KISHIELD

Security Audit

MetaPenny Token

February 25, 2022

Table of Contents

Audit Summary

This report has been prepared for MetaPenny Token on the Binance Smart Chain network. KISHIELD provides both client-centered and user-centered examination of the smart contracts and their current status when applicable. This report represents the security assessment made to find issues and vulnerabilities on the source code along with the current liquidity and token holder statistics of the protocol.

A comprehensive examination has been performed, utilizing Cross Referencing, Static Analysis, In-House Security Tools, and line-by-line Manual Review.

The auditing process pays special attention to the following considerations:

- Ensuring contract logic meets the specifications and intentions of the client without exposing the user's funds to risk.
- Testing the smart contracts against both common and uncommon attack vectors.
- Inspecting liquidity and holders statistics to inform the current status to both users and client when applicable.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Verifying contract functions that allow trusted and/or untrusted actors to mint, lock, pause, and transfer assets.
- Thorough line-by-line manual review of the entire codebase by industry experts.

Project Overview

Token Summary

Parameter	Result
Address	0x8E6A704A4e55f402a0e63cEe6cAbb0B002204603
Name	MetaPenny
Token Tracker	MetaPenny (MetaPenny)
Decimals	18
Supply	100,000,000,000
Platform	Binance Smart Chain
compiler	v0.8.4+commit.c7e474f2
Optimization	Yes with 200 runs
LicenseType	MIT
Language	Solidity
Codebase	https://bscscan.com/address/0x8e6a704a4e55f402a0e63cee6cabb0b002204603
Url	https://pennyinu.com/

Main Contract Assessed

Name	Contract	Live
MetaPenny	0x8E6A704A4e55f402a0e63cEe6cAbb0B002204603	Yes

Smart Contract Vulnerability Checks

Vulnerability	Automatic Scan	Manual Scan	Result
Unencrypted Private Data On-Chain	Complete	Complete	Low / No Risk
Code With No Effects	Complete	Complete	Low / No Risk
Message call with hardcoded gas amount	Complete	Complete	Low / No Risk
Hash Collisions With Multiple Variable Length Arguments	Complete	Complete	Low / No Risk
Unexpected Ether balance	Complete	Complete	Low / No Risk
Presence of unused variables	Complete	Complete	Low / No Risk
Right-To-Left-Override control character (U+202E)	Complete	Complete	Low / No Risk
Typographical Error	Complete	Complete	Low / No Risk
DoS With Block Gas Limit	Complete	Complete	Low / No Risk
Arbitrary Jump with Function Type Variable	Complete	Complete	Low / No Risk
Insufficient Gas Griefing	Complete	Complete	Low / No Risk
Incorrect Inheritance Order	Complete	Complete	Low / No Risk
Write to Arbitrary Storage Location	Complete	Complete	Low / No Risk
Requirement Violation	Complete	Complete	Low / No Risk
Missing Protection against Signature Replay Attacks	Complete	Complete	Low / No Risk
Weak Sources of Randomness from Chain Attributes	Complete	Complete	Low / No Risk

Vulnerability	Automatic Scan	Manual Scan	Result
Authorization through tx.origin	Complete	Complete	Low / No Risk
Delegatecall to Untrusted Callee	Complete	Complete	Low / No Risk
Use of Deprecated Solidity Functions	Complete	Complete	Low / No Risk
Assert Violation	Complete	Complete	Low / No Risk
Reentrancy	Complete	Complete	Low / No Risk
Unprotected SELFDESTRUCT Instruction	Complete	Complete	Low / No Risk
Unprotected Ether Withdrawal	Complete	Complete	Low / No Risk
Unchecked Call Return Value	Complete	Complete	Low / No Risk
Outdated Compiler Version	Complete	Complete	Low / No Risk
Integer Overflow and Underflow	Complete	Complete	Low / No Risk
Function Default Visibility	Complete	Complete	Low / No Risk

Contract Ownership

The contract ownership of MetaPenny is not currently renounced. The ownership of the contract grants special powers to the protocol creators, making them the sole addresses that can call sensible ownable functions that may alter the state of the protocol.

The current owner is the address 0x5F4F9A3EaC277811D710bc4b44Fdde2722FB1d81 which can be viewed from:
HERE

The owner wallet has the power to call the functions displayed on the priviliged functions chart below, if the owner wallet is compromised this privileges could be exploited.

We recommend the team to renounce ownership at the right timing if possible, or gradually migrate to a timelock with governing functionalities in respect of transparency and safety considerations.

Priviliged Functions (onlyOwner)

Function Name	Parameters	Visibility
renounceOwnership	none	public
transferOwnership	address newOwner	public
__Context_init	none	internal
__Context_init_unchained	none	internal
__ERC20_init	none	internal
__ERC20_init_unchained	none	internal
__Ownable_init	none	internal
__Ownable_init_unchained	none	internal
renounceOwnership	none	public
transferOwnership	address newOwner	public
__DividendPayingToken_init	none	internal
distributeCAKEDividends	uint256 amount	public
initialize	none	external
excludeFromDividends	address account	external
updateClaimWait	uint256 newClaimWait	external
updateMinimumTokenBalanceForDividends	uint256 amount	external

Function Name	Parameters	Visibility
setBalance	address account, uint256 newBalance	external
processAccount	address account, bool automatic	public
setSwapTokensAtAmount	uint256 amount	external
updateDividendTracker	address newAddress	public
updateUniswapV2Router	address newAddress	public
excludeFromFees	address account, bool excluded	public
excludeMultipleAccountsFromFees	calldata accounts, bool excluded	public
setMarketingWallet	address wallet	external
setTokenRewardsFee	uint256 value	external
setLiquiditFee	uint256 value	external
setMarketingFee	uint256 value	external
setAutomatedMarketMakerPair	address pair, bool value	public
updateGasForProcessing	uint256 newValue	public
updateClaimWait	uint256 claimWait	external
updateMinimumTokenBalanceForDividends	uint256 amount	external
excludeFromDividends	address account	external

Important Notes To The Users:

- The total sell/buy fees can not be larger than 25%
- The owner can not stop Trading
- The transfer function is implemented correctly
- The owner can set a new address for the divident tracker
- The owner can migrate the token to a new router
- The owner can include/exclude addresses from fees and dividents.
- The owner can change the fees realated to buy and sell
- The owner can include/exclude addresses from fees and dividents.
- The owner can update the min token balance required to recieve dividends
- The owner can add to the balance of any address by calling setBalance() with a non excluded address and the newBalance being higher than minimumTokenBalanceForDividends, this triggers a mint() inside _setBalance()
- The _transfer() in DividendPayingToken contract has dead code after require(false)
- No high-risk Exploits Were Found in the Source Code.

Audit Passed

Statistics

Liquidity Info

Parameter	Result
Pair Address	0x4DaB9142190D013A03Df8b5cA0e01310a2e3956C
MetaPenny Reserves	0.00 MetaPenny
BNB Reserves	0.00 BNB
Liquidity Value	$0 USD

Token (MetaPenny) Holders Info

Parameter	Result
MetaPenny Percentage Burnt	0.00%
MetaPenny Amount Burnt	0 MetaPenny
Top 10 Percentage Own	100.00%
Top 10 Amount Owned	100,000,000,000 MetaPenny
Top 10 Aprox Value	$NaN USD

LP (MetaPenny/BNB) Holders Info

Parameter	Result
MetaPenny/BNB % Burnt	0.00%
MetaPenny/BNB Amount Burnt	0 MetaPenny
Top 10 Percentage Owned	0.00%
Top 10 Amount Owned	0 MetaPenny
Locked Tokens Percentage	0.00%
Locked Tokens Amount	0 MetaPenny

* All the data diplayed above was taken on-chain at block 15578493
* The tokens on industry-standard burn wallets are not included on the top 10 wallets calculations

Liquidity Ownership

The token does not have liquity at the moment of the audit, block 14772401

KISHIELD

Disclaimer

KISHIELD has conducted an independent audit to verify the integrity of and highlight any vulnerabilities or errors, intentional or unintentional, that may be present in the codes that were provided for the scope of this audit. This audit report does not constitute agreement, acceptance or advocation for the Project that was audited, and users relying on this audit report should not consider this as having any merit for financial advice in any shape, form or nature. The contracts audited do not account for any economic developments that may be pursued by the Project in question, and that the veracity of the findings thus presented in this report relate solely to the proficiency, competence, aptitude and discretion of our independent auditors, who make no guarantees nor assurance that the contracts are completely free of exploits, bugs, vulnerabilities or deprecation of technologies.

All information provided in this report does not constitute financial or investment advice, nor should it be used to signal that any persons reading this report should invest their funds without sufficient individual due diligence regardless of the findings presented in this report. Information is provided ‘as is’, and KISHIELD is under no covenant to the completeness, accuracy or solidity of the contracts audited. In no event will KISHIELD or its partners, employees, agents or parties related to the provision of this audit report be liable to any parties for, or lack thereof, decisions and/or actions with regards to the information provided in this audit report.

The assessment services provided by KISHIELD 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.