expelee

Building the Futuristic Blockchain Ecosystem

SECURITY AUDIT REPORT

PERB

expelee

TABLE OF CONTENTS

OVERVIEW

The Expelee team has performed a line-by-line manual analysis and automated review of the smart contract. The smart contract was analysed mainly for common smart contract vulnerabilities, exploits, and manipulation hacks. According to the smart contract audit:

Audit Result	**Passed**
KYC Verification	**No**
Audit Date	**7 May 2023**

CONTRACT DETAILS

Token Name: BasedPepeEth

Symbol: PERB

Network: ERC

Language: Solidity

Total Supply: 1,000,000,000,000

Contract SHA-256 Checksum:
474c2309a12cf894c6a821ae4fad64378ef1efc4

Owner's Wallet: ---

Deployer's Wallet: ---

Testnet:
https://testnet.bscscan.com/token/0xEFAE1EdAFC6926cA2D281Aa810074ac502f6DDd7#readContract

OWNER PRIVILEGES

- Contract owner is able to set buy/sell tax each one more than 12%
- Contract owner is able to set tarnsfer tax (0% tax)
- Contract owner is not able to blacklist an arbitrary wallet
- Contract owner is not able to set limit for buy/sell/transfer/holding amounts
- Contract owner is not able to mint new tokens
- Contract owner is not able to disable trades

TESTNET VERSION

Adding Liquidity ✓
Tx:
https://testnet.bscscan.com/tx/0xb2ca9e4c9c4ae7e611b363ea1a7c7454d6f2ac17f8526793f3fe1639485dcc4e

==========

Buying from a fee excluded wallet ✓
Tx (0% tax):
https://testnet.bscscan.com/tx/0xe01821791e1531de9aceeed2410a7847b2c79090daea8bf61b2e5886cb406c04

==========

Selling from a fee excluded wallet ✓
Tx (0% tax):
https://testnet.bscscan.com/tx/0x9326a81b912dd68fd1be965c4a79840616ba91444aa26a5eac2886e5b1dd1301

==========

Transferring using a fee excluded wallet ✓
Tx (0% tax):
https://testnet.bscscan.com/tx/0x46f472ed49aa12adb7e79ee912c664fbf32709672f081c4cae3e007a1aaac581

==========

TESTNET VERSION

Buying from a regular wallet ✔
Tx (up to 12% tax):
https://testnet.bscscan.com/tx/0x0dd6fff59697910a88f8f8d7bc4514fee695673f84ddd8d492909a0e21b6b490

==========

Selling from a regular wallet ✔
Tx (up to 12% tax):
https://testnet.bscscan.com/tx/0x49ff30452b8b06d660706338f9fc32cee969ad9a5de713f5f6fdf090cbbae0ab

==========

Transferring a regular wallet ✔
Tx (0% tax):
https://testnet.bscscan.com/tx/0x0da68809ff8fa6ab4c751f79eb26184480cee844c4bd0de681fac03730751306

==========

Internal swap (ETH to marketing wallet, Auto liquidity, rewards) ✔
Tx :
https://testnet.bscscan.com/tx/0x49ff30452b8b06d660706338f9fc32cee969ad9a5de713f5f6fdf090cbbae0ab

AUDIT METHODOLOGY

Audit Details

Our comprehensive audit report provides a full overview of the audited system's architecture, smart contract codebase, and details on any vulnerabilities found within the system.

Audit Goals

The audit goal is to ensure that the project is built to protect investors and users, preventing potentially catastrophic vulnerabilities after launch , that lead to scams and rugpulls.

Code Quality

Our analysis includes both automatic tests and manual code analysis for the following aspects:

- Exploits
- Back-doors
- Vulnerability
- Accuracy
- Readability

Tools

- DE
- Open Zeppelin
- Code Analyzer
- Solidity Code
- Compiler
- Hardhat

VULNERABILITY CHECKS

Design Logic	Passed
Compiler warnings	Passed
Private user data leaks	Passed
Timestamps dependence	Passed
Integer overflow and underflow	Passed
Race conditions & reentrancy. Cross-function race conditions	Passed
Possible delays in data delivery	Passed
Oracle calls	Passed
Front Running	Passed
DoS with Revert	Passed
DoS with block gas limit	Passed
Methods execution permissions	Passed
Economy model	Passed
Impact of the exchange rate on the logic	Passed
Malicious event log	Passed
Scoping and declarations	Passed
Uninitialized storage pointers	Passed
Arithmetic accuracy	Passed
Cross-function race conditions	Passed
Safe Zepplin module	Passed

RISK CLASSIFICATION

When performing smart contract audits, our specialists look for known vulnerabilities as well as logical and acces control issues within the code. The exploitation of these issues by malicious actors may cause serious financial damage to projects that failed to get an audit in time. We categorize these vulnerabilities by the following levels:

High Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Medium Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Low Risk

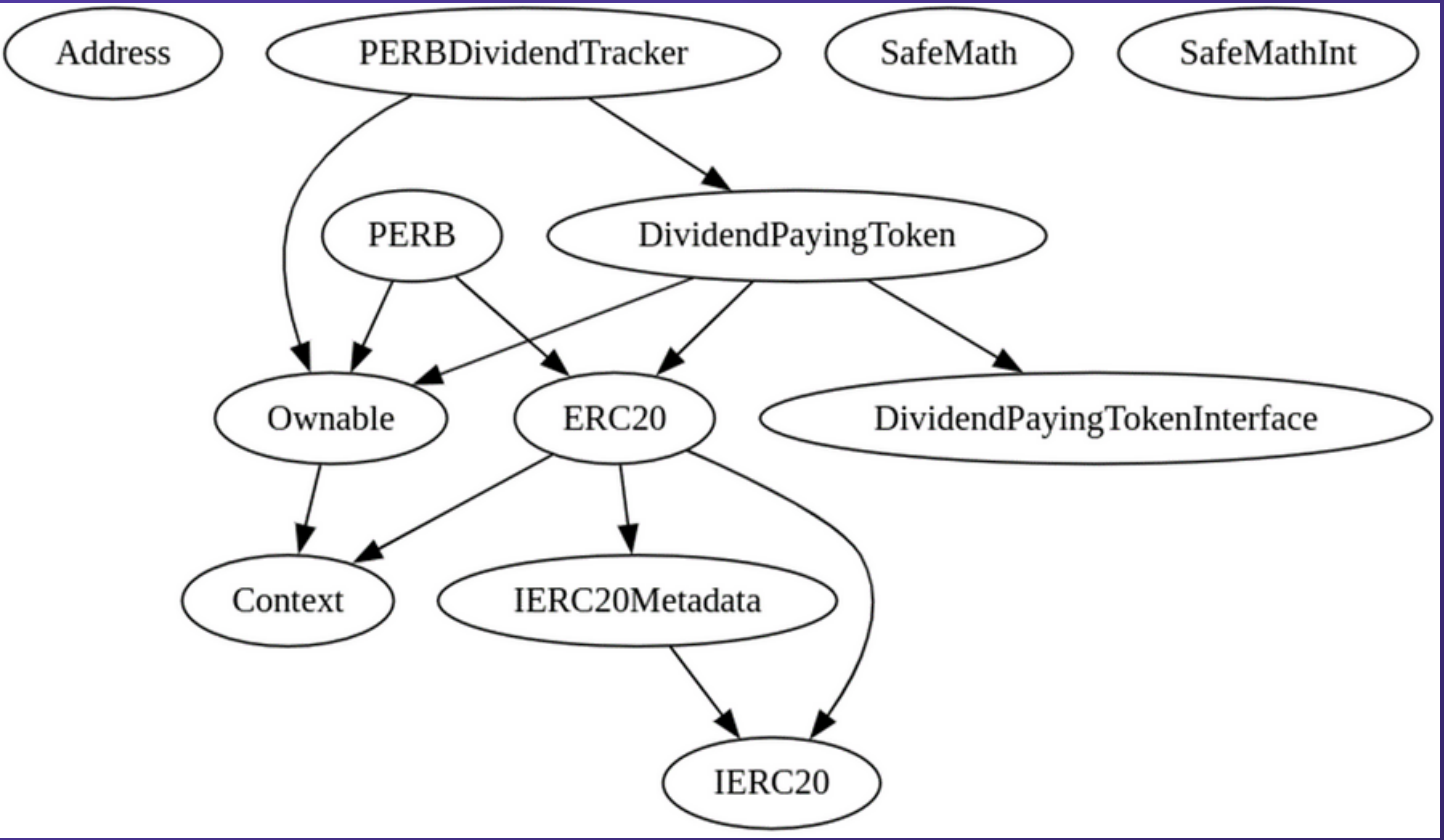
Issues on this level are minor details and warning that can remain unfixed.

Informational

Issues on this level are minor details and warning that can remain unfixed.

expelee

INHERITANCE TREES



expelee

FUNCTION DETAILS

Symbol	Meaning
🔴	Function can modify state
💵	Function is payable

Contract Assesment

Contract	Type	Bases		
└	**Function Name**	**Visibility**	**Mutability**	**Modifiers**
Address	Library			
└	sendValue	Internal 🔒	🔴	
PERB	Implementation	ERC20, Ownable		
└	<Constructor>	Public ❗	🔴	ERC20
└	<Receive Ether>	External ❗	💵	NO ❗
└	processDividendTracker	External ❗	🔴	NO ❗
└	setBuyTaxes	External ❗	🔴	onlyOwner
└	setSellTaxes	External ❗	🔴	onlyOwner
└	claim	External ❗	🔴	NO ❗
└	withdrawToken	External ❗	🔴	onlyOwner
└	withdrawETH	External ❗	🔴	onlyOwner
└	excludeFromFees	Public ❗	🔴	onlyOwner
└	excludeMultipleAccountsFromFees	Public ❗	🔴	onlyOwner
└	excludeFromDividends	External ❗	🔴	onlyOwner
└	setMarketingWallet	External ❗	🔴	onlyOwner
└	setSwapTokensAtAmount	External ❗	🔴	onlyOwner
└	setSwapEnabled	External ❗	🔴	onlyOwner
└	setMinBalanceForDividends	External ❗	🔴	onlyOwner
└	_setAutomatedMarketMakerPair	Private 🔐	🔴	

expelee

FUNCTION DETAILS

└	setGasForProcessing	External ❗	🔴	onlyOwner
└	setClaimWait	External ❗	🔴	onlyOwner
└	getClaimWait	External ❗	NO ❗	
└	getTotalDividendsDistributed	External ❗	NO ❗	
└	isExcludedFromFees	Public ❗	NO ❗	
└	withdrawableDividendOf	Public ❗	NO ❗	
└	getCurrentRewardToken	External ❗	NO ❗	
└	dividendTokenBalanceOf	Public ❗	NO ❗	
└	getAccountDividendsInfo	External ❗	NO ❗	
└	getAccountDividendsInfoAtIndex	External ❗	NO ❗	
└	getLastProcessedIndex	External ❗	NO ❗	
└	getNumberOfDividendTokenHolders	External ❗	NO ❗	
└	_transfer	Internal 🔒	🔴	
└	swapAndLiquify	Private 🔐	🔴	
└	swapTokensForBNB	Private 🔐	🔴	
└	addLiquidity	Private 🔐	🔴	

PERBDividendTracker	Implementation	Ownable, DividendPayingToken
└	<Constructor>	Public ❗	🔴	DividendPayingToken
└	_transfer	Internal 🔒		
└	setMinBalanceForDividends	External ❗	🔴	onlyOwner
└	excludeFromDividends	External ❗	🔴	onlyOwner
└	updateClaimWait	External ❗	🔴	onlyOwner
└	getLastProcessedIndex	External ❗	NO ❗	
└	getNumberOfTokenHolders	External ❗	NO ❗	
└	getCurrentRewardToken	External ❗	NO ❗	
└	getAccount	Public ❗	NO ❗	
└	getAccountAtIndex	Public ❗	NO ❗	
└	canAutoClaim	Private 🔐		
└	setBalance	Public ❗	🔴	onlyOwner
└	process	Public ❗	🔴	NO ❗
└	processAccount	Public ❗	🔴	onlyOwner

DividendPayingToken	Implementation	ERC20, DividendPayingTokenInterface, Ownable
└	<Constructor>	Public ❗	🔴	ERC20
└	<Receive Ether>	External ❗	💵	NO ❗
└	distributeDividends	Public ❗	💵	NO ❗
└	_withdrawDividendOfUser	Internal 🔒	🔴	
└	setRewardToken	External ❗	🔴	onlyOwner
└	swapBnbForCustomToken	Internal 🔒	🔴	
└	dividendOf	Public ❗	NO ❗	
└	withdrawableDividendOf	Public ❗	NO ❗	
└	withdrawnDividendOf	Public ❗	NO ❗	
└	accumulativeDividendOf	Public ❗	NO ❗	
└	_transfer	Internal 🔒	🔴	
└	_tokengeneration	Internal 🔒	🔴	

expelee

FUNCTION DETAILS

└	_burn	Internal 🔒	🔴	
└	_setBalance	Internal 🔒	🔴	
ERC20	Implementation	Context, IERC20, IERC20Metadata		
└	<Constructor>	Public ❗	🔴	NO ❗
└	name	Public ❗		NO ❗
└	symbol	Public ❗		NO ❗
└	decimals	Public ❗		NO ❗
└	totalSupply	Public ❗		NO ❗
└	balanceOf	Public ❗		NO ❗
└	transfer	Public ❗	🔴	NO ❗
└	allowance	Public ❗		NO ❗
└	approve	Public ❗	🔴	NO ❗
└	transferFrom	Public ❗	🔴	NO ❗
└	increaseAllowance	Public ❗	🔴	NO ❗
└	decreaseAllowance	Public ❗	🔴	NO ❗
└	_transfer	Internal 🔒	🔴	
└	_tokengeneration	Internal 🔒	🔴	
└	_burn	Internal 🔒	🔴	
└	_approve	Internal 🔒	🔴	
└	_beforeTokenTransfer	Internal 🔒	🔴	
IERC20	Interface			
└	totalSupply	External ❗		NO ❗
└	balanceOf	External ❗		NO ❗
└	transfer	External ❗	🔴	NO ❗
└	allowance	External ❗		NO ❗
└	approve	External ❗	🔴	NO ❗
└	transferFrom	External ❗	🔴	NO ❗
IERC20Metadata	Interface	IERC20		
└	name	External ❗		NO ❗
└	symbol	External ❗		NO ❗
└	decimals	External ❗		NO ❗
Context	Implementation			
└	_msgSender	Internal 🔒		
└	_msgData	Internal 🔒		
SafeMath	Library			
└	add	Internal 🔒		
└	sub	Internal 🔒		
└	sub	Internal 🔒		
└	mul	Internal 🔒		
└	div	Internal 🔒		
└	div	Internal 🔒		

expelee

FUNCTION DETAILS

└	mod	Internal 🔒		
└	mod	Internal 🔒		
SafeMathInt	Library				
└	mul	Internal 🔒		
└	div	Internal 🔒		
└	sub	Internal 🔒		
└	add	Internal 🔒		
└	abs	Internal 🔒		
└	toUint256Safe	Internal 🔒		
SafeMathUint	Library				
└	toInt256Safe	Internal 🔒		
DividendPayingTokenInterface	Interface				
└	dividendOf	External ❗	|NO ❗	
└	distributeDividends	External ❗	💵 |NO ❗	
└	withdrawableDividendOf	External ❗	|NO ❗	
└	withdrawnDividendOf	External ❗	|NO ❗	
└	accumulativeDividendOf	External ❗	|NO ❗	
Ownable	Implementation	Context			
└	<Constructor>	Public ❗	🔴 |NO ❗	
└	owner	Public ❗	|NO ❗	
└	renounceOwnership	Public ❗	🔴	onlyOwner	
└	transferOwnership	Public ❗	🔴	onlyOwner	
IPair	Interface				
└	sync	External ❗	🔴 |NO ❗	
IFactory	Interface				
└	createPair	External ❗	🔴 |NO ❗	
└	getPair	External ❗	|NO ❗	
IRouter	Interface				
└	factory	External ❗	|NO ❗	
└	WETH	External ❗	|NO ❗	
└	addLiquidityETH	External ❗	💵 |NO ❗	
└	swapExactTokensForTokensSupportingFeeOnTransferTokens	External ❗	🔴 |NO ❗	
└	swapExactETHForTokens	External ❗	💵 |NO ❗	
└	swapExactTokensForETHSupportingFeeOnTransferTokens	External ❗	🔴 |NO ❗	
└	swapExactETHForTokensSupportingFeeOnTransferTokens	External ❗	💵 |NO ❗	
IterableMapping	Library				
└	get	Internal 🔒		
└	getIndexOfKey	Internal 🔒		
└	getKeyAtIndex	Internal 🔒		

FUNCTION DETAILS

└	size	Internal 🔒		
└	set	Internal 🔒	🔴	
└	remove	Internal 🔒	🔴	

Legend

Symbol	Meaning
🔴	Function can modify state
💵	Function is payable

MANUAL REVIEW

Severity Criteria

Expelee assesses the severity of disclosed vulnerabilities according to methodology based on OWASP standarts.

Vulnerabilities are dividend into three primary risk categroies:
High
Medium
Low

High-level considerations for vulnerabilities span the following key areas when conducting assessments:

- Malicious input handling
- Escalation of privileges
- Arithmetic
- Gas use

Overall Risk Severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

expelee

FINDINGS

Findings	Severity	Found
High Risk	High	0
Medium Risk	Medium	0
Low Risk	Low	0
Suggestion & discussion	Informational	0
Gas Optimizations	Gas Opt.	0

ABOUT EXPELEE

Expelee is a product-based aspirational Web3 start-up. Coping up with numerous solutions for blockchain security and constructing a Web3 ecosystem from deal making platform to developer hosting open platform, while also developing our own commercial and sustainable blockchain.

www.expelee.com

expeleeofficial

expelee

Expelee

expelee

expelee_official

expelee-co

expelee

Building the Futuristic Blockchain Ecosystem

DISCLAIMER

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment. Team provides no guarantess against the sale of team tokens or the removal of liquidity by the project audited in this document.

Always do your own research and project yourselves from being scammed. The Expelee team has audited this project for general information and only expresses their opinion based on similar projects and checks from popular diagnostic tools.

Under no circumstances did Expelee receive a payment to manipulate those results or change the awarding badge that we will be adding in our website. Alway do your own research and protect yourselves from scams.

This document should not be presented as a reason to buy or not buy any particular token. The Expelee team disclaims any liability for the resulting losses.