pick and verify all the issues found by the tools

Expert audit

- Manual analysis of potential issues and vulnerabilities
- Contract code is reviewed thoroughly

KNOWN ISSUES CHECKED

Title	Result
Unencrypted Private Data On-Chain	✓ passed
Code With No Effects	✓ passed
Message call with hardcoded gas amount	✓ passed
Typographical Error	✓ passed
DoS With Block Gas Limit	✓ passed
Presence of unused variables	✓ passed
Incorrect Inheritance Order	✓ passed
Requirement Violation	✓ passed
Weak Sources of Randomness from Chain Attributes	✓ passed
Shadowing State Variables	✓ passed
Incorrect Constructor Name	✓ passed

Block values as a proxy for time	✓ passed
Authorization through tx.origin	✓ passed
DoS with Failed Call	✓ passed
Delegatecall to Untrusted Callee	✓ passed
Use of Deprecated Solidity Functions	✓ passed
Assert Violation	✓ passed
State Variable Default Visibility	✓ passed
Reentrancy	✓ passed
Unprotected SELFDESTRUCT Instruction	✓ passed
Unprotected Ether Withdrawal	✓ passed
Unchecked Call Return Value	✓ passed
Floating Pragma	✓ passed
Outdated Compiler Version	✓ passed
Integer Overflow and Underflow	✓ passed
Function Default Visibility	✓ passed

ISSUE CLASSIFICATION

High risk	Issues leading to assets theft, locking or any other loss of assets or leading to contract malfunctioning.
Medium risk	Issues that can trigger a contract failure of malfunctioning.
Low risk	Issues that do now affect contract functionality. For example, unoptimised gas usage, outdated or unused code, code style violations, etc.

ISSUES

High risk issues

No issues were found

Medium risk issues

No issues were found

Low risk issues

No issues were found

CONCLUSION

A51 Finance Pilot contract was audited. No risk issues were found.

The token contract allows privileged accounts to mint new tokens.

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 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 without RapidLabs prior written consent.

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 RapidLabs 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.

AUTOMATED ANALYSIS

```
INFO:Detectors:
Pilot.updateMinter(address) (contracts/Token.sol#588-593) should emit an event
for:
        - _minter = newMinter (contracts/Token.sol#591)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-
events-access-control
INFO:Detectors:
Pilot.constructor(address,address[],uint256[])._timelock (contracts/Token.sol#576)
lacks a zero-check on :
                - timelock = _timelock (contracts/Token.sol#581)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-
zero-address-validation
INFO:Detectors:
Pilot.permit(address,address,uint256,uint256,uint8,bytes32,bytes32) (contracts/
Token.sol#618-634) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(deadline >= block.timestamp,PILOT:: AUTH_EXPIRED)
(contracts/Token.sol#627)
Pilot.transferWithAuthorization(address,address,uint256,uint256,uint256,bytes32,uin
t8,bytes32,bytes32) (contracts/Token.sol#649-682) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(block.timestamp > validAfter,PILOT::
AUTH_NOT_YET_VALID) (contracts/Token.sol#660)
        - require(bool,string)(block.timestamp < validBefore,PILOT:: AUTH_EXPIRED)
(contracts/Token.sol#661)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-
timestamp
INFO:Detectors:
Pilot.getChainId() (contracts/Token.sol#643-647) uses assembly
        - INLINE ASM (contracts/Token.sol#644-646)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-
usage
INFO:Detectors:
Context._msgData() (contracts/Token.sol#133-135) is never used and should be
removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
```

```
Pragma version>=0.8.6 (contracts/Token.sol#2) allows old versions
solc-0.8.6 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
Pilot.timelock (contracts/Token.sol#533) should be immutable
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-immutable
INFO:Slither:. analyzed (6 contracts with 85 detectors), 9 result(s) found
```

RAPID LABS

YOUR PROJECT SECURED

RL@RAPIDLABS.FINANCE

RAPIDLABS.FINANCE