<u>Party Doge</u>
<u>INITIAL AUDIT REPORT</u>

<u>by AudiBloq, August 2021</u>

Introduction :

This Audit Report highlights the overall security of the **Party Doge** Smart Contract. With this report, we have tried to ensure the reliability of their smart contract by a complete assessment of their system's architecture and the smart contract codebase.

Auditing Approach and Methodologies applied :

The AudiBloq team has performed thorough testing of the project, starting with analysing the code design patterns in which we reviewed the smart contract architecture to ensure it is structured and safe use of third party smart contracts and libraries.

Our team then performed a formal line by line inspection of the Smart Contract to find any potential issue like race conditions, transaction-ordering dependence, timestamp dependence, and denial of service attacks.

In the Unit testing Phase, we coded/conducted Custom unit tests written for each function in the contract to verify that each function works as expected. In Automated Testing, We tested the Smart Contract with our in-house developed tools to identify vulnerabilities and security flaws.

The code was tested in collaboration with our multiple team members, and this included -

1. Testing the functionality of the Smart Contract to determine proper logic has been followed throughout the process.
2. Analysing the complexity of the code by thorough, manual review of the code, line-by-line.

3. Deploying the code on testnet using multiple clients to run live tests

4. Analysing failure preparations to check how the Smart Contract performs in case of bugs and vulnerabilities.

5. Checking whether all the libraries used in the code are on the latest version.

6. Analysing the security of the on-chain data.

Audit Details

- Project Name: Party Doge

- Party Doge Token

- Languages: Solidity (Smart contract), Javascript (Unit Testing)

- Platforms and Tools: Remix IDE, Truffle, Truffle Team, Ganache, Slither, Surya

<u>Summary of Party Doge Smart Contract :</u>

AudiBloq conducted a security audit of a smart contract of Party Doge. Party Doge contracts are used as token contracts with .

Audit Goals

The focus of the audit was to verify that the smart contract system is secure, resilient and working according to its specifications. The audit activities can be grouped into the following three categories:

<u>Security</u>: Identifying security related issues within each contract and the system of contracts.

<u>Sound Architecture</u>: Evaluation of the architecture of this system through the lens of established smart contract best practices and general software best practices.

Code Correctness and Quality: A full review of the contract source code. The primary areas of focus include:

- Correctness
- Readability
- Sections of code with high complexity
- Quantity and quality of test coverage

Security Level references :

Every issue in this report was assigned a severity level from the following:

High severity issues will bring problems and should be fixed.

Medium severity issues could potentially bring problems and should eventually be fixed.

Low severity issues are minor details and warnings that can remain unfixed but would be better fixed at some point in the future.

Number of issues per severity

	Low	Medium	High
Open	5	0	0
Closed	0	0	0

Manual Audit:

High severity issues:-

No High Severity Issue.

Medium Severity Issues:-

No High Severity Issue.

Low Severity Issues:-

1 **Contain Assembly Code**

- Address.isContract (./PartyDoge.sol#85-94) is declared view but contains assembly code

Status: Not Fixed

2 Reentrancy

Reentrancy in PartyDoge._transfer (./PartyDoge.sol#620-737):

External calls:

- sellHistory.bnbAmount = _getSellBnBAmount(amount) (./PartyDoge.sol#638)

- swapTokens(contractTokenBalance) (./PartyDoge.sol#649)

- _removeOldSellHistories() (./PartyDoge.sol#676)

State variables written after the call(s):

- _sellHistories (./PartyDoge.sol#676)

Reentrancy in PartyDoge._transfer (./PartyDoge.sol#620-737):

External calls:

- sellHistory.bnbAmount = _getSellBnBAmount(amount) (./PartyDoge.sol#638)

- swapTokens(contractTokenBalance) (./PartyDoge.sol#649)

- _removeOldSellHistories() (./PartyDoge.sol#676)

- buyBackTokens(_bBSLimit) (./PartyDoge.sol#684)

State variables written after the call(s):

- _rOwned (./PartyDoge.sol#736)

- _rTotal (./PartyDoge.sol#736)

- _tOwned (./PartyDoge.sol#736)

- inSwapAndLiquify (./PartyDoge.sol#684)

Status: Not Fixed

3 Local Variable Never Initialized

- sellHistory in PartyDoge._transfer (./PartyDoge.sol#636) is a local variable never initialiazed

Status: Not Fixed

4 Does not use return variable

PartyDoge.includeInReward (./PartyDoge.sol#599-610) does not use the value returned by external calls:

- _excluded.pop() (./PartyDoge.sol#606)

PartyDoge._removeOldSellHistories (./PartyDoge.sol#949-971) does not use the value returned by external calls:

- _sellHistories.pop() (./PartyDoge.sol#968)

Status: Not Fixed

5 Reentrancy

Reentrancy in PartyDoge._transfer (./PartyDoge.sol#620-737):

   External calls:

   - sellHistory.bnbAmount = _getSellBnBAmount(amount) (./PartyDoge.sol#638)

   State variables written after the call(s):

   - _sellHistories (./PartyDoge.sol#640)

   - _startTimeForSwap (./PartyDoge.sol#647)

Reentrancy in PartyDoge._transfer (./PartyDoge.sol#620-737):

   External calls:

   - sellHistory.bnbAmount = _getSellBnBAmount(amount) (./PartyDoge.sol#638)

   - swapTokens(contractTokenBalance) (./PartyDoge.sol#649)

   State variables written after the call(s):

   - _allowances (./PartyDoge.sol#649)

Reentrancy in PartyDoge._transfer (./PartyDoge.sol#620-737):

   External calls:

   - sellHistory.bnbAmount = _getSellBnBAmount(amount) (./PartyDoge.sol#638)

   - swapTokens(contractTokenBalance) (./PartyDoge.sol#649)

   - _removeOldSellHistories() (./PartyDoge.sol#676)

   - buyBackTokens(_bBSLimit) (./PartyDoge.sol#684)

State variables written after the call(s):

- _liquidityFee (./PartyDoge.sol#700)

- _liquidityFee (./PartyDoge.sol#702)

- _liquidityFee (./PartyDoge.sol#706)

- _liquidityFee (./PartyDoge.sol#708)

- _liquidityFee (./PartyDoge.sol#713)

- _liquidityFee (./PartyDoge.sol#715)

- _liquidityFee (./PartyDoge.sol#720)

- _liquidityFee (./PartyDoge.sol#727)

- _liquidityFee (./PartyDoge.sol#730)

- _liquidityFee (./PartyDoge.sol#736)

- _previousLiquidityFee (./PartyDoge.sol#700)

- _previousLiquidityFee (./PartyDoge.sol#706)

- _previousLiquidityFee (./PartyDoge.sol#713)

- _previousLiquidityFee (./PartyDoge.sol#727)

- _previousLiquidityFee (./PartyDoge.sol#736)

- _previousTaxFee (./PartyDoge.sol#700)

- _previousTaxFee (./PartyDoge.sol#706)

- _previousTaxFee (./PartyDoge.sol#713)

- _previousTaxFee (./PartyDoge.sol#727)

- _previousTaxFee (./PartyDoge.sol#736)

- _tFeeTotal (./PartyDoge.sol#736)

- _taxFee (./PartyDoge.sol#700)

- _taxFee (./PartyDoge.sol#701)

- _taxFee (./PartyDoge.sol#706)

- _taxFee (./PartyDoge.sol#707)

- _taxFee (./PartyDoge.sol#713)

- _taxFee (./PartyDoge.sol#714)

- _taxFee (./PartyDoge.sol#719)

- _taxFee (./PartyDoge.sol#727)

- _taxFee (./PartyDoge.sol#729)

- _taxFee (./PartyDoge.sol#736)

Reentrancy in PartyDoge.changeRouterVersion (./PartyDoge.sol#1070-1083):

External calls:

- _pair = IUniswapV2Factory(_uniswapV2Router.factory()).getPair(address(this),_uniswapV2Router.WETH()) (./PartyDoge.sol#1073)

- _pair = IUniswapV2Factory(_uniswapV2Router.factory()).createPair(address(this),_uniswapV2Router.WETH()) (./PartyDoge.sol#1076-1077)

State variables written after the call(s):

- uniswapV2Pair (./PartyDoge.sol#1079)

- uniswapV2Router (./PartyDoge.sol#1082)

Reentrancy in PartyDoge.constructor (./PartyDoge.sol#471-493):

External calls:

- uniswapV2Pair = IUniswapV2Factory(_uniswapV2Router.factory()).createPair(address(this),_uniswapV2Router.WETH()) (./PartyDoge.sol#481-482)

State variables written after the call(s):

- _isExcludedFromFee (./PartyDoge.sol#487)

- _isExcludedFromFee (./PartyDoge.sol#488)

- _startTimeForSwap (./PartyDoge.sol#490)

- uniswapV2Router (./PartyDoge.sol#484)

Reentrancy in PartyDoge.swapTokensForEth (./PartyDoge.sol#757-775):

External calls:

- path[1] = uniswapV2Router.WETH() (./PartyDoge.sol#761)

State variables written after the call(s):

- _allowances (./PartyDoge.sol#763)

Reentrancy in PartyDoge.transferFrom (./PartyDoge.sol#530-534):

External calls:

- _transfer(sender,recipient,amount) (./PartyDoge.sol#531)

State variables written after the call(s):

- _allowances (./PartyDoge.sol#532)

Reference:
https://github.com/trailofbits/slither/wiki/Detectors-Documentation#reentrancy-vulnerabilities-2

Status: Not Fixed

Automated Audit

Slither Tool Result :

```
pragma solidity 0.5.10 (./PartyDoge.sol#1): it allows old versions
Reference: https://github.com/trailofbits/slither/wiki/Detectors-Documentation#incorrect-version-of-solidity
INFO:Detectors:
Function 'Context._msgSender' (./PartyDoge.sol#4-6) is not in mixedCase
Function 'Context._msgData' (./PartyDoge.sol#8-11) is not in mixedCase
Parameter '' of IUniswapV2Factory.allPairs (./PartyDoge.sol#164) is not in mixedCase
Parameter '' of IUniswapV2Factory.setFeeTo (./PartyDoge.sol#169) is not in mixedCase
Parameter '' of IUniswapV2Factory.setFeeToSetter (./PartyDoge.sol#170) is not in mixedCase
Function 'IUniswapV2Pair.DOMAIN_SEPARATOR' (./PartyDoge.sol#191) is not in mixedCase
Function 'IUniswapV2Pair.PERMIT_TYPEHASH' (./PartyDoge.sol#192) is not in mixedCase
Function 'IUniswapV2Pair.MINIMUM_LIQUIDITY' (./PartyDoge.sol#208) is not in mixedCase
Parameter '' of IUniswapV2Pair.initialize (./PartyDoge.sol#222) is not in mixedCase
Parameter '_scope_0' of IUniswapV2Pair.initialize (./PartyDoge.sol#222) is not in mixedCase
Function 'IUniswapV2Router01.WETH' (./PartyDoge.sol#229) is not in mixedCase
Function 'PartyDoge._approve' (./PartyDoge.sol#612-618) is not in mixedCase
Function 'PartyDoge._transfer' (./PartyDoge.sol#620-737) is not in mixedCase
Function 'PartyDoge._tokenTransfer' (./PartyDoge.sol#789-805) is not in mixedCase
Function 'PartyDoge._transferStandard' (./PartyDoge.sol#807-814) is not in mixedCase
Function 'PartyDoge._transferToExcluded' (./PartyDoge.sol#816-824) is not in mixedCase
Function 'PartyDoge._transferFromExcluded' (./PartyDoge.sol#826-834) is not in mixedCase
Function 'PartyDoge._transferBothExcluded' (./PartyDoge.sol#836-845) is not in mixedCase
Function 'PartyDoge._reflectFee' (./PartyDoge.sol#847-850) is not in mixedCase
Function 'PartyDoge._getValues' (./PartyDoge.sol#852-856) is not in mixedCase
Function 'PartyDoge._getTValues' (./PartyDoge.sol#858-863) is not in mixedCase
Function 'PartyDoge._getRValues' (./PartyDoge.sol#865-871) is not in mixedCase
Function 'PartyDoge._getRate' (./PartyDoge.sol#873-876) is not in mixedCase
Function 'PartyDoge._getCurrentSupply' (./PartyDoge.sol#878-888) is not in mixedCase
Function 'PartyDoge._takeLiquidity' (./PartyDoge.sol#890-896) is not in mixedCase
Parameter '_amount' of PartyDoge.calculateTaxFee (./PartyDoge.sol#898) is not in mixedCase
Parameter '_amount' of PartyDoge.calculateLiquidityFee (./PartyDoge.sol#904) is not in mixedCase
Function 'PartyDoge._getSellBnBAmount' (./PartyDoge.sol#938-947) is not in mixedCase
Function 'PartyDoge._removeOldSellHistories' (./PartyDoge.sol#949-971) is not in mixedCase
Function 'PartyDoge.SetBuyBackMaxTimeForHistories' (./PartyDoge.sol#973-975) is not in mixedCase
Function 'PartyDoge.SetBuyBackDivisor' (./PartyDoge.sol#977-979) is not in mixedCase
Function 'PartyDoge.GetBuyBackTimeInterval' (./PartyDoge.sol#981-983) is not in mixedCase
Function 'PartyDoge.SetBuyBackTimeInterval' (./PartyDoge.sol#985-987) is not in mixedCase
Function 'PartyDoge.SetBuyBackRangeRate' (./PartyDoge.sol#989-992) is not in mixedCase
Function 'PartyDoge.GetSwapMinutes' (./PartyDoge.sol#994-996) is not in mixedCase
Function 'PartyDoge.SetSwapMinutes' (./PartyDoge.sol#998-1000) is not in mixedCase
```

```
        - _taxFee (./PartyDoge.sol#736)
Reentrancy in PartyDoge.changeRouterVersion (./PartyDoge.sol#1070-1083):
        External calls:
        - _pair = IUniswapV2Factory(_uniswapV2Router.factory()).getPair(address(this),_uniswapV2Router.WETH()) (./PartyDoge.sol#1073)
        - _pair = IUniswapV2Factory(_uniswapV2Router.factory()).createPair(address(this),_uniswapV2Router.WETH()) (./PartyDoge.sol#1076-1077)
        State variables written after the call(s):
        - uniswapV2Pair (./PartyDoge.sol#1079)
        - uniswapV2Router (./PartyDoge.sol#1082)
Reentrancy in PartyDoge.constructor (./PartyDoge.sol#471-493):
        External calls:
        - uniswapV2Pair = IUniswapV2Factory(_uniswapV2Router.factory()).createPair(address(this),_uniswapV2Router.WETH()) (./PartyDoge.sol#481
-482)
        State variables written after the call(s):
        - _isExcludedFromFee (./PartyDoge.sol#487)
        - _isExcludedFromFee (./PartyDoge.sol#488)
        - _startTimeForSwap (./PartyDoge.sol#490)
        - uniswapV2Router (./PartyDoge.sol#484)
Reentrancy in PartyDoge.swapTokensForEth (./PartyDoge.sol#757-775):
        External calls:
        - path[1] = uniswapV2Router.WETH() (./PartyDoge.sol#761)
        State variables written after the call(s):
        - _allowances (./PartyDoge.sol#763)
Reentrancy in PartyDoge.transferFrom (./PartyDoge.sol#530-534):
        External calls:
        - _transfer(sender,recipient,amount) (./PartyDoge.sol#531)
        State variables written after the call(s):
        - _allowances (./PartyDoge.sol#532)
Reference: https://github.com/trailofbits/slither/wiki/Detectors-Documentation#reentrancy-vulnerabilities-2
INFO:Detectors:
Ownable.unlock (./PartyDoge.sol#146-151) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(block.timestamp > _lockTime,Contract is locked until 7 days) (./PartyDoge.sol#148)
PartyDoge._transfer (./PartyDoge.sol#620-737) uses timestamp for comparisons
        Dangerous comparisons:
        - _sellHistories[i].time >= startTime (./PartyDoge.sol#666-669)
        - balance > _bBSLimit (./PartyDoge.sol#683-685)
        - overMinimumTokenBalance && _startTimeForSwap + _intervalMinutesForSwap <= block.timestamp (./PartyDoge.sol#646-650)
PartyDoge.buyBackTokens (./PartyDoge.sol#751-755) uses timestamp for comparisons
        Dangerous comparisons:
```

```
Reference: https://github.com/trailofbits/slither/wiki/Detectors-Documentation#local-variable-shadowing
INFO:Detectors:
PartyDoge.includeInReward has external calls inside a loop:
        - _excluded.pop() (./PartyDoge.sol#606)
PartyDoge._removeOldSellHistories has external calls inside a loop:
        - _sellHistories.pop() (./PartyDoge.sol#968)
Reference: https://github.com/trailofbits/slither/wiki/Detectors-Documentation/_edit#calls-inside-a-loop
INFO:Detectors:
Reentrancy in PartyDoge._transfer (./PartyDoge.sol#620-737):
        External calls:
        - sellHistory.bnbAmount = _getSellBnBAmount(amount) (./PartyDoge.sol#638)
        State variables written after the call(s):
        - _sellHistories (./PartyDoge.sol#640)
        - _startTimeForSwap (./PartyDoge.sol#647)
Reentrancy in PartyDoge._transfer (./PartyDoge.sol#620-737):
        External calls:
        - sellHistory.bnbAmount = _getSellBnBAmount(amount) (./PartyDoge.sol#638)
        - swapTokens(contractTokenBalance) (./PartyDoge.sol#649)
        State variables written after the call(s):
        - _allowances (./PartyDoge.sol#649)
Reentrancy in PartyDoge._transfer (./PartyDoge.sol#620-737):
        External calls:
        - sellHistory.bnbAmount = _getSellBnBAmount(amount) (./PartyDoge.sol#638)
        - swapTokens(contractTokenBalance) (./PartyDoge.sol#649)
        - _removeOldSellHistories() (./PartyDoge.sol#676)
        - buyBackTokens(_bBSLimit) (./PartyDoge.sol#684)
        State variables written after the call(s):
        - _liquidityFee (./PartyDoge.sol#700)
        - _liquidityFee (./PartyDoge.sol#702)
        - _liquidityFee (./PartyDoge.sol#706)
        - _liquidityFee (./PartyDoge.sol#708)
        - _liquidityFee (./PartyDoge.sol#713)
        - _liquidityFee (./PartyDoge.sol#715)
        - _liquidityFee (./PartyDoge.sol#720)
        - _liquidityFee (./PartyDoge.sol#727)
        - _liquidityFee (./PartyDoge.sol#730)
        - _liquidityFee (./PartyDoge.sol#736)
        - _previousLiquidityFee (./PartyDoge.sol#700)
```

```
INFO:Detectors:
Address.isContract (./PartyDoge.sol#85-94) is declared view but contains assembly code
Reference: https://github.com/trailofbits/slither/wiki/Detectors-Documentation#constant-functions-changing-the-state
INFO:Detectors:
Reentrancy in PartyDoge._transfer (./PartyDoge.sol#620-737):
        External calls:
        - sellHistory.bnbAmount = _getSellBnBAmount(amount) (./PartyDoge.sol#638)
        - swapTokens(contractTokenBalance) (./PartyDoge.sol#649)
        - _removeOldSellHistories() (./PartyDoge.sol#676)
        State variables written after the call(s):
        - _sellHistories (./PartyDoge.sol#676)
Reentrancy in PartyDoge._transfer (./PartyDoge.sol#620-737):
        External calls:
        - sellHistory.bnbAmount = _getSellBnBAmount(amount) (./PartyDoge.sol#638)
        - swapTokens(contractTokenBalance) (./PartyDoge.sol#649)
        - _removeOldSellHistories() (./PartyDoge.sol#676)
        - buyBackTokens(_bBSLimit) (./PartyDoge.sol#684)
        State variables written after the call(s):
        - _rOwned (./PartyDoge.sol#736)
        - _rTotal (./PartyDoge.sol#736)
        - _tOwned (./PartyDoge.sol#736)
        - inSwapAndLiquify (./PartyDoge.sol#684)
Reference: https://github.com/trailofbits/slither/wiki/Detectors-Documentation#reentrancy-vulnerabilities-1
INFO:Detectors:
sellHistory in PartyDoge._transfer (./PartyDoge.sol#636) is a local variable never initialiazed
Reference: https://github.com/trailofbits/slither/wiki/Detectors-Documentation#uninitialized-local-variables
INFO:Detectors:
PartyDoge.includeInReward (./PartyDoge.sol#599-610) does not use the value returned by external calls:
        -_excluded.pop() (./PartyDoge.sol#606)
PartyDoge._removeOldSellHistories (./PartyDoge.sol#949-971) does not use the value returned by external calls:
        -_sellHistories.pop() (./PartyDoge.sol#968)
Reference: https://github.com/trailofbits/slither/wiki/Detectors-Documentation#unused-return
INFO:Detectors:
PartyDoge.allowance.owner (local variable @ ./PartyDoge.sol#521) shadows:
        - Ownable.owner (function @ ./PartyDoge.sol#111-113)
PartyDoge._approve.owner (local variable @ ./PartyDoge.sol#612) shadows:
        - Ownable.owner (function @ ./PartyDoge.sol#111-113)
Reference: https://github.com/trailofbits/slither/wiki/Detectors-Documentation#local-variable-shadowing
```

Results

Some false positive errors have been reported by the tool, all other errors have been covered in issues explained above, under low level severity issues.

Implementation Recommendations :

Function 'Context._msgSender' (./PartyDoge.sol#4-6) is not in mixedCase

Function 'Context._msgData' (./PartyDoge.sol#8-11) is not in mixedCase

Parameter '' of IUniswapV2Factory.allPairs (./PartyDoge.sol#164) is not in mixedCase

Parameter '' of IUniswapV2Factory.setFeeTo (./PartyDoge.sol#169) is not in mixedCase

Parameter '' of IUniswapV2Factory.setFeeToSetter (./PartyDoge.sol#170) is not in mixedCase

Function 'IUniswapV2Pair.DOMAIN_SEPARATOR' (./PartyDoge.sol#191) is not in mixedCase

Function 'IUniswapV2Pair.PERMIT_TYPEHASH' (./PartyDoge.sol#192) is not in mixedCase

Function 'IUniswapV2Pair.MINIMUM_LIQUIDITY' (./PartyDoge.sol#208) is not in mixedCase

Parameter '' of IUniswapV2Pair.initialize (./PartyDoge.sol#222) is not in mixedCase

Parameter '_scope_0' of IUniswapV2Pair.initialize (./PartyDoge.sol#222) is not in mixedCase

Function 'IUniswapV2Router01.WETH' (./PartyDoge.sol#229) is not in mixedCase

Function 'PartyDoge._approve' (./PartyDoge.sol#612-618) is not in mixedCase

Function 'PartyDoge._transfer' (./PartyDoge.sol#620-737) is not in mixedCase

Function 'PartyDoge._tokenTransfer' (./PartyDoge.sol#789-805) is not in mixedCase

Function 'PartyDoge._transferStandard' (./PartyDoge.sol#807-814) is not in mixedCase

Function 'PartyDoge._transferToExcluded' (./PartyDoge.sol#816-824) is not in mixedCase

Function 'PartyDoge._transferFromExcluded' (./PartyDoge.sol#826-834) is not in mixedCase

Function 'PartyDoge._transferBothExcluded' (./PartyDoge.sol#836-845) is not in mixedCase

Function 'PartyDoge._reflectFee' (./PartyDoge.sol#847-850) is not in mixedCase

Function 'PartyDoge._getValues' (./PartyDoge.sol#852-856) is not in mixedCase

Function 'PartyDoge._getTValues' (./PartyDoge.sol#858-863) is not in mixedCase

Function 'PartyDoge._getRValues' (./PartyDoge.sol#865-871) is not in mixedCase

Function 'PartyDoge._getRate' (./PartyDoge.sol#873-876) is not in mixedCase

Function 'PartyDoge._getCurrentSupply' (./PartyDoge.sol#878-888) is not in mixedCase

Function 'PartyDoge._takeLiquidity' (./PartyDoge.sol#890-896) is not in mixedCase

Parameter '_amount' of PartyDoge.calculateTaxFee (./PartyDoge.sol#898) is not in mixedCase

Parameter '_amount' of PartyDoge.calculateLiquidityFee (./PartyDoge.sol#904) is not in mixedCase

Function 'PartyDoge._getSellBnBAmount' (./PartyDoge.sol#938-947) is not in mixedCase

Function 'PartyDoge._removeOldSellHistories' (./PartyDoge.sol#949-971) is not in mixedCase

Function 'PartyDoge.SetBuyBackMaxTimeForHistories' (./PartyDoge.sol#973-975) is not in mixedCase

Function 'PartyDoge.SetBuyBackDivisor' (./PartyDoge.sol#977-979) is not in mixedCase

Function 'PartyDoge.GetBuyBackTimeInterval' (./PartyDoge.sol#981-983) is not in mixedCase

Function 'PartyDoge.SetBuyBackTimeInterval' (./PartyDoge.sol#985-987) is not in mixedCase

Function 'PartyDoge.SetBuyBackRangeRate' (./PartyDoge.sol#989-992) is not in mixedCase

Function 'PartyDoge.GetSwapMinutes' (./PartyDoge.sol#994-996) is not in mixedCase

Function 'PartyDoge.SetSwapMinutes' (./PartyDoge.sol#998-1000) is not in mixedCase

Parameter '_minimumTokensBeforeSwap' of PartyDoge.setNumTokensSellToAddToBuyBack (./PartyDoge.sol#1032) is not in mixedCase

Parameter '_enabled' of PartyDoge.setSwapAndLiquifyEnabled (./PartyDoge.sol#1037) is not in mixedCase

Parameter '_enabled' of PartyDoge.setBuyBackEnabled (./PartyDoge.sol#1042) is not in mixedCase

Parameter '_enabled' of PartyDoge.setAutoBuyBackEnabled (./PartyDoge.sol#1047) is not in mixedCase

Parameter '_router' of PartyDoge.changeRouterVersion (./PartyDoge.sol#1070) is not in mixedCase

Parameter '_token' of PartyDoge.transferForeignToken (./PartyDoge.sol#1086) is not in mixedCase

Parameter '_to' of PartyDoge.transferForeignToken (./PartyDoge.sol#1086) is not in mixedCase

Function 'PartyDoge.Sweep' (./PartyDoge.sol#1092-1095) is not in mixedCase

Parameter '_address' of PartyDoge.setAddressFee (./PartyDoge.sol#1097) is not in mixedCase

Parameter '_enable' of PartyDoge.setAddressFee (./PartyDoge.sol#1097) is not in mixedCase

Parameter '_addressTaxFee' of PartyDoge.setAddressFee (./PartyDoge.sol#1097) is not in mixedCase

Parameter '_addressLiquidityFee' of PartyDoge.setAddressFee (./PartyDoge.sol#1097) is not in mixedCase

Parameter '_address' of PartyDoge.setBuyAddressFee (./PartyDoge.sol#1103) is not in mixedCase

Parameter '_enable' of PartyDoge.setBuyAddressFee (./PartyDoge.sol#1103) is not in mixedCase

Parameter '_addressTaxFee' of PartyDoge.setBuyAddressFee (./PartyDoge.sol#1103) is not in mixedCase

Parameter '_addressLiquidityFee' of PartyDoge.setBuyAddressFee (./PartyDoge.sol#1103) is not in mixedCase

Parameter '_address' of PartyDoge.setSellAddressFee (./PartyDoge.sol#1109) is not in mixedCase

Parameter '_enable' of PartyDoge.setSellAddressFee (./PartyDoge.sol#1109) is not in mixedCase

Parameter '_addressTaxFee' of PartyDoge.setSellAddressFee (./PartyDoge.sol#1109) is not in mixedCase

Parameter '_addressLiquidityFee' of PartyDoge.setSellAddressFee (./PartyDoge.sol#1109) is not in mixedCase

Variable 'PartyDoge._taxFee' (./PartyDoge.sol#403) is not in mixedCase

Variable 'PartyDoge._liquidityFee' (./PartyDoge.sol#406) is not in mixedCase

Variable 'PartyDoge._buyTaxFee' (./PartyDoge.sol#409) is not in mixedCase

Variable 'PartyDoge._buyLiquidityFee' (./PartyDoge.sol#410) is not in mixedCase

Variable 'PartyDoge._sellTaxFee' (./PartyDoge.sol#412) is not in mixedCase

Variable 'PartyDoge._sellLiquidityFee' (./PartyDoge.sol#413) is not in mixedCase

Variable 'PartyDoge._startTimeForSwap' (./PartyDoge.sol#415) is not in mixedCase

Variable 'PartyDoge._intervalMinutesForSwap' (./PartyDoge.sol#416) is not in mixedCase

Variable 'PartyDoge._buyBackRangeRate' (./PartyDoge.sol#418) is not in mixedCase

Variable 'PartyDoge._addressFees' (./PartyDoge.sol#421) is not in mixedCase

Variable 'PartyDoge._maxTxAmount' (./PartyDoge.sol#425) is not in mixedCase

Variable 'PartyDoge._sellHistories' (./PartyDoge.sol#430) is not in mixedCase

Variable 'PartyDoge._isAutoBuyBack' (./PartyDoge.sol#431) is not in mixedCase

---

Variable 'PartyDoge._buyBackDivisor' (./PartyDoge.sol#432) is not in mixedCase

Variable 'PartyDoge._buyBackTimeInterval' (./PartyDoge.sol#433) is not in mixedCase

Variable 'PartyDoge._buyBackMaxTimeForHistories' (./PartyDoge.sol#434) is not in mixedCase

Variable 'PartyDoge._isEnabledBuyBackAndBurn' (./PartyDoge.sol#443) is not in mixedCase

Disclaimer

AudiBloq audit is not a security warranty, investment advice, or an endorsement of the Party Doge contract. Securing smart contracts is a multistep process; therefore, running a bug bounty program as a complement to this audit is strongly recommended.

Closing Summary:

The use case of the smart contract is very well designed and Implemented. Overall, the code is well written and demonstrates effective use of abstraction, separation of concerns, and modularity. The **Party Doge** development team

demonstrated high technical capabilities, both in the design of the architecture and in the implementation.

Some low-severity issues have been reported and documented above; it is recommended to fix them before deploying.