EtherAuthority

www.EtherAuthority.io
audit@etherauthority.io

SMART CONTRACT

Security Audit Report

Customer:	BipGo Protocol
Platform:	Binance Smart Chain
Language:	Solidity
Date:	October 13th, 2021

Table of contents

THIS IS SECURITY AUDIT REPORT DOCUMENT AND WHICH MAY CONTAIN INFORMATION WHICH IS CONFIDENTIAL. WHICH INCLUDES ANY POTENTIAL VULNERABILITIES AND MALICIOUS CODES WHICH CAN BE USED TO EXPLOIT THE SOFTWARE. THIS MUST BE REFERRED INTERNALLY AND ONLY SHOULD BE MADE AVAILABLE TO THE PUBLIC AFTER ISSUES ARE RESOLVED.

Introduction

EtherAuthority was contracted by the BipGo Protocol team to perform the Security audit of the BipGo Protocol smart contracts code. The audit has been performed using manual analysis as well as using automated software tools. This report presents all the findings regarding the audit performed on October 13th, 2021.

The purpose of this audit was to address the following:

- Ensure that all claimed functions exist and function correctly.
- Identify any security vulnerabilities that may be present in the smart contract.

Project Background

The TOKEN will have a total supply of 400 million tokens, representing the total market capitalization over the life of the project (no additional TOKENS can be minted).

Audit scope

Name	Code Review and Security Analysis Report for BipGo Protocol Smart Contracts
Platform	**BSC / Solidity**
File 1	Token.sol
File 1 MD5 Hash	0ACE9879F6735F72BABAE49D99CEE014
File 2	PrivatePreSale.sol
File 2 MD5 Hash	ACE9AB1F468A98E1C43C7937199E4A41
File 3	PublicPresale.sol
File 3 MD5 Hash	2A554BDEB6278E898E9541D89A23C04D
Audit Date	October 13th, 2021

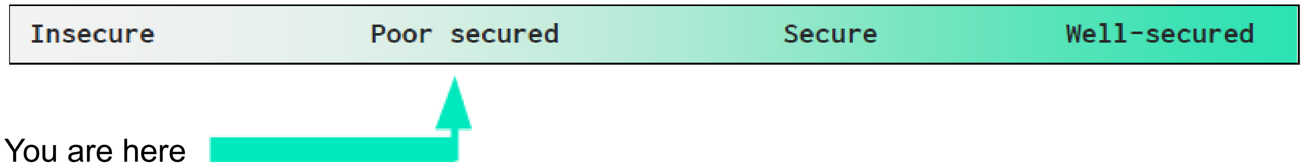
Claimed Smart Contract Features

Claimed Feature Detail	Our Observation
File 1: Token.sol ● Name**:** Token ● Symbol: TKN ● Decimals: 18 The total supply of TOKENS will be distributed over time in 9 accounts.The percentage of tokens that will be distributed in these accounts are shown below: 1. Private Sale: 6% 2. Public Sale: 5% 3. Liquidity: 1% 4. Team: 20% 5. Advisors: 5% 6. Marketing: 10% 7. Development: 10% 8. Share-To-Earn: 25% 9. Staking: 18%	**YES, This is valid.**
File 2: PrivatePreSale.sol ● Discount: 20% Price: 1 $ BNB = 3,750 $TOKEN ● Minimum contribution: 10 $BNB ● Maximum contribution: Unlimited ● Sales method: First come first serve	**YES, This is valid.**
File 3: PublicPresale.sol ● Price: 1 $BNB = 3,000 $TOKENS ● Minimum contribution: 0.1 $ BNB ● Maximum contribution: Unlimited ● Maximum number of interested parties:	**YES, This is valid.**

Unlimited ● Sales method: Oversubscription	

Audit Summary

According to the standard audit assessment, Customer`s solidity smart contracts are **"Not Secured"**. This token contract does contain owner control, which does not make it fully decentralized.



We used various tools like Slither, Solhint and Remix IDE. At the same time this finding is based on critical analysis of the manual audit.

All issues found during automated analysis were manually reviewed and applicable vulnerabilities are presented in the Audit overview section. General overview is presented in AS-IS section and all identified issues can be found in the Audit overview section.

We found 0 critical, 1 high, 2 medium and 3 low and some very low level issues.

Investors Advice: Technical audit of the smart contract does not guarantee the ethical nature of the project. Any owner controlled functions should be executed by the owner with responsibility. All investors/users are advised to do their due diligence before investing in the project.

Technical Quick Stats

Main Category	Subcategory	Result
Contract Programming	Solidity version not specified	Passed
	Solidity version too old	Moderated
	Integer overflow/underflow	Passed
	Function input parameters lack of check	Passed
	Function input parameters check bypass	Passed
	Function access control lacks management	Passed
	Critical operation lacks event log	Moderated
	Human/contract checks bypass	Passed
	Random number generation/use vulnerability	Passed
	Fallback function misuse	Passed
	Race condition	Passed
	Logical vulnerability	Passed
	Features claimed	Not Passed
	Other programming issues	Moderated
Code Specification	Function visibility not explicitly declared	Passed
	Var. storage location not explicitly declared	Passed
	Use keywords/functions to be deprecated	Passed
	Unused code	Passed
Gas Optimization	“Out of Gas” Issue	Passed
	High consumption ‘for/while’ loop	Moderated
	High consumption ‘storage’ storage	Passed
	Assert() misuse	Passed
Business Risk	The maximum limit for mintage not set	Passed
	“Short Address” Attack	Passed
	“Double Spend” Attack	Passed

Overall Audit Result: NOT PASSED

Code Quality

This audit scope has 3 smart contract files. Smart contracts also contain Libraries, Smart contracts inherits and Interfaces. These are compact and well written contracts.

The libraries in BipGo are part of its logical algorithm. A library is a different type of smart contract that contains reusable code. Once deployed on the blockchain (only once), it is assigned a specific address and its properties / methods can be reused many times by other contracts in the BipGo Protocol.

The BipGo Protocol team has **not** provided scenario and unit test scripts, which would have helped to determine the integrity of the code in an automated way.

Code parts are **Not well** commented on smart contracts.

Documentation

We were given a BipGo smart contracts code in the form of a file. The hashes of that code are mentioned above in the table.

As mentioned above, code parts are **Not well** commented. So it is not easy to quickly understand the programming flow as well as complex code logic. Comments are very helpful in understanding the overall architecture of the protocol.

Use of Dependencies

As per our observation, the libraries are used in this smart contract infrastructure that are based on well known industry standard open source projects. And their core code blocks are written well.

Apart from libraries, its functions are not used in external smart contract calls.

AS-IS overview

PublicPresale.sol

(1) Functions

Sl.	Functions	Type	Observation	Conclusion
1	constructor	read	Passed	No Issue
2	onlyAdmin	modifier	Passed	No Issue
3	transferOwnership	external	Critical operation lacks event log	Refer audit finding section
4	whiteListAddress	external	Critical operation lacks event log	Refer audit finding section
5	removeWhiteListAddress	external	Infinite loop	Refer audit finding section
6	receive	external	Ambiguous error message	Refer audit finding section
7	returnBNB	external	Critical operation lacks event log	Refer audit finding section
8	distribute15Nov21	external	Passed	No Issue
9	distribute15Dec22	external	Passed	No Issue
10	distribute15Jan22	external	Passed	No Issue
11	_distribute	write	Infinite loop	Refer audit finding section
12	endPublicPreSaleEarlier	external	access only Admin	No Issue
13	withdrawAll	external	access only Admin	No Issue

PrivatePreSale.sol

(1) Functions

Sl.	Functions	Type	Observation	Conclusion
1	constructor	read	Passed	No Issue
2	onlyAdmin	modifier	Passed	No Issue
3	transferOwnership	write	Critical operation lacks event log	Refer audit finding section
4	whiteListAddress	external	Infinite loop	Refer audit finding section
5	removeWhiteListAddress	external	Infinite loop	Refer audit finding section
6	receive	external	Ambiguous error message	Refer audit finding section
7	distribute15Nov21	external	Passed	No Issue
8	distribute15Dec21	external	Passed	No Issue
9	distribute15Jan22	external	Passed	No Issue
10	distribute15Febn22	external	Passed	No Issue
11	distribute15Mar22	external	Passed	No Issue
12	distribute15Apr22	external	Passed	No Issue

13	distribute15May22	external	Passed	No Issue
14	distribute15Jun22	external	Passed	No Issue
15	distribute15Jul22	external	Passed	No Issue
16	distribute15Aug22	external	Passed	No Issue
17	distribute15Sep22	external	Passed	No Issue
18	distribute15Oct22	external	Passed	No Issue
19	distribute15Novt22	external	Passed	No Issue
20	_distribute	write	Critical operation lacks event log	Refer audit finding section
21	withdrawAll	external	Critical operation lacks event log	Refer audit finding section

Token.sol

(1) Functions

Sl.	Functions	Type	Observation	Conclusion
1	constructor	read	Passed	No Issue
2	_setDefaultValues	write	Passed	No Issue
3	_setCoinDistribution	write	Passed	No Issue
4	onlyAdmin	modifier	Passed	No Issue
5	hasPrivatePreSaleContractNotYetSet	modifier	Passed	No Issue
6	hasPublicPreSaleContractNotYetSet	modifier	Passed	No Issue
7	isPrivatePreSaleContract	modifier	Passed	No Issue
8	isPublicPreSaleContract	modifier	Passed	No Issue
9	whenPaused	modifier	Passed	No Issue
10	whenNotPaused	modifier	Passed	No Issue
11	transferOwnership	external	access only Admin	No Issue
12	setPrivatePreSaleContractNotYetSet	external	access only Admin	No Issue
13	setPublicPreSaleContractNotYetSet	external	access only Admin	No Issue
14	totalSupply	read	Passed	No Issue
15	balanceOf	external	Passed	No Issue
16	transfer	external	Passed	No Issue
17	allowance	external	Passed	No Issue
18	approve	external	Passed	No Issue
19	transferFrom	external	Passed	No Issue
20	increaseAllowance	write	Passed	No Issue
21	decreaseAllowance	write	Passed	No Issue
22	_transfer	internal	Passed	No Issue
23	_approve	internal	Passed	No Issue
24	transferPrivatePresale	external	Critical operation lacks event log	Refer audit finding section
25	transferPublicPresale	external	Critical operation lacks event log	Refer audit finding section

26	transferLocalCategories	external	Critical operation lacks event log	Refer audit finding section
27	_categoriesTransfer	write	Passed	No Issue
28	pause	external	access only Admin	No Issue
29	unpause	external	access only Admin	No Issue
30	pausedAddress	external	access only Admin	No Issue
31	unPausedAddress	external	access only Admin	No Issue
32	receive	external	Passed	No Issue

Severity Definitions

Risk Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to token loss etc.
High	High-level vulnerabilities are difficult to exploit; however, they also have significant impact on smart contract execution, e.g. public access to crucial
Medium	Medium-level vulnerabilities are important to fix; however, they can't lead to tokens lose
Low	Low-level vulnerabilities are mostly related to outdated, unused etc. code snippets, that can't have significant impact on execution
Lowest / Code Style / Best Practice	Lowest-level vulnerabilities, code style violations and info statements can't affect smart contract execution and can be ignored.

Audit Findings

Token.sol

Critical

No Critical severity vulnerabilities were found.

High

(1) Token allocation:

Tokens don't get allocated to Team, Advisors, ShareToEarn and Staking as per the requirement.

Medium

No Medium severity vulnerabilities were found.

Low

(1) Critical operation lacks event log:

Missing event log for : transferPrivatePresale, transferPublicPresale, transferLocalCategories.

Resolution: Please write an event log for listed events.

Very Low / Informational / Best practices:

(1) Use latest solidity version:

```
pragma solidity ^0.8.0;
```

Using the latest solidity will prevent any compiler level bugs.

Resolution: Please use 0.8.9 which is the latest version.

(2) Missing error message in require condition:

It is best practice to add custom error messages in every required condition, which would be helpful in debugging as well as giving clear indication to any transaction failure.

Resolution: Add custom error messages in every required condition.

PrivatePreSale.sol

Critical

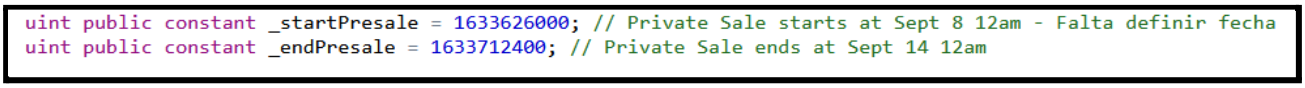
No Critical severity vulnerabilities were found.

High

No High severity vulnerabilities were found.

Medium

(1) Date values are wrong:

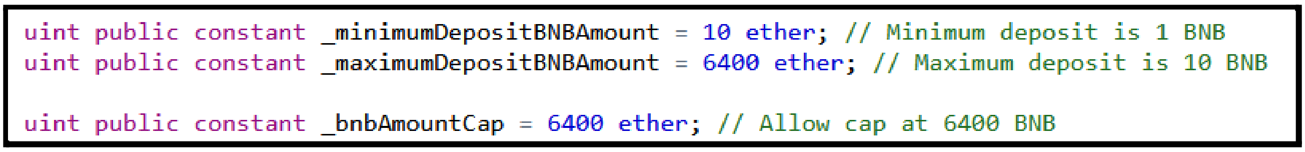


```
uint public constant _startPresale = 1633626000; // Private Sale starts at Sept 8 12am - Falta definir fecha
uint public constant _endPresale = 1633712400; // Private Sale ends at Sept 14 12am
```

Here the start and end time are past values.

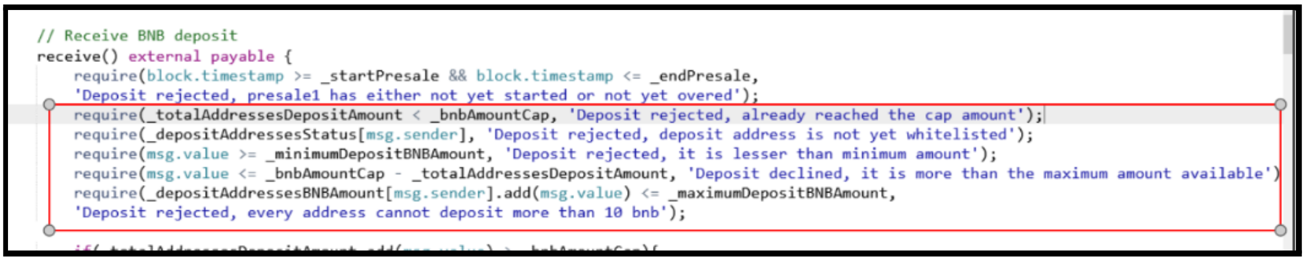
Resolution: Please correct the values and comment also.

(2) Minimum and Maximum limits are wrong:



```
uint public constant _minimumDepositBNBAmount = 10 ether; // Minimum deposit is 1 BNB
uint public constant _maximumDepositBNBAmount = 6400 ether; // Maximum deposit is 10 BNB

uint public constant _bnbAmountCap = 6400 ether; // Allow cap at 6400 BNB
```



```
// Receive BNB deposit
receive() external payable {
    require(block.timestamp >= _startPresale && block.timestamp <= _endPresale,
    'Deposit rejected, presale1 has either not yet started or not yet overed');
    require(_totalAddressesDepositAmount < _bnbAmountCap, 'Deposit rejected, already reached the cap amount');
    require(_depositAddressesStatus[msg.sender], 'Deposit rejected, deposit address is not yet whitelisted');
    require(msg.value >= _minimumDepositBNBAmount, 'Deposit rejected, it is lesser than minimum amount');
    require(msg.value <= _bnbAmountCap - _totalAddressesDepositAmount, 'Deposit declined, it is more than the maximum amount available')
    require(_depositAddressesBNBAmount[msg.sender].add(msg.value) <= _maximumDepositBNBAmount,
    'Deposit rejected, every address cannot deposit more than 10 bnb');

    if(_totalAddressesDepositAmount.add(msg.value) > _bnbAmountCap){
```

Minimum and Maximum limit for token purchase value don't match with the values mentioned in the document.

Resolution: Correct the limits.

Low

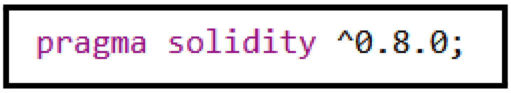
(1) Critical operation lacks event log:

Missing event log for : _distribute , withdrawAll,whiteListAddress , transferOwnership.

Resolution: Please write an event log for listed events.

Very Low / Informational / Best practices:

(1) Use latest solidity version:



```
pragma solidity ^0.8.0;
```

Using the latest solidity will prevent any compiler level bugs.

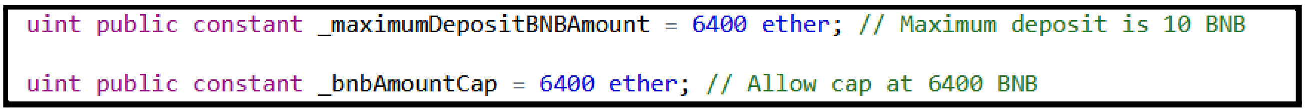
Resolution: Please use 0.8.9 which is the latest version.

(2) Infinite loop:

In whiteListAddress,removeWhiteListAddress,_distribute functions for loop do not have an upper length limit , which costs more gas.

Resolution: Upper limit should be limited in for loops.

(3) Identical value for 2 variables:



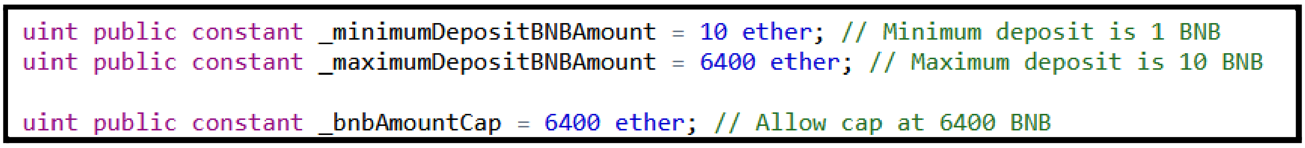
```
uint public constant _maximumDepositBNBAmount = 6400 ether; // Maximum deposit is 10 BNB

uint public constant _bnbAmountCap = 6400 ether; // Allow cap at 6400 BNB
```

Same value has been assigned to these 2 constant variables.

Resolution: We suggest using only 1 variable if the value is the same for both and other can be removed.

(4) Comment indicates different value:



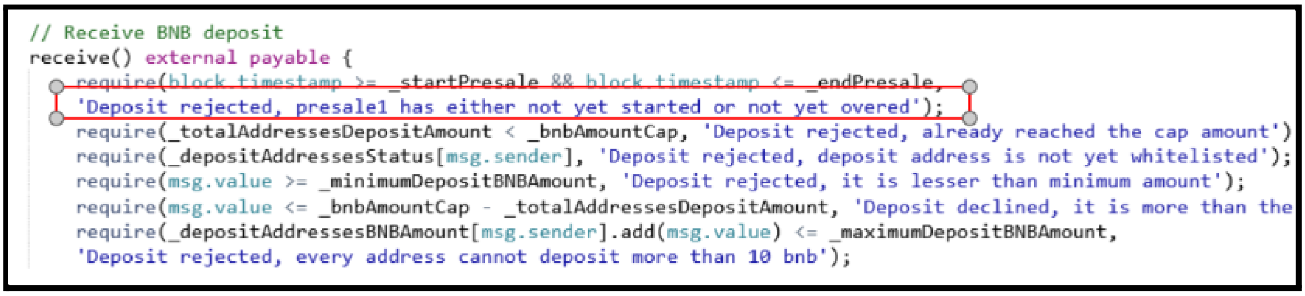
```
uint public constant _minimumDepositBNBAmount = 10 ether; // Minimum deposit is 1 BNB
uint public constant _maximumDepositBNBAmount = 6400 ether; // Maximum deposit is 10 BNB

uint public constant _bnbAmountCap = 6400 ether; // Allow cap at 6400 BNB
```

Comment indicates different values for these variables.

Resolution: We suggest double confirmation before deployment.

(5) Ambiguous error message:



```
// Receive BNB deposit
receive() external payable {
    require(block.timestamp >= _startPresale && block.timestamp <= _endPresale,
    'Deposit rejected, presale1 has either not yet started or not yet overed');
    require(_totalAddressesDepositAmount < _bnbAmountCap, 'Deposit rejected, already reached the cap amount')
    require(_depositAddressesStatus[msg.sender], 'Deposit rejected, deposit address is not yet whitelisted');
    require(msg.value >= _minimumDepositBNBAmount, 'Deposit rejected, it is lesser than minimum amount');
    require(msg.value <= _bnbAmountCap - _totalAddressesDepositAmount, 'Deposit declined, it is more than the
    require(_depositAddressesBNBAmount[msg.sender].add(msg.value) <= _maximumDepositBNBAmount,
    'Deposit rejected, every address cannot deposit more than 10 bnb');
```

The condition for checking the Presale end time must be greater than the current time, but the error message indicates it is wrong.

Resolution: Please correct the error message.

PublicPresale.sol

Critical

No Critical severity vulnerabilities were found.

High

No High severity vulnerabilities were found.

Medium

No Medium severity vulnerabilities were found.

Low

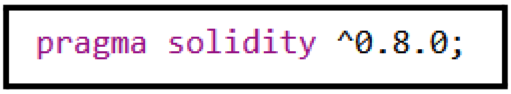
(1) Critical operation lacks event log:

Missing event log for : returnBNB , transferOwnership , whiteListAddress.

Resolution: Please write an event log for listed events.

Very Low / Informational / Best practices:

(1) Use latest solidity version:



```
pragma solidity ^0.8.0;
```

Using the latest solidity will prevent any compiler level bugs.

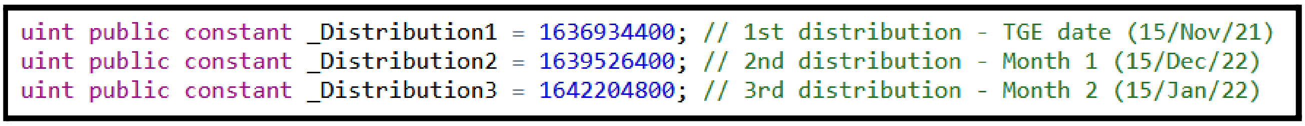
Resolution: Please use 0.8.9 which is the latest version.

(2) Infinite loop:

In _distribute , returnBNB , whiteListAddress,removeWhiteListAddress functions for loop do not have an upper length limit , which costs more gas.

Resolution: Upper limit should be limited in for loops.

(3) Incorrect date in comment:

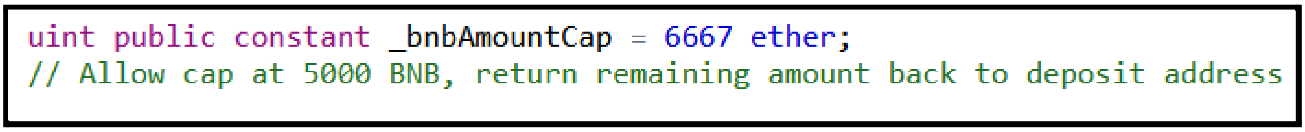


```
uint public constant _Distribution1 = 1636934400; // 1st distribution - TGE date (15/Nov/21)
uint public constant _Distribution2 = 1639526400; // 2nd distribution - Month 1 (15/Dec/22)
uint public constant _Distribution3 = 1642204800; // 3rd distribution - Month 2 (15/Jan/22)
```

Here _Distribution2's value has been set for 15/Dec/21 but the comment says it is 15/Dec/22.

Resolution: We suggest you correct the comment.

(4) Comment indicates different value:

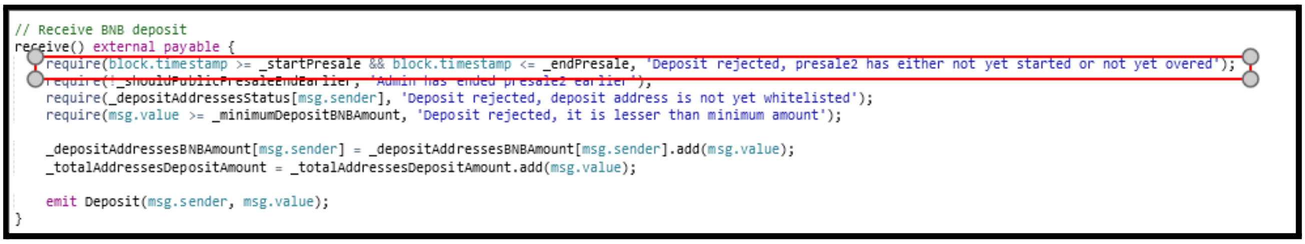


```
uint public constant _bnbAmountCap = 6667 ether;
// Allow cap at 5000 BNB, return remaining amount back to deposit address
```

Comment indicates different values for these variables.

Resolution: We suggest double confirmation before deployment.

(5) Ambiguous error message:



```
// Receive BNB deposit
receive() external payable {
    require(block.timestamp >= _startPresale && block.timestamp <= _endPresale, 'Deposit rejected, presale2 has either not yet started or not yet overed');
    require(!_shouldPublicPresaleEndEarlier, 'Admin has ended presale2 earlier');
    require(_depositAddressesStatus[msg.sender], 'Deposit rejected, deposit address is not yet whitelisted');
    require(msg.value >= _minimumDepositBNBAmount, 'Deposit rejected, it is lesser than minimum amount');

    _depositAddressesBNBAmount[msg.sender] = _depositAddressesBNBAmount[msg.sender].add(msg.value);
    _totalAddressesDepositAmount = _totalAddressesDepositAmount.add(msg.value);

    emit Deposit(msg.sender, msg.value);
}
```

The condition for checking the Presale end time must be greater than the current time, but the error message indicates it is wrong.

Resolution: Please correct the error message.

Centralization

These smart contracts have some functions which can be executed by the Admin (Owner) only. If the admin wallet private key would be compromised, then it would create trouble. Following are Admin functions:

- transferOwnership: The PrivatePreSale Admin owner can transfer ownership.
- whiteListAddress: The PrivatePreSale Admin owner can add deposit addresses and whitelist them.
- removeWhiteListAddress: The PrivatePreSale Admin owner can remove deposit addresses and unwhitelist them.
- withdrawAll: The PrivatePreSale Admin owner can allow the admin to withdraw all the deposited BNB.
- transferOwnership: The Public admin owner can transfer ownership.
- whiteListAddress: The PublicPresale Admin owner can add deposit addresses and whitelist them.
- removeWhiteListAddress: The PublicPresale Admin owner can remove deposit addresses and unwhitelist them.

- endPublicPreSaleEarlier: The PublicPresale Admin owner can allow admin to end Presale2 earlier.
- withdrawAll: The PublicPresale Admin owner allows the admin to withdraw all the deposited BNB.
- transferOwnership: The Token Admin owner can transfer ownership.
- setPrivatePreSaleContractNotYetSet: The Token Admin owner can set the private presale contract address.
- setPublicPreSaleContractNotYetSet: The Token Admin owner can set the public presale contract address.
- transferPrivatePresale: The Token owner can allow the preSale1 external contract to trigger the transfer function.
- transferPublicPresale: The Token owner can allow the preSale2 external contract to trigger the transfer function.
- pause: The Token Admin owner can set pause status true.
- unpause: The Token Admin owner can set the pause status false.
- pausedAddress: The Token Admin owner can set the pause address status false.
- unPausedAddress: The Token Admin owner can set the pause address status false.

Conclusion

We were given a contract code. And we have used all possible tests based on given objects as files. We observed some issues in the smart contracts and those issues are not critical ones. So, **it's not good to go to production**.

Since possible test cases can be unlimited for such smart contracts protocol, we provide no such guarantee of future outcomes. We have used all the latest static tools and manual observations to cover maximum possible test cases to scan everything.

Smart contracts within the scope were manually reviewed and analyzed with static analysis tools. Smart Contract's high level description of functionality was presented in As-is overview section of the report.

Audit report contains all found security vulnerabilities and other issues in the reviewed code.

Security state of the reviewed contract, based on standard audit procedure scope, is **"Not Secured"**.

Our Methodology

We like to work with a transparent process and make our reviews a collaborative effort. The goals of our security audits are to improve the quality of systems we review and aim for sufficient remediation to help protect users. The following is the methodology we use in our security audit process.

Manual Code Review:

In manually reviewing all of the code, we look for any potential issues with code logic, error handling, protocol and header parsing, cryptographic errors, and random number generators. We also watch for areas where more defensive programming could reduce the risk of future mistakes and speed up future audits. Although our primary focus is on the in-scope code, we examine dependency code and behavior when it is relevant to a particular line of investigation.

Vulnerability Analysis:

Our audit techniques included manual code analysis, user interface interaction, and whitebox penetration testing. We look at the project's web site to get a high level understanding of what functionality the software under review provides. We then meet with the developers to gain an appreciation of their vision of the software. We install and use the relevant software, exploring the user interactions and roles. While we do this, we brainstorm threat models and attack surfaces. We read design documentation, review other audit results, search for similar projects, examine source code dependencies, skim open issue tickets, and generally investigate details other than the implementation.

Documenting Results:

We follow a conservative, transparent process for analyzing potential security vulnerabilities and seeing them through successful remediation. Whenever a potential issue is discovered, we immediately create an Issue entry for it in this document, even though we have not yet verified the feasibility and impact of the issue. This process is conservative because we document our suspicions early even if they are later shown to not represent exploitable vulnerabilities. We generally follow a process of first documenting the suspicion with unresolved questions, then confirming the issue through code analysis, live experimentation, or automated tests. Code analysis is the most tentative, and we strive to provide test code, log captures, or screenshots demonstrating our confirmation. After this we analyze the feasibility of an attack in a live system.

Suggested Solutions:

We search for immediate mitigations that live deployments can take, and finally we suggest the requirements for remediation engineering for future releases. The mitigation and remediation recommendations should be scrutinized by the developers and deployment engineers, and successful mitigation and remediation is an ongoing collaborative process after we deliver our report, and before the details are made public.

Disclaimers

EtherAuthority.io Disclaimer

EtherAuthority team has analyzed this smart contract in accordance with the best industry practices at the date of this report, in relation to: cybersecurity vulnerabilities and issues in smart contract source code, the details of which are disclosed in this report, (Source Code); the Source Code compilation, deployment and functionality (performing the intended functions).

Due to the fact that the total number of test cases are unlimited, the audit makes no statements or warranties on security of the code. It also cannot be considered as a sufficient assessment regarding the utility and safety of the code, bugfree status or any other statements of the contract. While we have done our best in conducting the analysis and producing this report, it is important to note that you should not rely on this report only. We also suggest conducting a bug bounty program to confirm the high level of security of this smart contract.

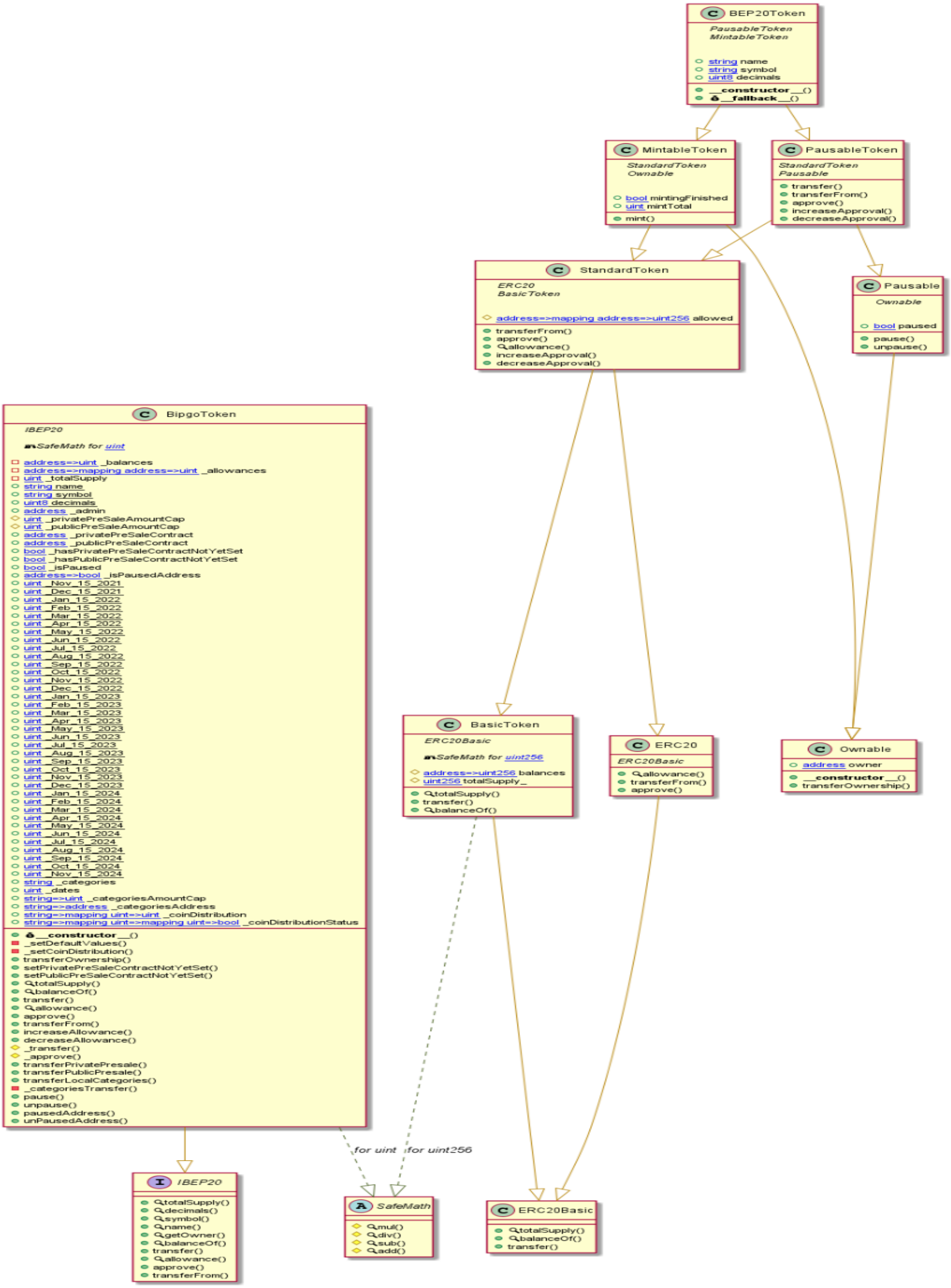
Technical Disclaimer

Smart contracts are deployed and executed on the blockchain platform. The platform, its programming language, and other software related to the smart contract can have their own vulnerabilities that can lead to hacks. Thus, the audit can’t guarantee explicit security of the audited smart contracts.

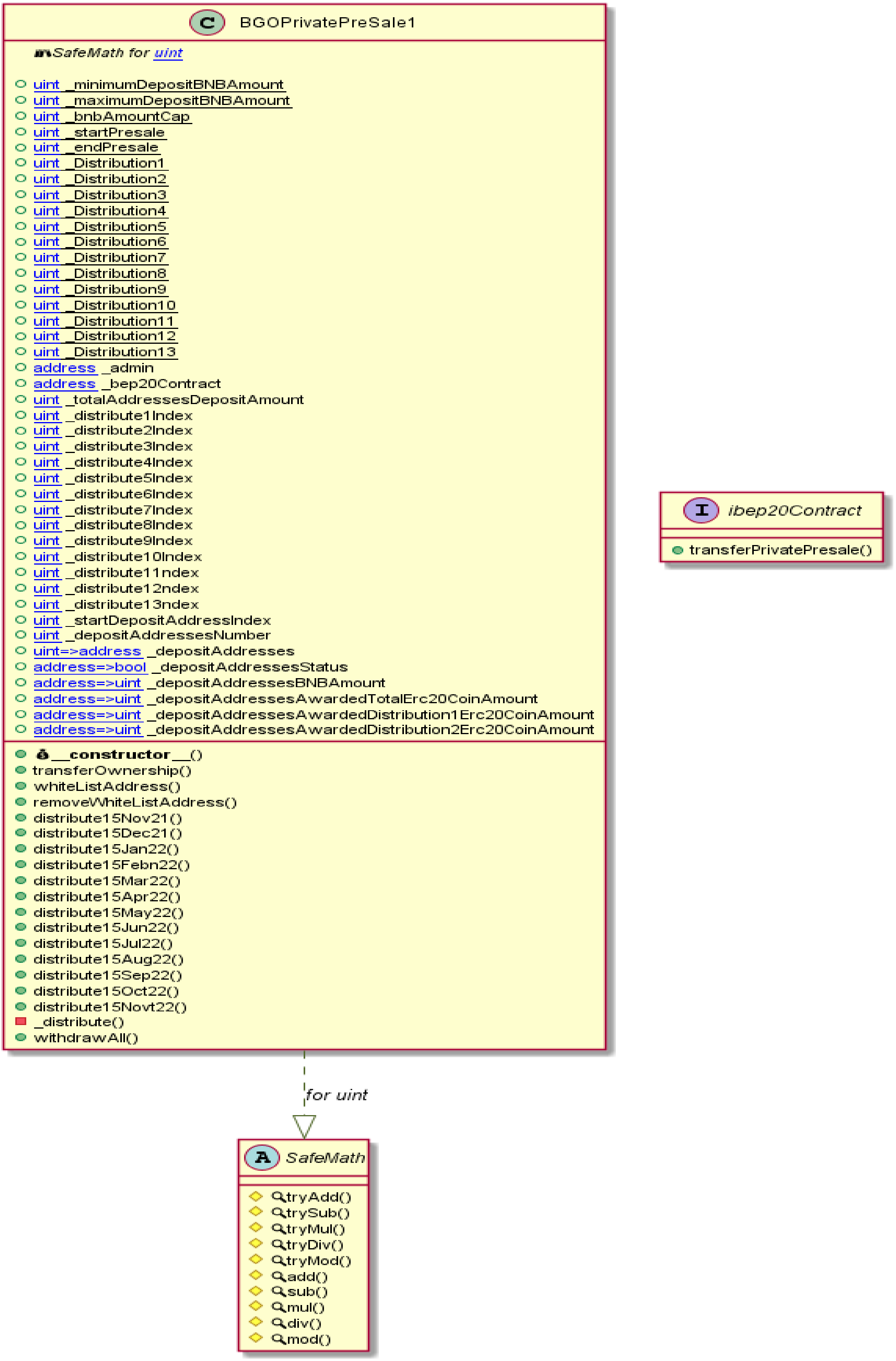
Appendix

Code Flow Diagram - BipGo Prototcol

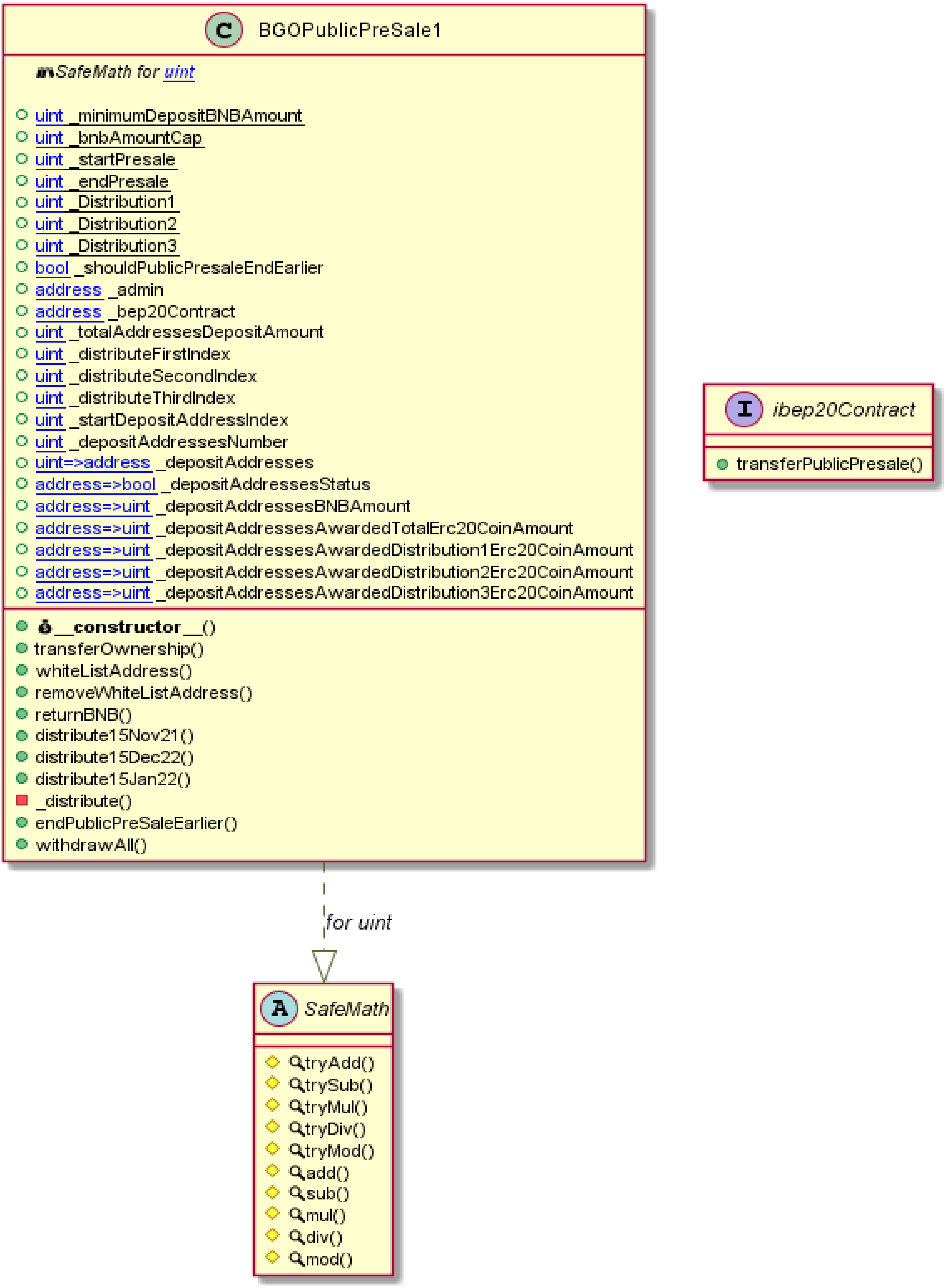
Token Diagram



PrivatePreSale Diagram



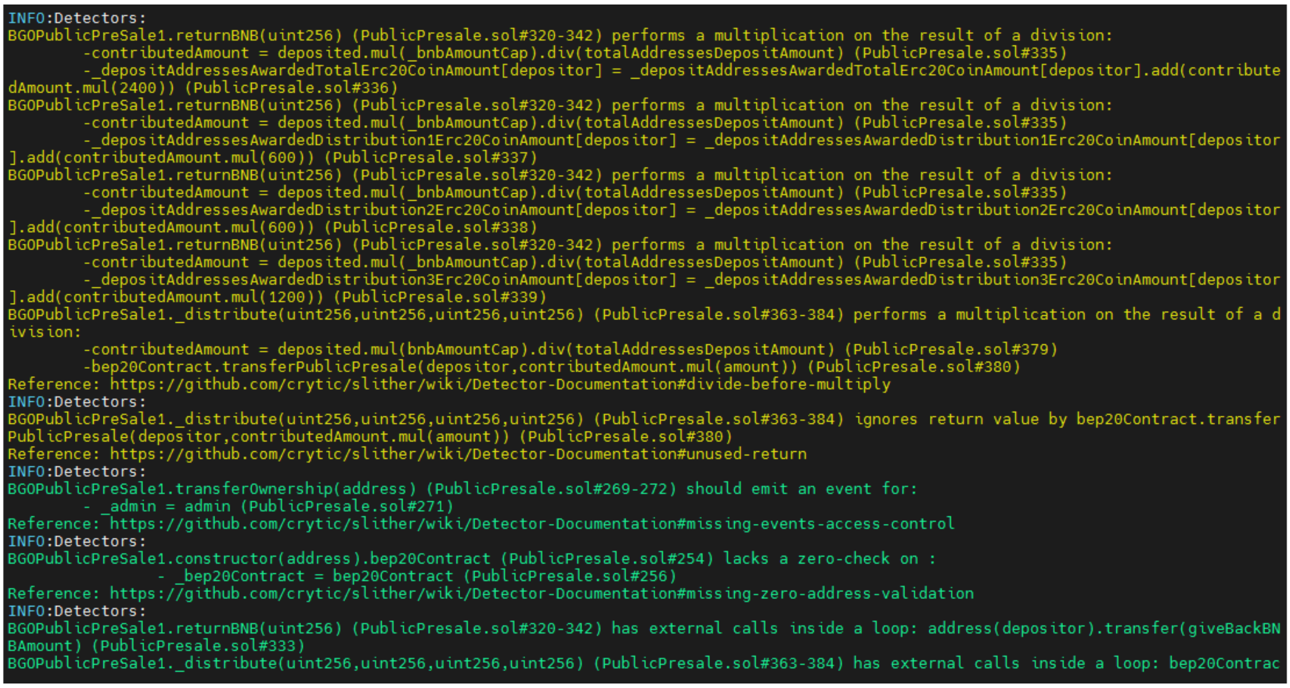
PublicPresale Diagram



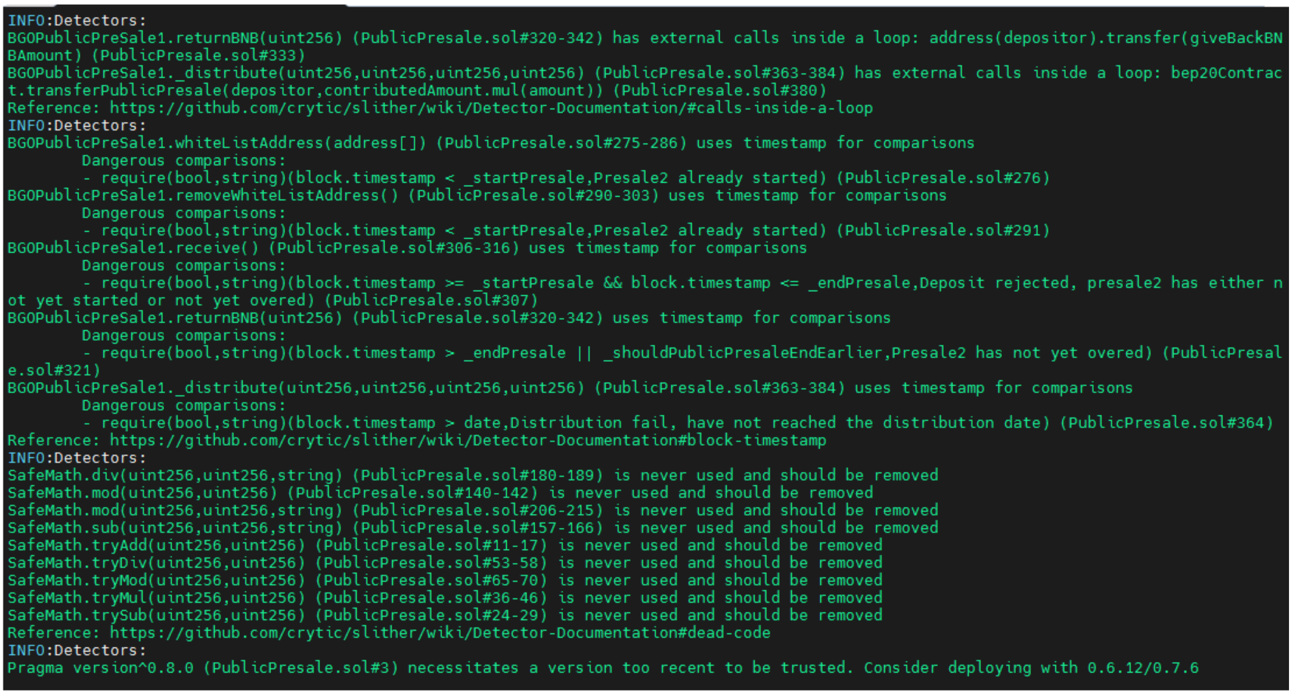
BGOPublicPreSale1
SafeMath for uint
uint _minimumDepositBNBAmount
uint _bnbAmountCap
uint _startPresale
uint _endPresale
uint _Distribution1
uint _Distribution2
uint _Distribution3
bool _shouldPublicPresaleEndEarlier
address _admin
address _bep20Contract
uint _totalAddressesDepositAmount
uint _distributeFirstIndex
uint _distributeSecondIndex
uint _distributeThirdIndex
uint _startDepositAddressIndex
uint _depositAddressesNumber
uint=>address _depositAddresses
address=>bool _depositAddressesStatus
address=>uint _depositAddressesBNBAmount
address=>uint _depositAddressesAwardedTotalErc20CoinAmount
address=>uint _depositAddressesAwardedDistribution1Erc20CoinAmount
address=>uint _depositAddressesAwardedDistribution2Erc20CoinAmount
address=>uint _depositAddressesAwardedDistribution3Erc20CoinAmount
__constructor__()
transferOwnership()
whiteListAddress()
removeWhiteListAddress()
returnBNB()
distribute15Nov21()
distribute15Dec22()
distribute15Jan22()
_distribute()
endPublicPreSaleEarlier()
withdrawAll()
ibep20Contract
transferPublicPresale()
for uint
SafeMath
tryAdd()
trySub()
tryMul()
tryDiv()
tryMod()
add()
sub()
mul()
div()
mod()

Slither Results Log

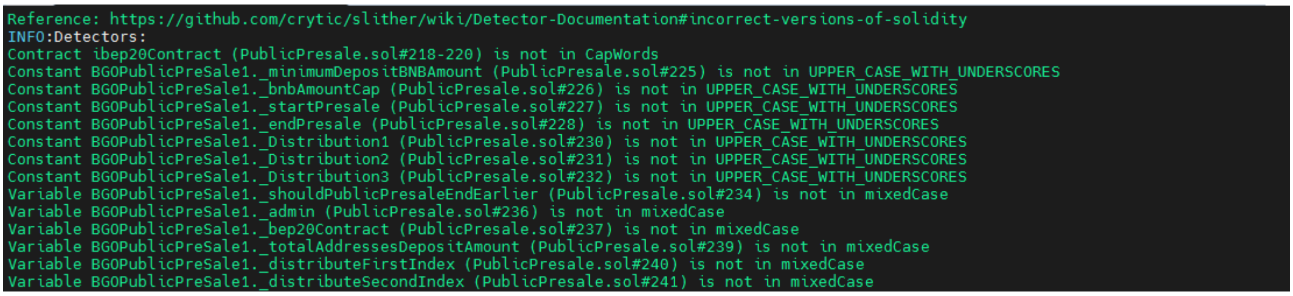
Slither log >> PublicPresale.sol



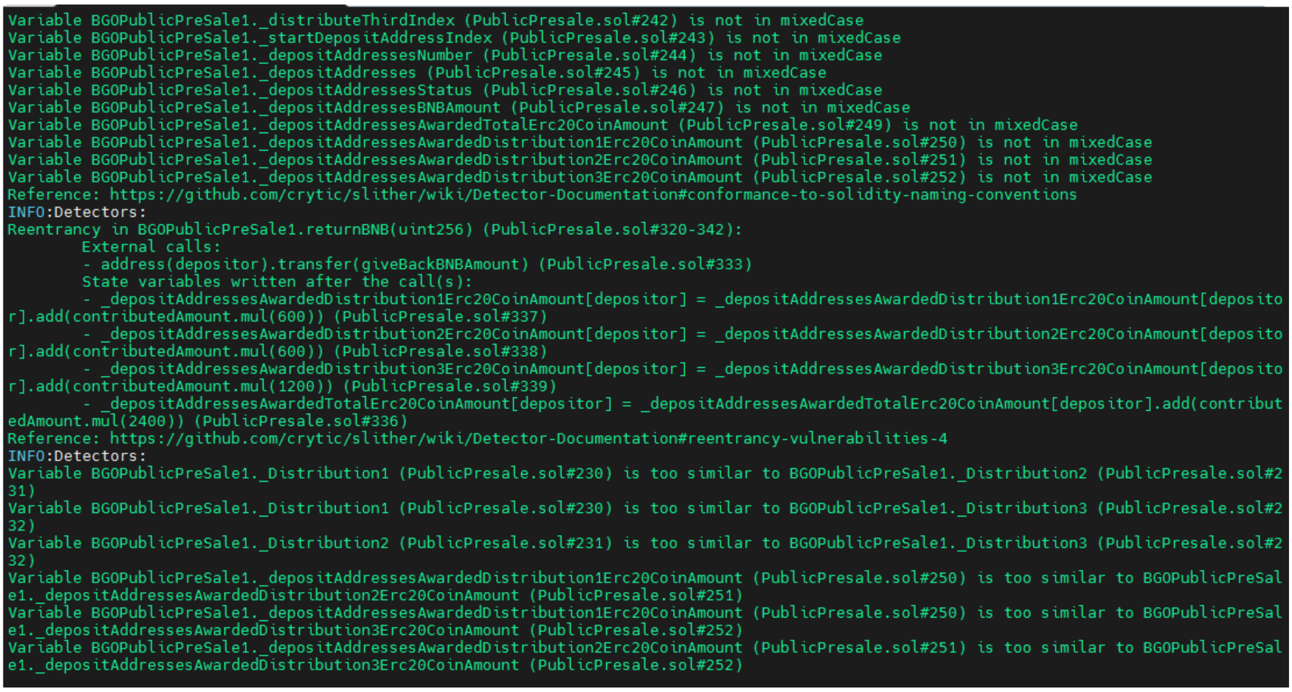
```
INFO:Detectors:
BGOPublicPreSale1.returnBNB(uint256) (PublicPresale.sol#320-342) performs a multiplication on the result of a division:
	-contributedAmount = deposited.mul(_bnbAmountCap).div(totalAddressesDepositAmount) (PublicPresale.sol#335)
	-_depositAddressesAwardedTotalErc20CoinAmount[depositor] = _depositAddressesAwardedTotalErc20CoinAmount[depositor].add(contribute
dAmount.mul(2400)) (PublicPresale.sol#336)
BGOPublicPreSale1.returnBNB(uint256) (PublicPresale.sol#320-342) performs a multiplication on the result of a division:
	-contributedAmount = deposited.mul(_bnbAmountCap).div(totalAddressesDepositAmount) (PublicPresale.sol#335)
	-_depositAddressesAwardedDistribution1Erc20CoinAmount[depositor] = _depositAddressesAwardedDistribution1Erc20CoinAmount[depositor
].add(contributedAmount.mul(600)) (PublicPresale.sol#337)
BGOPublicPreSale1.returnBNB(uint256) (PublicPresale.sol#320-342) performs a multiplication on the result of a division:
	-contributedAmount = deposited.mul(_bnbAmountCap).div(totalAddressesDepositAmount) (PublicPresale.sol#335)
	-_depositAddressesAwardedDistribution2Erc20CoinAmount[depositor] = _depositAddressesAwardedDistribution2Erc20CoinAmount[depositor
].add(contributedAmount.mul(600)) (PublicPresale.sol#338)
BGOPublicPreSale1.returnBNB(uint256) (PublicPresale.sol#320-342) performs a multiplication on the result of a division:
	-contributedAmount = deposited.mul(_bnbAmountCap).div(totalAddressesDepositAmount) (PublicPresale.sol#335)
	-_depositAddressesAwardedDistribution3Erc20CoinAmount[depositor] = _depositAddressesAwardedDistribution3Erc20CoinAmount[depositor
].add(contributedAmount.mul(1200)) (PublicPresale.sol#339)
BGOPublicPreSale1._distribute(uint256,uint256,uint256,uint256) (PublicPresale.sol#363-384) performs a multiplication on the result of a d
ivision:
	-contributedAmount = deposited.mul(bnbAmountCap).div(totalAddressesDepositAmount) (PublicPresale.sol#379)
	-bep20Contract.transferPublicPresale(depositor,contributedAmount.mul(amount)) (PublicPresale.sol#380)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#divide-before-multiply
INFO:Detectors:
BGOPublicPreSale1._distribute(uint256,uint256,uint256,uint256) (PublicPresale.sol#363-384) ignores return value by bep20Contract.transfer
PublicPresale(depositor,contributedAmount.mul(amount)) (PublicPresale.sol#380)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-return
INFO:Detectors:
BGOPublicPreSale1.transferOwnership(address) (PublicPresale.sol#269-272) should emit an event for:
	- _admin = admin (PublicPresale.sol#271)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-events-access-control
INFO:Detectors:
BGOPublicPreSale1.constructor(address).bep20Contract (PublicPresale.sol#254) lacks a zero-check on :
		- _bep20Contract = bep20Contract (PublicPresale.sol#256)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-address-validation
INFO:Detectors:
BGOPublicPreSale1.returnBNB(uint256) (PublicPresale.sol#320-342) has external calls inside a loop: address(depositor).transfer(giveBackBN
BAmount) (PublicPresale.sol#333)
BGOPublicPreSale1._distribute(uint256,uint256,uint256,uint256) (PublicPresale.sol#363-384) has external calls inside a loop: bep20Contrac
```



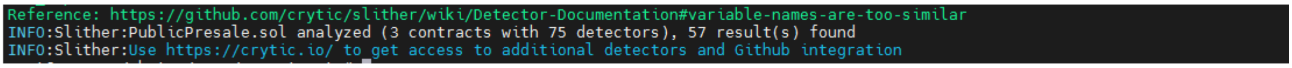
```
INFO:Detectors:
BGOPublicPreSale1.returnBNB(uint256) (PublicPresale.sol#320-342) has external calls inside a loop: address(depositor).transfer(giveBackBN
BAmount) (PublicPresale.sol#333)
BGOPublicPreSale1._distribute(uint256,uint256,uint256,uint256) (PublicPresale.sol#363-384) has external calls inside a loop: bep20Contrac
t.transferPublicPresale(depositor,contributedAmount.mul(amount)) (PublicPresale.sol#380)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation/#calls-inside-a-loop
INFO:Detectors:
BGOPublicPreSale1.whiteListAddress(address[]) (PublicPresale.sol#275-286) uses timestamp for comparisons
	Dangerous comparisons:
	- require(bool,string)(block.timestamp < _startPresale,Presale2 already started) (PublicPresale.sol#276)
BGOPublicPreSale1.removeWhiteListAddress() (PublicPresale.sol#290-303) uses timestamp for comparisons
	Dangerous comparisons:
	- require(bool,string)(block.timestamp < _startPresale,Presale2 already started) (PublicPresale.sol#291)
BGOPublicPreSale1.receive() (PublicPresale.sol#306-316) uses timestamp for comparisons
	Dangerous comparisons:
	- require(bool,string)(block.timestamp >= _startPresale && block.timestamp <= _endPresale,Deposit rejected, presale2 has either n
ot yet started or not yet overed) (PublicPresale.sol#307)
BGOPublicPreSale1.returnBNB(uint256) (PublicPresale.sol#320-342) uses timestamp for comparisons
	Dangerous comparisons:
	- require(bool,string)(block.timestamp > _endPresale || _shouldPublicPresaleEndEarlier,Presale2 has not yet overed) (PublicPresal
e.sol#321)
BGOPublicPreSale1._distribute(uint256,uint256,uint256,uint256) (PublicPresale.sol#363-384) uses timestamp for comparisons
	Dangerous comparisons:
	- require(bool,string)(block.timestamp > date,Distribution fail, have not reached the distribution date) (PublicPresale.sol#364)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-timestamp
INFO:Detectors:
SafeMath.div(uint256,uint256,string) (PublicPresale.sol#180-189) is never used and should be removed
SafeMath.mod(uint256,uint256) (PublicPresale.sol#140-142) is never used and should be removed
SafeMath.mod(uint256,uint256,string) (PublicPresale.sol#206-215) is never used and should be removed
SafeMath.sub(uint256,uint256,string) (PublicPresale.sol#157-166) is never used and should be removed
SafeMath.tryAdd(uint256,uint256) (PublicPresale.sol#11-17) is never used and should be removed
SafeMath.tryDiv(uint256,uint256) (PublicPresale.sol#53-58) is never used and should be removed
SafeMath.tryMod(uint256,uint256) (PublicPresale.sol#65-70) is never used and should be removed
SafeMath.tryMul(uint256,uint256) (PublicPresale.sol#36-46) is never used and should be removed
SafeMath.trySub(uint256,uint256) (PublicPresale.sol#24-29) is never used and should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
Pragma version^0.8.0 (PublicPresale.sol#3) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
```



```
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
Contract ibep20Contract (PublicPresale.sol#218-220) is not in CapWords
Constant BGOPublicPreSale1._minimumDepositBNBAmount (PublicPresale.sol#225) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPublicPreSale1._bnbAmountCap (PublicPresale.sol#226) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPublicPreSale1._startPresale (PublicPresale.sol#227) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPublicPreSale1._endPresale (PublicPresale.sol#228) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPublicPreSale1._Distribution1 (PublicPresale.sol#230) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPublicPreSale1._Distribution2 (PublicPresale.sol#231) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPublicPreSale1._Distribution3 (PublicPresale.sol#232) is not in UPPER_CASE_WITH_UNDERSCORES
Variable BGOPublicPreSale1._shouldPublicPresaleEndEarlier (PublicPresale.sol#234) is not in mixedCase
Variable BGOPublicPreSale1._admin (PublicPresale.sol#236) is not in mixedCase
Variable BGOPublicPreSale1._bep20Contract (PublicPresale.sol#237) is not in mixedCase
Variable BGOPublicPreSale1._totalAddressesDepositAmount (PublicPresale.sol#239) is not in mixedCase
Variable BGOPublicPreSale1._distributeFirstIndex (PublicPresale.sol#240) is not in mixedCase
Variable BGOPublicPreSale1._distributeSecondIndex (PublicPresale.sol#241) is not in mixedCase
```

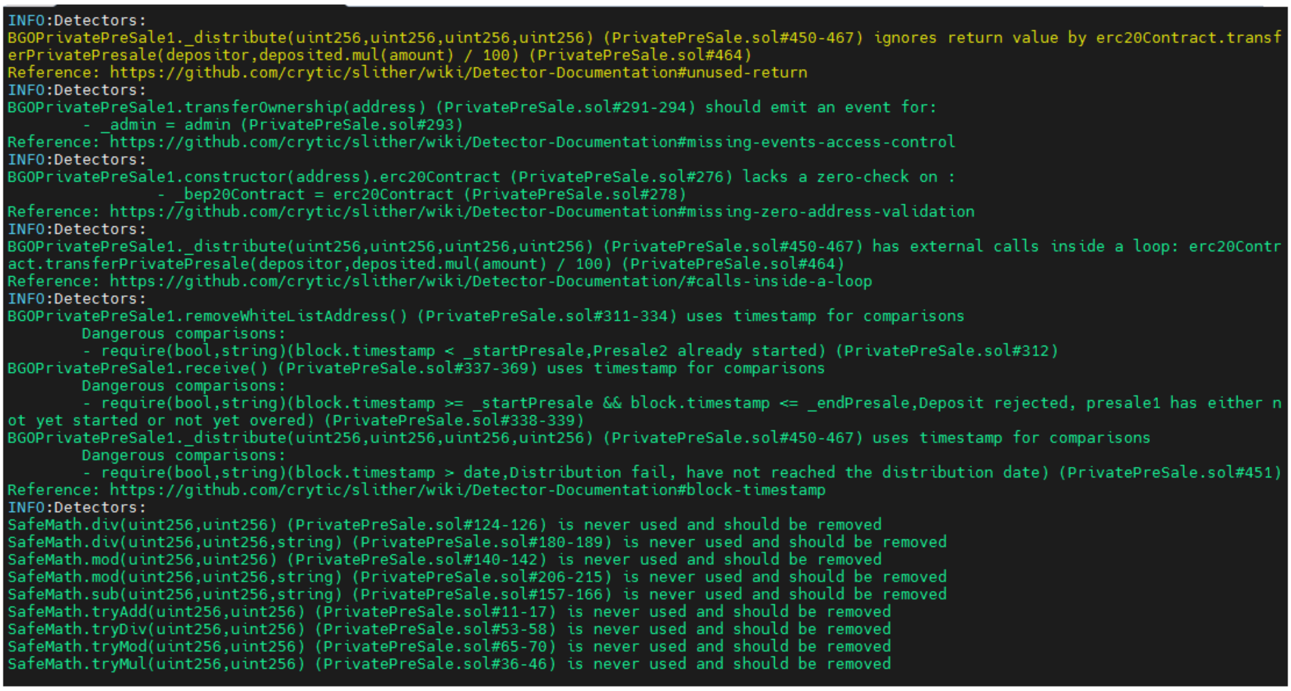


```
Variable BGOPublicPreSale1._distributeThirdIndex (PublicPresale.sol#242) is not in mixedCase
Variable BGOPublicPreSale1._startDepositAddressIndex (PublicPresale.sol#243) is not in mixedCase
Variable BGOPublicPreSale1._depositAddressesNumber (PublicPresale.sol#244) is not in mixedCase
Variable BGOPublicPreSale1._depositAddresses (PublicPresale.sol#245) is not in mixedCase
Variable BGOPublicPreSale1._depositAddressesStatus (PublicPresale.sol#246) is not in mixedCase
Variable BGOPublicPreSale1._depositAddressesBNBAmount (PublicPresale.sol#247) is not in mixedCase
Variable BGOPublicPreSale1._depositAddressesAwardedTotalErc20CoinAmount (PublicPresale.sol#249) is not in mixedCase
Variable BGOPublicPreSale1._depositAddressesAwardedDistribution1Erc20CoinAmount (PublicPresale.sol#250) is not in mixedCase
Variable BGOPublicPreSale1._depositAddressesAwardedDistribution2Erc20CoinAmount (PublicPresale.sol#251) is not in mixedCase
Variable BGOPublicPreSale1._depositAddressesAwardedDistribution3Erc20CoinAmount (PublicPresale.sol#252) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
INFO:Detectors:
Reentrancy in BGOPublicPreSale1.returnBNB(uint256) (PublicPresale.sol#320-342):
        External calls:
        - address(depositor).transfer(giveBackBNBAmount) (PublicPresale.sol#333)
        State variables written after the call(s):
        - _depositAddressesAwardedDistribution1Erc20CoinAmount[depositor] = _depositAddressesAwardedDistribution1Erc20CoinAmount[deposito
r].add(contributedAmount.mul(600)) (PublicPresale.sol#337)
        - _depositAddressesAwardedDistribution2Erc20CoinAmount[depositor] = _depositAddressesAwardedDistribution2Erc20CoinAmount[deposito
r].add(contributedAmount.mul(600)) (PublicPresale.sol#338)
        - _depositAddressesAwardedDistribution3Erc20CoinAmount[depositor] = _depositAddressesAwardedDistribution3Erc20CoinAmount[deposito
r].add(contributedAmount.mul(1200)) (PublicPresale.sol#339)
        - _depositAddressesAwardedTotalErc20CoinAmount[depositor] = _depositAddressesAwardedTotalErc20CoinAmount[depositor].add(contribut
edAmount.mul(2400)) (PublicPresale.sol#336)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-4
INFO:Detectors:
Variable BGOPublicPreSale1._Distribution1 (PublicPresale.sol#230) is too similar to BGOPublicPreSale1._Distribution2 (PublicPresale.sol#2
31)
Variable BGOPublicPreSale1._Distribution1 (PublicPresale.sol#230) is too similar to BGOPublicPreSale1._Distribution3 (PublicPresale.sol#2
32)
Variable BGOPublicPreSale1._Distribution2 (PublicPresale.sol#231) is too similar to BGOPublicPreSale1._Distribution3 (PublicPresale.sol#2
32)
Variable BGOPublicPreSale1._depositAddressesAwardedDistribution1Erc20CoinAmount (PublicPresale.sol#250) is too similar to BGOPublicPreSal
e1._depositAddressesAwardedDistribution2Erc20CoinAmount (PublicPresale.sol#251)
Variable BGOPublicPreSale1._depositAddressesAwardedDistribution1Erc20CoinAmount (PublicPresale.sol#250) is too similar to BGOPublicPreSal
e1._depositAddressesAwardedDistribution3Erc20CoinAmount (PublicPresale.sol#252)
Variable BGOPublicPreSale1._depositAddressesAwardedDistribution2Erc20CoinAmount (PublicPresale.sol#251) is too similar to BGOPublicPreSal
e1._depositAddressesAwardedDistribution3Erc20CoinAmount (PublicPresale.sol#252)
```

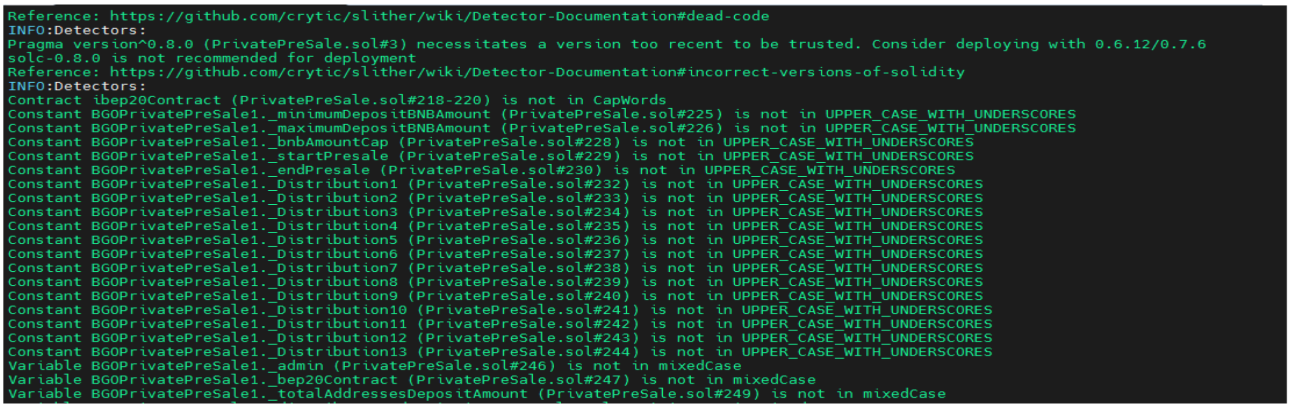


```
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-are-too-similar
INFO:Slither:PublicPresale.sol analyzed (3 contracts with 75 detectors), 57 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and Github integration
```

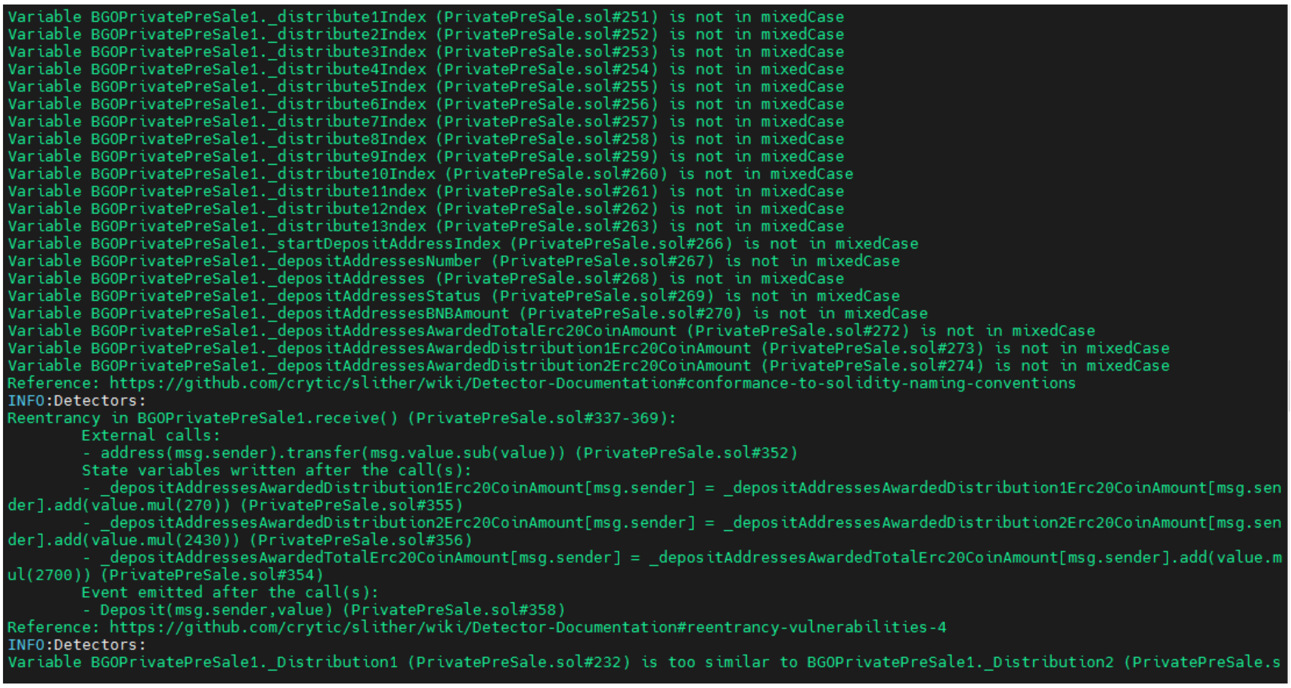
Slither log >> PrivatePreSale.sol



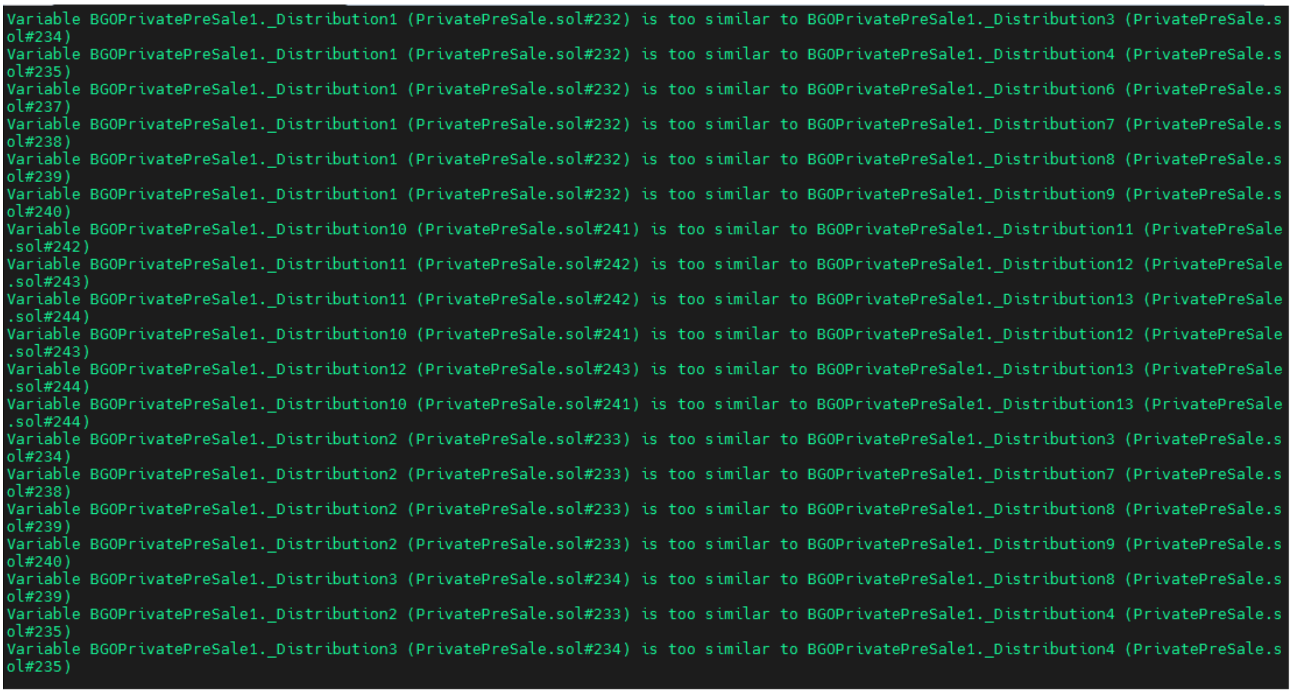
```
INFO:Detectors:
BGOPrivatePreSale1._distribute(uint256,uint256,uint256,uint256) (PrivatePreSale.sol#450-467) ignores return value by erc20Contract.transf
erPrivatePresale(depositor,deposited.mul(amount) / 100) (PrivatePreSale.sol#464)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-return
INFO:Detectors:
BGOPrivatePreSale1.transferOwnership(address) (PrivatePreSale.sol#291-294) should emit an event for:
        - _admin = admin (PrivatePreSale.sol#293)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-events-access-control
INFO:Detectors:
BGOPrivatePreSale1.constructor(address).erc20Contract (PrivatePreSale.sol#276) lacks a zero-check on :
                - _bep20Contract = erc20Contract (PrivatePreSale.sol#278)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-address-validation
INFO:Detectors:
BGOPrivatePreSale1._distribute(uint256,uint256,uint256,uint256) (PrivatePreSale.sol#450-467) has external calls inside a loop: erc20Contr
act.transferPrivatePresale(depositor,deposited.mul(amount) / 100) (PrivatePreSale.sol#464)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation/#calls-inside-a-loop
INFO:Detectors:
BGOPrivatePreSale1.removeWhiteListAddress() (PrivatePreSale.sol#311-334) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(block.timestamp < _startPresale,Presale2 already started) (PrivatePreSale.sol#312)
BGOPrivatePreSale1.receive() (PrivatePreSale.sol#337-369) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(block.timestamp >= _startPresale && block.timestamp <= _endPresale,Deposit rejected, presale1 has either n
ot yet started or not yet overed) (PrivatePreSale.sol#338-339)
BGOPrivatePreSale1._distribute(uint256,uint256,uint256,uint256) (PrivatePreSale.sol#450-467) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(block.timestamp > date,Distribution fail, have not reached the distribution date) (PrivatePreSale.sol#451)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-timestamp
INFO:Detectors:
SafeMath.div(uint256,uint256) (PrivatePreSale.sol#124-126) is never used and should be removed
SafeMath.div(uint256,uint256,string) (PrivatePreSale.sol#180-189) is never used and should be removed
SafeMath.mod(uint256,uint256) (PrivatePreSale.sol#140-142) is never used and should be removed
SafeMath.mod(uint256,uint256,string) (PrivatePreSale.sol#206-215) is never used and should be removed
SafeMath.sub(uint256,uint256,string) (PrivatePreSale.sol#157-166) is never used and should be removed
SafeMath.tryAdd(uint256,uint256) (PrivatePreSale.sol#11-17) is never used and should be removed
SafeMath.tryDiv(uint256,uint256) (PrivatePreSale.sol#53-58) is never used and should be removed
SafeMath.tryMod(uint256,uint256) (PrivatePreSale.sol#65-70) is never used and should be removed
SafeMath.tryMul(uint256,uint256) (PrivatePreSale.sol#36-46) is never used and should be removed
```



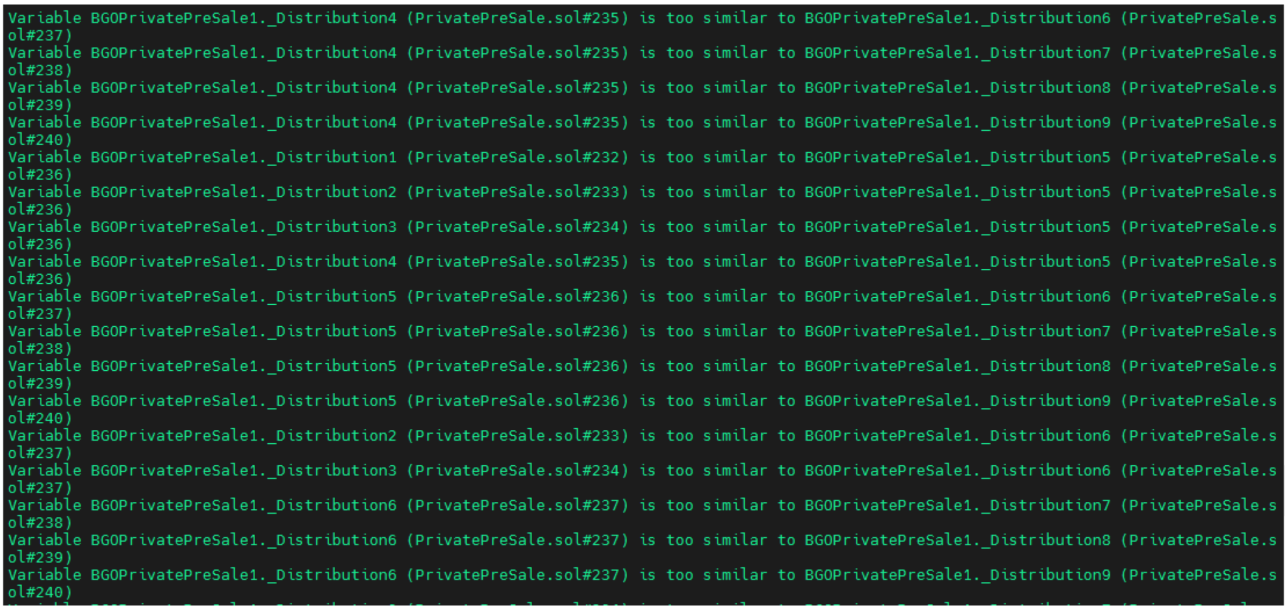
```
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
Pragma version^0.8.0 (PrivatePreSale.sol#3) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
solc-0.8.0 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
Contract ibep20Contract (PrivatePreSale.sol#218-220) is not in CapWords
Constant BGOPrivatePreSale1._minimumDepositBNBAmount (PrivatePreSale.sol#225) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._maximumDepositBNBAmount (PrivatePreSale.sol#226) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._bnbAmountCap (PrivatePreSale.sol#228) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._startPresale (PrivatePreSale.sol#229) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._endPresale (PrivatePreSale.sol#230) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution1 (PrivatePreSale.sol#232) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution2 (PrivatePreSale.sol#233) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution3 (PrivatePreSale.sol#234) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution4 (PrivatePreSale.sol#235) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution5 (PrivatePreSale.sol#236) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution6 (PrivatePreSale.sol#237) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution7 (PrivatePreSale.sol#238) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution8 (PrivatePreSale.sol#239) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution9 (PrivatePreSale.sol#240) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution10 (PrivatePreSale.sol#241) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution11 (PrivatePreSale.sol#242) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution12 (PrivatePreSale.sol#243) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BGOPrivatePreSale1._Distribution13 (PrivatePreSale.sol#244) is not in UPPER_CASE_WITH_UNDERSCORES
Variable BGOPrivatePreSale1._admin (PrivatePreSale.sol#246) is not in mixedCase
Variable BGOPrivatePreSale1._bep20Contract (PrivatePreSale.sol#247) is not in mixedCase
Variable BGOPrivatePreSale1._totalAddressesDepositAmount (PrivatePreSale.sol#249) is not in mixedCase
```



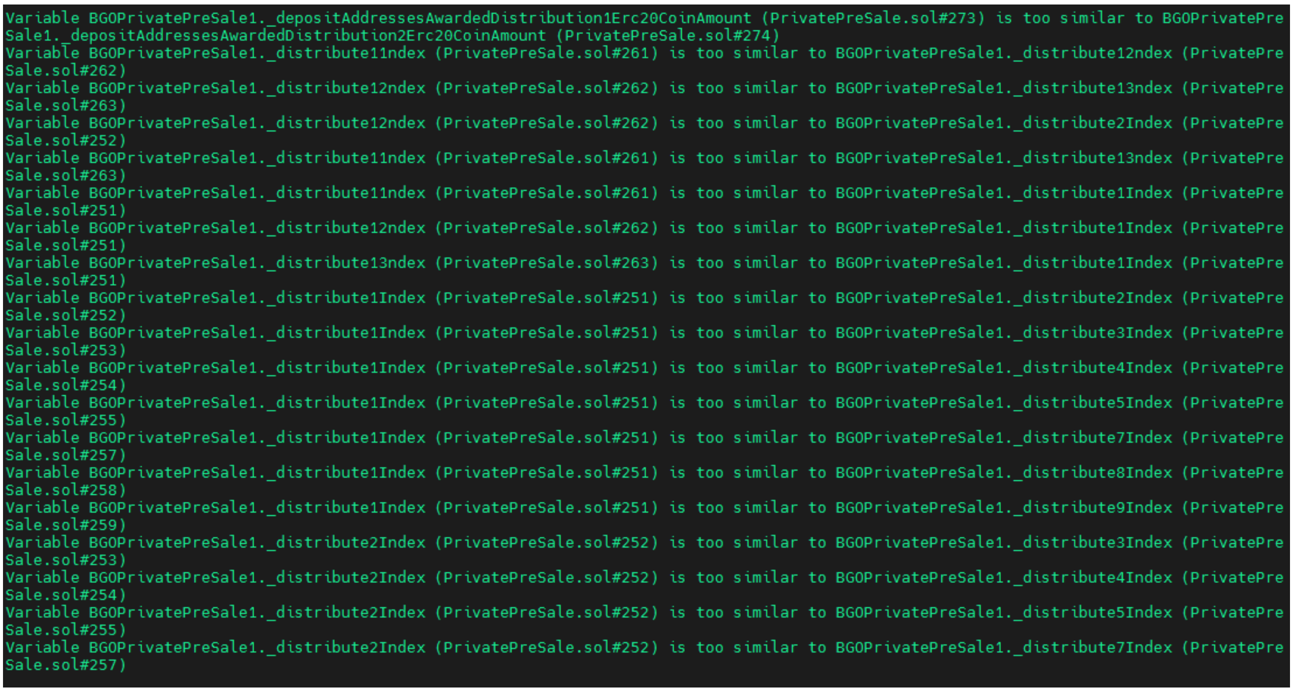
```
Variable BGOPrivatePreSale1._distribute1Index (PrivatePreSale.sol#251) is not in mixedCase
Variable BGOPrivatePreSale1._distribute2Index (PrivatePreSale.sol#252) is not in mixedCase
Variable BGOPrivatePreSale1._distribute3Index (PrivatePreSale.sol#253) is not in mixedCase
Variable BGOPrivatePreSale1._distribute4Index (PrivatePreSale.sol#254) is not in mixedCase
Variable BGOPrivatePreSale1._distribute5Index (PrivatePreSale.sol#255) is not in mixedCase
Variable BGOPrivatePreSale1._distribute6Index (PrivatePreSale.sol#256) is not in mixedCase
Variable BGOPrivatePreSale1._distribute7Index (PrivatePreSale.sol#257) is not in mixedCase
Variable BGOPrivatePreSale1._distribute8Index (PrivatePreSale.sol#258) is not in mixedCase
Variable BGOPrivatePreSale1._distribute9Index (PrivatePreSale.sol#259) is not in mixedCase
Variable BGOPrivatePreSale1._distribute10Index (PrivatePreSale.sol#260) is not in mixedCase
Variable BGOPrivatePreSale1._distribute11ndex (PrivatePreSale.sol#261) is not in mixedCase
Variable BGOPrivatePreSale1._distribute12ndex (PrivatePreSale.sol#262) is not in mixedCase
Variable BGOPrivatePreSale1._distribute13ndex (PrivatePreSale.sol#263) is not in mixedCase
Variable BGOPrivatePreSale1._startDepositAddressIndex (PrivatePreSale.sol#266) is not in mixedCase
Variable BGOPrivatePreSale1._depositAddressesNumber (PrivatePreSale.sol#267) is not in mixedCase
Variable BGOPrivatePreSale1._depositAddresses (PrivatePreSale.sol#268) is not in mixedCase
Variable BGOPrivatePreSale1._depositAddressesStatus (PrivatePreSale.sol#269) is not in mixedCase
Variable BGOPrivatePreSale1._depositAddressesBNBAmount (PrivatePreSale.sol#270) is not in mixedCase
Variable BGOPrivatePreSale1._depositAddressesAwardedTotalErc20CoinAmount (PrivatePreSale.sol#272) is not in mixedCase
Variable BGOPrivatePreSale1._depositAddressesAwardedDistribution1Erc20CoinAmount (PrivatePreSale.sol#273) is not in mixedCase
Variable BGOPrivatePreSale1._depositAddressesAwardedDistribution2Erc20CoinAmount (PrivatePreSale.sol#274) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
INFO:Detectors:
Reentrancy in BGOPrivatePreSale1.receive() (PrivatePreSale.sol#337-369):
        External calls:
        - address(msg.sender).transfer(msg.value.sub(value)) (PrivatePreSale.sol#352)
        State variables written after the call(s):
        - _depositAddressesAwardedDistribution1Erc20CoinAmount[msg.sender] = _depositAddressesAwardedDistribution1Erc20CoinAmount[msg.sen
der].add(value.mul(270)) (PrivatePreSale.sol#355)
        - _depositAddressesAwardedDistribution2Erc20CoinAmount[msg.sender] = _depositAddressesAwardedDistribution2Erc20CoinAmount[msg.sen
der].add(value.mul(2430)) (PrivatePreSale.sol#356)
        - _depositAddressesAwardedTotalErc20CoinAmount[msg.sender] = _depositAddressesAwardedTotalErc20CoinAmount[msg.sender].add(value.m
ul(2700)) (PrivatePreSale.sol#354)
        Event emitted after the call(s):
        - Deposit(msg.sender,value) (PrivatePreSale.sol#358)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-4
INFO:Detectors:
Variable BGOPrivatePreSale1._Distribution1 (PrivatePreSale.sol#232) is too similar to BGOPrivatePreSale1._Distribution2 (PrivatePreSale.s
```



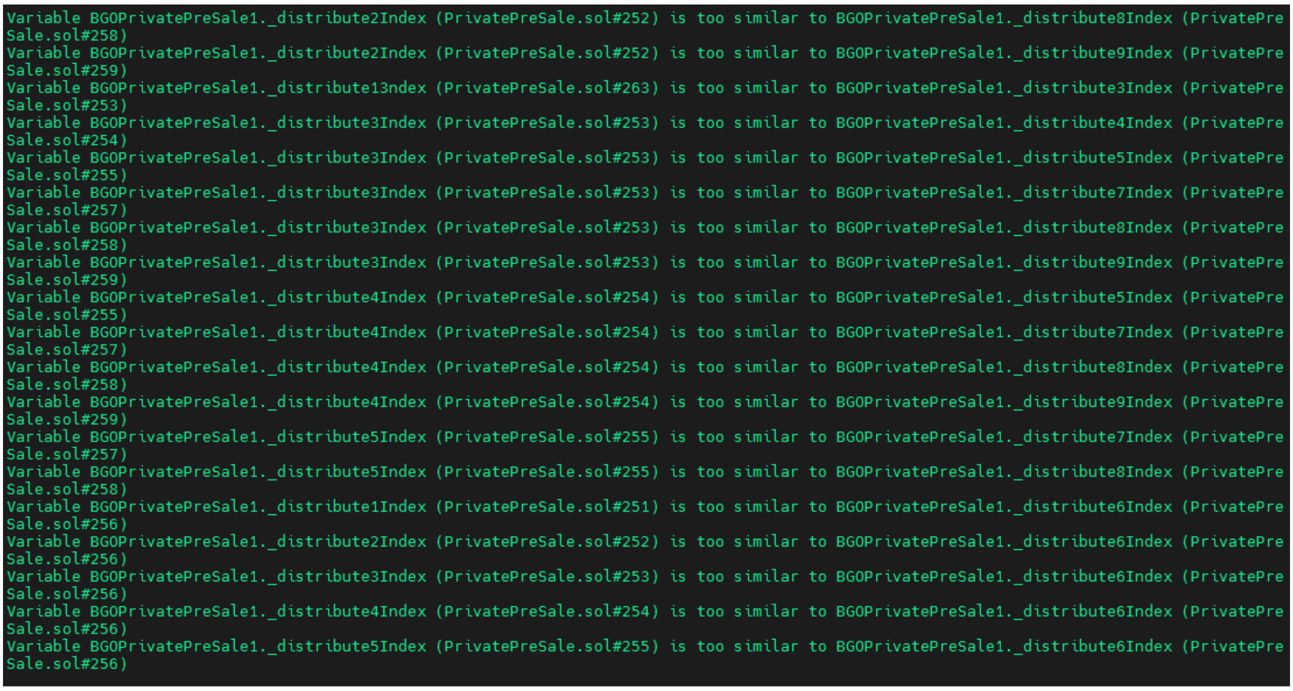
```
Variable BGOPrivatePreSale1._Distribution1 (PrivatePreSale.sol#232) is too similar to BGOPrivatePreSale1._Distribution3 (PrivatePreSale.s
ol#234)
Variable BGOPrivatePreSale1._Distribution1 (PrivatePreSale.sol#232) is too similar to BGOPrivatePreSale1._Distribution4 (PrivatePreSale.s
ol#235)
Variable BGOPrivatePreSale1._Distribution1 (PrivatePreSale.sol#232) is too similar to BGOPrivatePreSale1._Distribution6 (PrivatePreSale.s
ol#237)
Variable BGOPrivatePreSale1._Distribution1 (PrivatePreSale.sol#232) is too similar to BGOPrivatePreSale1._Distribution7 (PrivatePreSale.s
ol#238)
Variable BGOPrivatePreSale1._Distribution1 (PrivatePreSale.sol#232) is too similar to BGOPrivatePreSale1._Distribution8 (PrivatePreSale.s
ol#239)
Variable BGOPrivatePreSale1._Distribution1 (PrivatePreSale.sol#232) is too similar to BGOPrivatePreSale1._Distribution9 (PrivatePreSale.s
ol#240)
Variable BGOPrivatePreSale1._Distribution10 (PrivatePreSale.sol#241) is too similar to BGOPrivatePreSale1._Distribution11 (PrivatePreSale
.sol#242)
Variable BGOPrivatePreSale1._Distribution11 (PrivatePreSale.sol#242) is too similar to BGOPrivatePreSale1._Distribution12 (PrivatePreSale
.sol#243)
Variable BGOPrivatePreSale1._Distribution11 (PrivatePreSale.sol#242) is too similar to BGOPrivatePreSale1._Distribution13 (PrivatePreSale
.sol#244)
Variable BGOPrivatePreSale1._Distribution10 (PrivatePreSale.sol#241) is too similar to BGOPrivatePreSale1._Distribution12 (PrivatePreSale
.sol#243)
Variable BGOPrivatePreSale1._Distribution12 (PrivatePreSale.sol#243) is too similar to BGOPrivatePreSale1._Distribution13 (PrivatePreSale
.sol#244)
Variable BGOPrivatePreSale1._Distribution10 (PrivatePreSale.sol#241) is too similar to BGOPrivatePreSale1._Distribution13 (PrivatePreSale
.sol#244)
Variable BGOPrivatePreSale1._Distribution2 (PrivatePreSale.sol#233) is too similar to BGOPrivatePreSale1._Distribution3 (PrivatePreSale.s
ol#234)
Variable BGOPrivatePreSale1._Distribution2 (PrivatePreSale.sol#233) is too similar to BGOPrivatePreSale1._Distribution7 (PrivatePreSale.s
ol#238)
Variable BGOPrivatePreSale1._Distribution2 (PrivatePreSale.sol#233) is too similar to BGOPrivatePreSale1._Distribution8 (PrivatePreSale.s
ol#239)
Variable BGOPrivatePreSale1._Distribution2 (PrivatePreSale.sol#233) is too similar to BGOPrivatePreSale1._Distribution9 (PrivatePreSale.s
ol#240)
Variable BGOPrivatePreSale1._Distribution3 (PrivatePreSale.sol#234) is too similar to BGOPrivatePreSale1._Distribution8 (PrivatePreSale.s
ol#239)
Variable BGOPrivatePreSale1._Distribution2 (PrivatePreSale.sol#233) is too similar to BGOPrivatePreSale1._Distribution4 (PrivatePreSale.s
ol#235)
Variable BGOPrivatePreSale1._Distribution3 (PrivatePreSale.sol#234) is too similar to BGOPrivatePreSale1._Distribution4 (PrivatePreSale.s
ol#235)
```



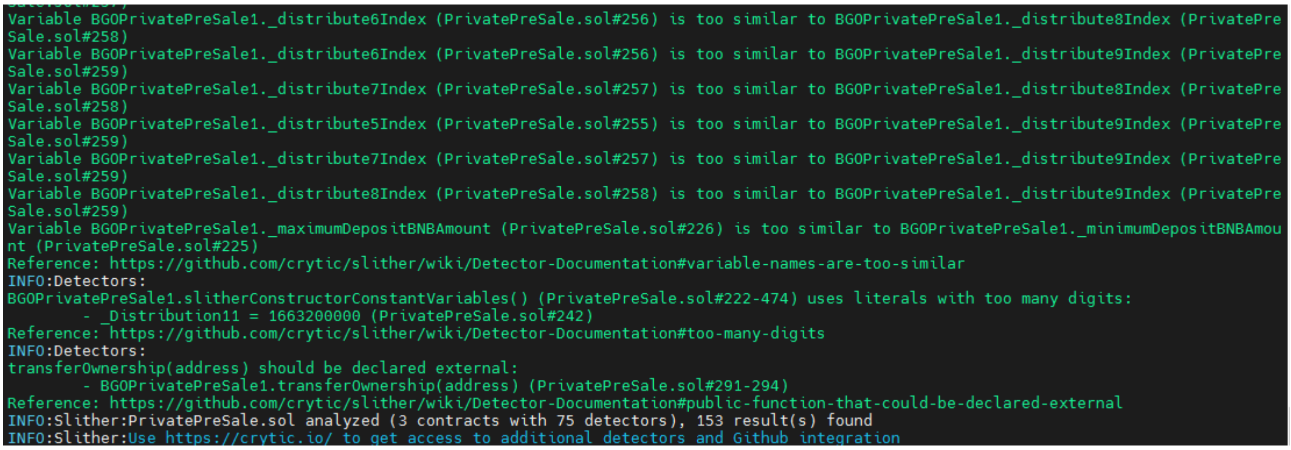
```
Variable BGOPrivatePreSale1._Distribution4 (PrivatePreSale.sol#235) is too similar to BGOPrivatePreSale1._Distribution6 (PrivatePreSale.s
ol#237)
Variable BGOPrivatePreSale1._Distribution4 (PrivatePreSale.sol#235) is too similar to BGOPrivatePreSale1._Distribution7 (PrivatePreSale.s
ol#238)
Variable BGOPrivatePreSale1._Distribution4 (PrivatePreSale.sol#235) is too similar to BGOPrivatePreSale1._Distribution8 (PrivatePreSale.s
ol#239)
Variable BGOPrivatePreSale1._Distribution4 (PrivatePreSale.sol#235) is too similar to BGOPrivatePreSale1._Distribution9 (PrivatePreSale.s
ol#240)
Variable BGOPrivatePreSale1._Distribution1 (PrivatePreSale.sol#232) is too similar to BGOPrivatePreSale1._Distribution5 (PrivatePreSale.s
ol#236)
Variable BGOPrivatePreSale1._Distribution2 (PrivatePreSale.sol#233) is too similar to BGOPrivatePreSale1._Distribution5 (PrivatePreSale.s
ol#236)
Variable BGOPrivatePreSale1._Distribution3 (PrivatePreSale.sol#234) is too similar to BGOPrivatePreSale1._Distribution5 (PrivatePreSale.s
ol#236)
Variable BGOPrivatePreSale1._Distribution4 (PrivatePreSale.sol#235) is too similar to BGOPrivatePreSale1._Distribution5 (PrivatePreSale.s
ol#236)
Variable BGOPrivatePreSale1._Distribution5 (PrivatePreSale.sol#236) is too similar to BGOPrivatePreSale1._Distribution6 (PrivatePreSale.s
ol#237)
Variable BGOPrivatePreSale1._Distribution5 (PrivatePreSale.sol#236) is too similar to BGOPrivatePreSale1._Distribution7 (PrivatePreSale.s
ol#238)
Variable BGOPrivatePreSale1._Distribution5 (PrivatePreSale.sol#236) is too similar to BGOPrivatePreSale1._Distribution8 (PrivatePreSale.s
ol#239)
Variable BGOPrivatePreSale1._Distribution5 (PrivatePreSale.sol#236) is too similar to BGOPrivatePreSale1._Distribution9 (PrivatePreSale.s
ol#240)
Variable BGOPrivatePreSale1._Distribution2 (PrivatePreSale.sol#233) is too similar to BGOPrivatePreSale1._Distribution6 (PrivatePreSale.s
ol#237)
Variable BGOPrivatePreSale1._Distribution3 (PrivatePreSale.sol#234) is too similar to BGOPrivatePreSale1._Distribution6 (PrivatePreSale.s
ol#237)
Variable BGOPrivatePreSale1._Distribution6 (PrivatePreSale.sol#237) is too similar to BGOPrivatePreSale1._Distribution7 (PrivatePreSale.s
ol#238)
Variable BGOPrivatePreSale1._Distribution6 (PrivatePreSale.sol#237) is too similar to BGOPrivatePreSale1._Distribution8 (PrivatePreSale.s
ol#239)
Variable BGOPrivatePreSale1._Distribution6 (PrivatePreSale.sol#237) is too similar to BGOPrivatePreSale1._Distribution9 (PrivatePreSale.s
ol#240)
```



```
Variable BGOPrivatePreSale1._depositAddressesAwardedDistribution1Erc20CoinAmount (PrivatePreSale.sol#273) is too similar to BGOPrivatePre
Sale1._depositAddressesAwardedDistribution2Erc20CoinAmount (PrivatePreSale.sol#274)
Variable BGOPrivatePreSale1._distribute11ndex (PrivatePreSale.sol#261) is too similar to BGOPrivatePreSale1._distribute12ndex (PrivatePre
Sale.sol#262)
Variable BGOPrivatePreSale1._distribute12ndex (PrivatePreSale.sol#262) is too similar to BGOPrivatePreSale1._distribute13ndex (PrivatePre
Sale.sol#263)
Variable BGOPrivatePreSale1._distribute12ndex (PrivatePreSale.sol#262) is too similar to BGOPrivatePreSale1._distribute2Index (PrivatePre
Sale.sol#252)
Variable BGOPrivatePreSale1._distribute11ndex (PrivatePreSale.sol#261) is too similar to BGOPrivatePreSale1._distribute13ndex (PrivatePre
Sale.sol#263)
Variable BGOPrivatePreSale1._distribute11ndex (PrivatePreSale.sol#261) is too similar to BGOPrivatePreSale1._distribute1Index (PrivatePre
Sale.sol#251)
Variable BGOPrivatePreSale1._distribute12ndex (PrivatePreSale.sol#262) is too similar to BGOPrivatePreSale1._distribute1Index (PrivatePre
Sale.sol#251)
Variable BGOPrivatePreSale1._distribute13ndex (PrivatePreSale.sol#263) is too similar to BGOPrivatePreSale1._distribute1Index (PrivatePre
Sale.sol#251)
Variable BGOPrivatePreSale1._distribute1Index (PrivatePreSale.sol#251) is too similar to BGOPrivatePreSale1._distribute2Index (PrivatePre
Sale.sol#252)
Variable BGOPrivatePreSale1._distribute1Index (PrivatePreSale.sol#251) is too similar to BGOPrivatePreSale1._distribute3Index (PrivatePre
Sale.sol#253)
Variable BGOPrivatePreSale1._distribute1Index (PrivatePreSale.sol#251) is too similar to BGOPrivatePreSale1._distribute4Index (PrivatePre
Sale.sol#254)
Variable BGOPrivatePreSale1._distribute1Index (PrivatePreSale.sol#251) is too similar to BGOPrivatePreSale1._distribute5Index (PrivatePre
Sale.sol#255)
Variable BGOPrivatePreSale1._distribute1Index (PrivatePreSale.sol#251) is too similar to BGOPrivatePreSale1._distribute7Index (PrivatePre
Sale.sol#257)
Variable BGOPrivatePreSale1._distribute1Index (PrivatePreSale.sol#251) is too similar to BGOPrivatePreSale1._distribute8Index (PrivatePre
Sale.sol#258)
Variable BGOPrivatePreSale1._distribute1Index (PrivatePreSale.sol#251) is too similar to BGOPrivatePreSale1._distribute9Index (PrivatePre
Sale.sol#259)
Variable BGOPrivatePreSale1._distribute2Index (PrivatePreSale.sol#252) is too similar to BGOPrivatePreSale1._distribute3Index (PrivatePre
Sale.sol#253)
Variable BGOPrivatePreSale1._distribute2Index (PrivatePreSale.sol#252) is too similar to BGOPrivatePreSale1._distribute4Index (PrivatePre
Sale.sol#254)
Variable BGOPrivatePreSale1._distribute2Index (PrivatePreSale.sol#252) is too similar to BGOPrivatePreSale1._distribute5Index (PrivatePre
Sale.sol#255)
Variable BGOPrivatePreSale1._distribute2Index (PrivatePreSale.sol#252) is too similar to BGOPrivatePreSale1._distribute7Index (PrivatePre
Sale.sol#257)
```

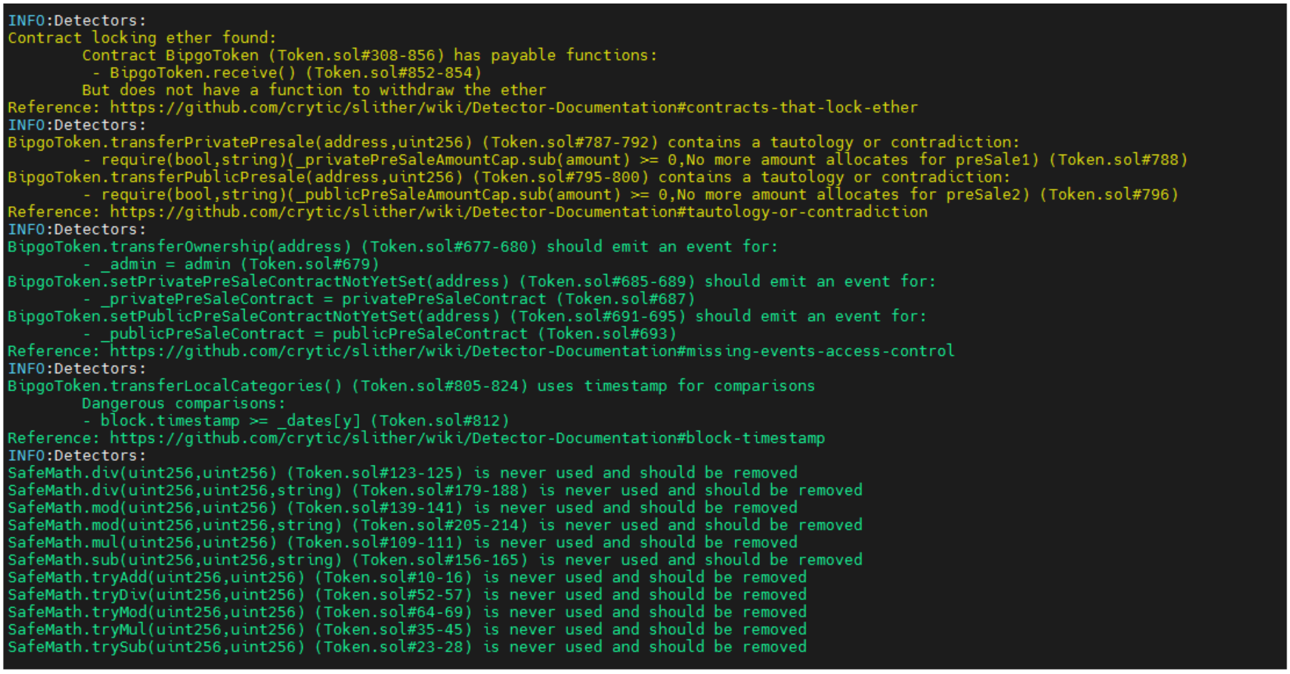


```
Variable BGOPrivatePreSale1._distribute2Index (PrivatePreSale.sol#252) is too similar to BGOPrivatePreSale1._distribute8Index (PrivatePre
Sale.sol#258)
Variable BGOPrivatePreSale1._distribute2Index (PrivatePreSale.sol#252) is too similar to BGOPrivatePreSale1._distribute9Index (PrivatePre
Sale.sol#259)
Variable BGOPrivatePreSale1._distribute13ndex (PrivatePreSale.sol#263) is too similar to BGOPrivatePreSale1._distribute3Index (PrivatePre
Sale.sol#253)
Variable BGOPrivatePreSale1._distribute3Index (PrivatePreSale.sol#253) is too similar to BGOPrivatePreSale1._distribute4Index (PrivatePre
Sale.sol#254)
Variable BGOPrivatePreSale1._distribute3Index (PrivatePreSale.sol#253) is too similar to BGOPrivatePreSale1._distribute5Index (PrivatePre
Sale.sol#255)
Variable BGOPrivatePreSale1._distribute3Index (PrivatePreSale.sol#253) is too similar to BGOPrivatePreSale1._distribute7Index (PrivatePre
Sale.sol#257)
Variable BGOPrivatePreSale1._distribute3Index (PrivatePreSale.sol#253) is too similar to BGOPrivatePreSale1._distribute8Index (PrivatePre
Sale.sol#258)
Variable BGOPrivatePreSale1._distribute3Index (PrivatePreSale.sol#253) is too similar to BGOPrivatePreSale1._distribute9Index (PrivatePre
Sale.sol#259)
Variable BGOPrivatePreSale1._distribute4Index (PrivatePreSale.sol#254) is too similar to BGOPrivatePreSale1._distribute5Index (PrivatePre
Sale.sol#255)
Variable BGOPrivatePreSale1._distribute4Index (PrivatePreSale.sol#254) is too similar to BGOPrivatePreSale1._distribute7Index (PrivatePre
Sale.sol#257)
Variable BGOPrivatePreSale1._distribute4Index (PrivatePreSale.sol#254) is too similar to BGOPrivatePreSale1._distribute8Index (PrivatePre
Sale.sol#258)
Variable BGOPrivatePreSale1._distribute4Index (PrivatePreSale.sol#254) is too similar to BGOPrivatePreSale1._distribute9Index (PrivatePre
Sale.sol#259)
Variable BGOPrivatePreSale1._distribute5Index (PrivatePreSale.sol#255) is too similar to BGOPrivatePreSale1._distribute7Index (PrivatePre
Sale.sol#257)
Variable BGOPrivatePreSale1._distribute5Index (PrivatePreSale.sol#255) is too similar to BGOPrivatePreSale1._distribute8Index (PrivatePre
Sale.sol#258)
Variable BGOPrivatePreSale1._distribute1Index (PrivatePreSale.sol#251) is too similar to BGOPrivatePreSale1._distribute6Index (PrivatePre
Sale.sol#256)
Variable BGOPrivatePreSale1._distribute2Index (PrivatePreSale.sol#252) is too similar to BGOPrivatePreSale1._distribute6Index (PrivatePre
Sale.sol#256)
Variable BGOPrivatePreSale1._distribute3Index (PrivatePreSale.sol#253) is too similar to BGOPrivatePreSale1._distribute6Index (PrivatePre
Sale.sol#256)
Variable BGOPrivatePreSale1._distribute4Index (PrivatePreSale.sol#254) is too similar to BGOPrivatePreSale1._distribute6Index (PrivatePre
Sale.sol#256)
Variable BGOPrivatePreSale1._distribute5Index (PrivatePreSale.sol#255) is too similar to BGOPrivatePreSale1._distribute6Index (PrivatePre
Sale.sol#256)
```

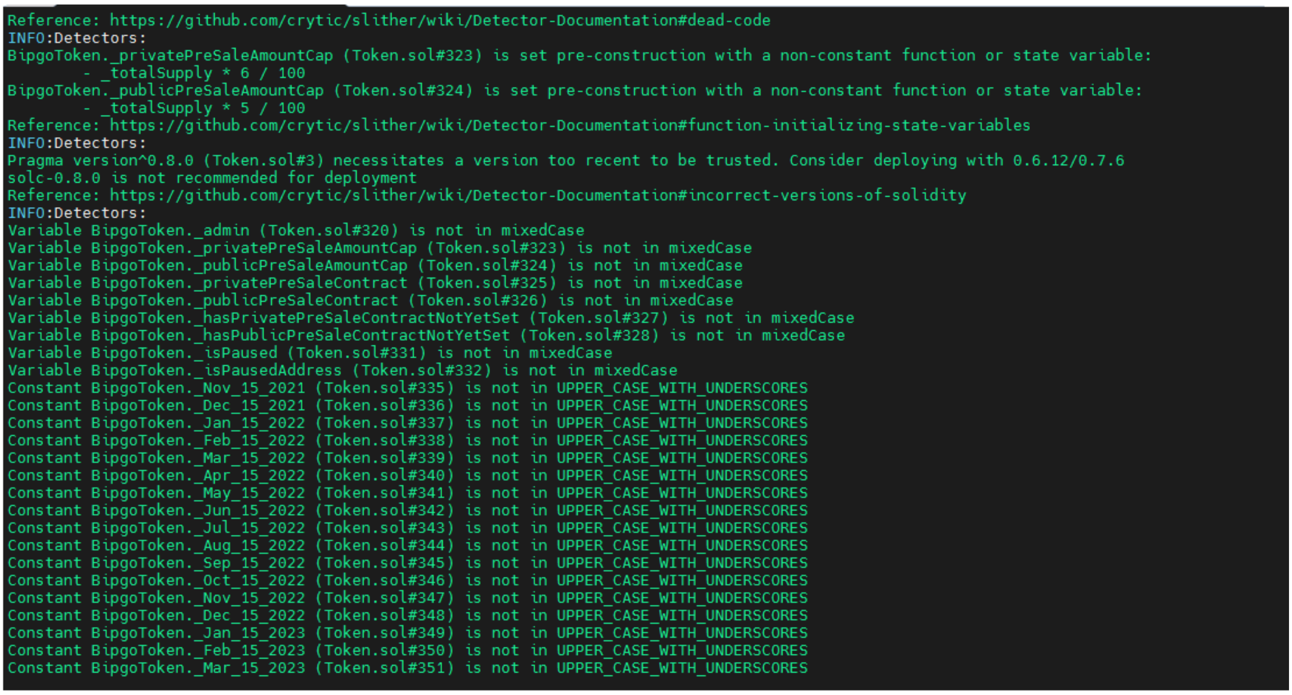


```
Variable BGOPrivatePreSale1._distribute6Index (PrivatePreSale.sol#256) is too similar to BGOPrivatePreSale1._distribute8Index (PrivatePre
Sale.sol#258)
Variable BGOPrivatePreSale1._distribute6Index (PrivatePreSale.sol#256) is too similar to BGOPrivatePreSale1._distribute9Index (PrivatePre
Sale.sol#259)
Variable BGOPrivatePreSale1._distribute7Index (PrivatePreSale.sol#257) is too similar to BGOPrivatePreSale1._distribute8Index (PrivatePre
Sale.sol#258)
Variable BGOPrivatePreSale1._distribute5Index (PrivatePreSale.sol#255) is too similar to BGOPrivatePreSale1._distribute9Index (PrivatePre
Sale.sol#259)
Variable BGOPrivatePreSale1._distribute7Index (PrivatePreSale.sol#257) is too similar to BGOPrivatePreSale1._distribute9Index (PrivatePre
Sale.sol#259)
Variable BGOPrivatePreSale1._distribute8Index (PrivatePreSale.sol#258) is too similar to BGOPrivatePreSale1._distribute9Index (PrivatePre
Sale.sol#259)
Variable BGOPrivatePreSale1._maximumDepositBNBAmount (PrivatePreSale.sol#226) is too similar to BGOPrivatePreSale1._minimumDepositBNBAmou
nt (PrivatePreSale.sol#225)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-are-too-similar
INFO:Detectors:
BGOPrivatePreSale1.slitherConstructorConstantVariables() (PrivatePreSale.sol#222-474) uses literals with too many digits:
        - _Distribution11 = 1663200000 (PrivatePreSale.sol#242)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits
INFO:Detectors:
transferOwnership(address) should be declared external:
        - BGOPrivatePreSale1.transferOwnership(address) (PrivatePreSale.sol#291-294)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-function-that-could-be-declared-external
INFO:Slither:PrivatePreSale.sol analyzed (3 contracts with 75 detectors), 153 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and Github integration
```

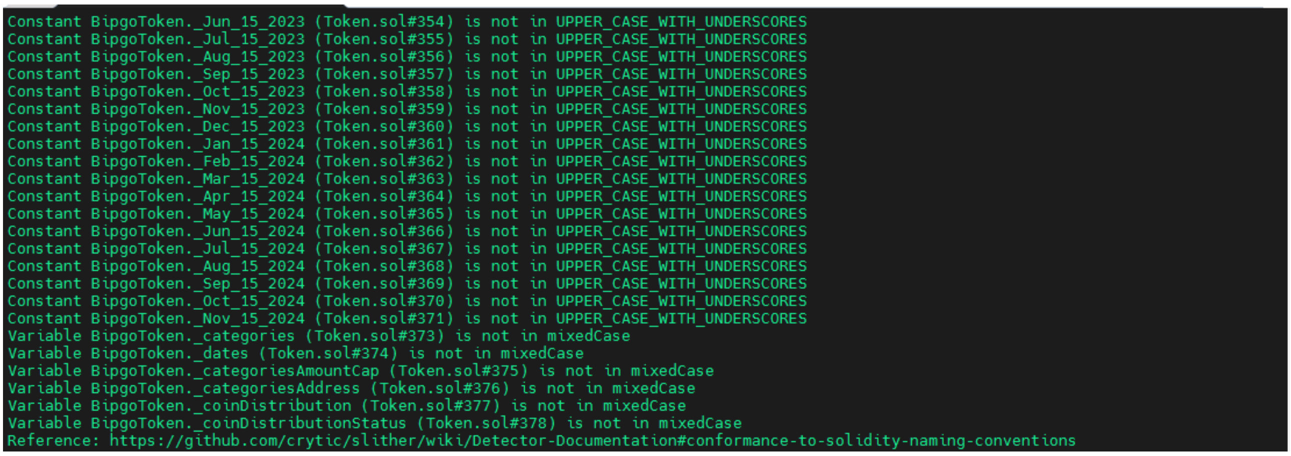
Slither log >> Token.sol



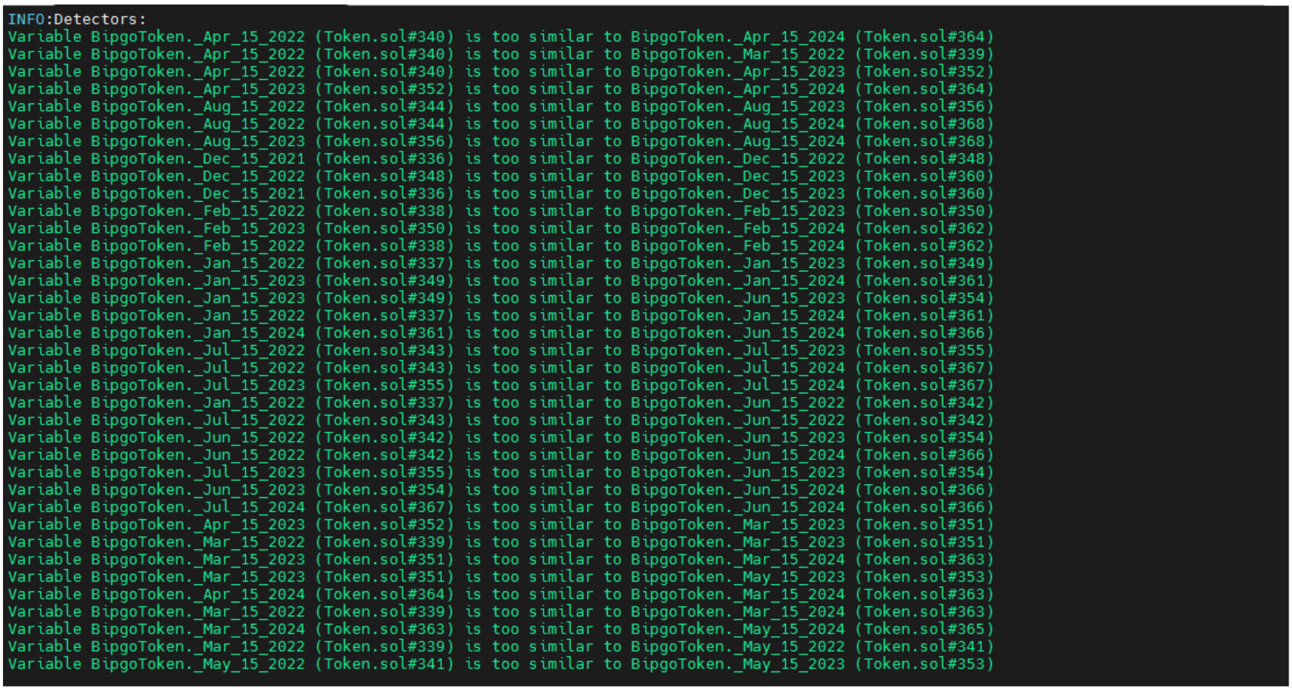
```
INFO:Detectors:
Contract locking ether found:
        Contract BipgoToken (Token.sol#308-856) has payable functions:
         - BipgoToken.receive() (Token.sol#852-854)
        But does not have a function to withdraw the ether
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#contracts-that-lock-ether
INFO:Detectors:
BipgoToken.transferPrivatePresale(address,uint256) (Token.sol#787-792) contains a tautology or contradiction:
        - require(bool,string)(_privatePreSaleAmountCap.sub(amount) >= 0,No more amount allocates for preSale1) (Token.sol#788)
BipgoToken.transferPublicPresale(address,uint256) (Token.sol#795-800) contains a tautology or contradiction:
        - require(bool,string)(_publicPreSaleAmountCap.sub(amount) >= 0,No more amount allocates for preSale2) (Token.sol#796)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#tautology-or-contradiction
INFO:Detectors:
BipgoToken.transferOwnership(address) (Token.sol#677-680) should emit an event for:
        - _admin = admin (Token.sol#679)
BipgoToken.setPrivatePreSaleContractNotYetSet(address) (Token.sol#685-689) should emit an event for:
        - _privatePreSaleContract = privatePreSaleContract (Token.sol#687)
BipgoToken.setPublicPreSaleContractNotYetSet(address) (Token.sol#691-695) should emit an event for:
        - _publicPreSaleContract = publicPreSaleContract (Token.sol#693)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-events-access-control
INFO:Detectors:
BipgoToken.transferLocalCategories() (Token.sol#805-824) uses timestamp for comparisons
        Dangerous comparisons:
        - block.timestamp >= _dates[y] (Token.sol#812)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-timestamp
INFO:Detectors:
SafeMath.div(uint256,uint256) (Token.sol#123-125) is never used and should be removed
SafeMath.div(uint256,uint256,string) (Token.sol#179-188) is never used and should be removed
SafeMath.mod(uint256,uint256) (Token.sol#139-141) is never used and should be removed
SafeMath.mod(uint256,uint256,string) (Token.sol#205-214) is never used and should be removed
SafeMath.mul(uint256,uint256) (Token.sol#109-111) is never used and should be removed
SafeMath.sub(uint256,uint256,string) (Token.sol#156-165) is never used and should be removed
SafeMath.tryAdd(uint256,uint256) (Token.sol#10-16) is never used and should be removed
SafeMath.tryDiv(uint256,uint256) (Token.sol#52-57) is never used and should be removed
SafeMath.tryMod(uint256,uint256) (Token.sol#64-69) is never used and should be removed
SafeMath.tryMul(uint256,uint256) (Token.sol#35-45) is never used and should be removed
SafeMath.trySub(uint256,uint256) (Token.sol#23-28) is never used and should be removed
```



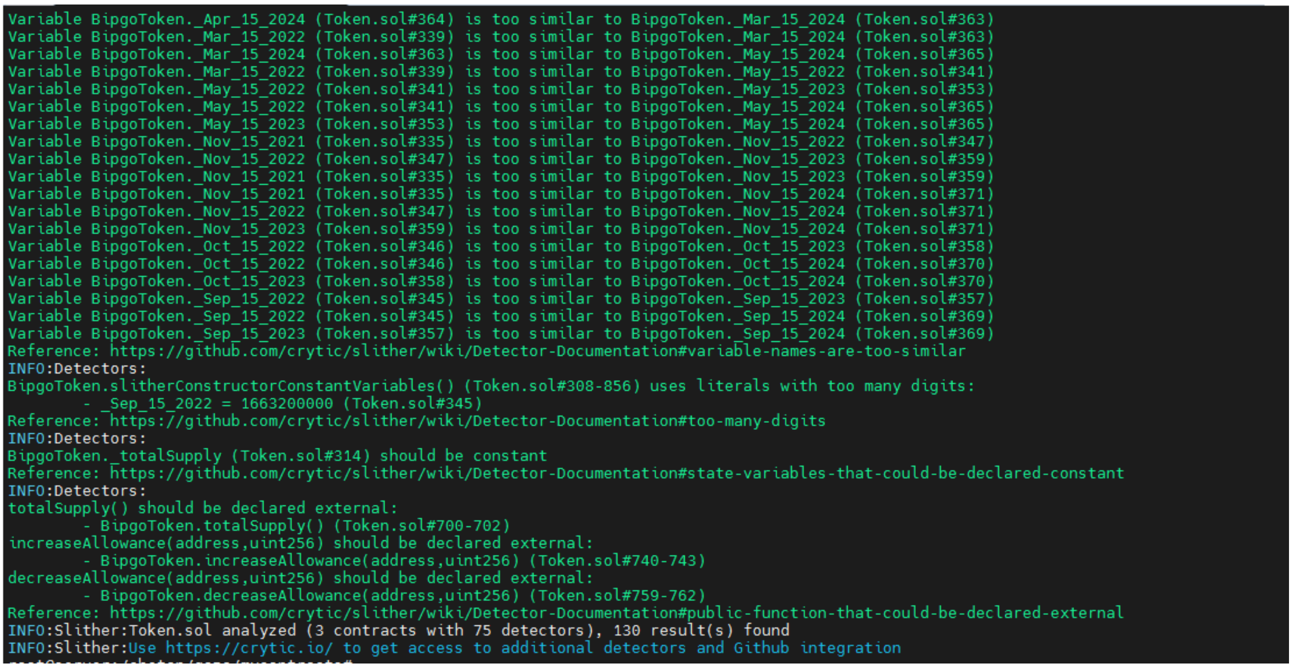
```
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
BipgoToken._privatePreSaleAmountCap (Token.sol#323) is set pre-construction with a non-constant function or state variable:
        - _totalSupply * 6 / 100
BipgoToken._publicPreSaleAmountCap (Token.sol#324) is set pre-construction with a non-constant function or state variable:
        - _totalSupply * 5 / 100
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#function-initializing-state-variables
INFO:Detectors:
Pragma version^0.8.0 (Token.sol#3) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
solc-0.8.0 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
Variable BipgoToken._admin (Token.sol#320) is not in mixedCase
Variable BipgoToken._privatePreSaleAmountCap (Token.sol#323) is not in mixedCase
Variable BipgoToken._publicPreSaleAmountCap (Token.sol#324) is not in mixedCase
Variable BipgoToken._privatePreSaleContract (Token.sol#325) is not in mixedCase
Variable BipgoToken._publicPreSaleContract (Token.sol#326) is not in mixedCase
Variable BipgoToken._hasPrivatePreSaleContractNotYetSet (Token.sol#327) is not in mixedCase
Variable BipgoToken._hasPublicPreSaleContractNotYetSet (Token.sol#328) is not in mixedCase
Variable BipgoToken._isPaused (Token.sol#331) is not in mixedCase
Variable BipgoToken._isPausedAddress (Token.sol#332) is not in mixedCase
Constant BipgoToken._Nov_15_2021 (Token.sol#335) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Dec_15_2021 (Token.sol#336) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Jan_15_2022 (Token.sol#337) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Feb_15_2022 (Token.sol#338) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Mar_15_2022 (Token.sol#339) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Apr_15_2022 (Token.sol#340) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._May_15_2022 (Token.sol#341) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Jun_15_2022 (Token.sol#342) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Jul_15_2022 (Token.sol#343) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Aug_15_2022 (Token.sol#344) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Sep_15_2022 (Token.sol#345) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Oct_15_2022 (Token.sol#346) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Nov_15_2022 (Token.sol#347) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Dec_15_2022 (Token.sol#348) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Jan_15_2023 (Token.sol#349) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Feb_15_2023 (Token.sol#350) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Mar_15_2023 (Token.sol#351) is not in UPPER_CASE_WITH_UNDERSCORES
```



```
Constant BipgoToken._Jun_15_2023 (Token.sol#354) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Jul_15_2023 (Token.sol#355) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Aug_15_2023 (Token.sol#356) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Sep_15_2023 (Token.sol#357) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Oct_15_2023 (Token.sol#358) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Nov_15_2023 (Token.sol#359) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Dec_15_2023 (Token.sol#360) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Jan_15_2024 (Token.sol#361) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Feb_15_2024 (Token.sol#362) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Mar_15_2024 (Token.sol#363) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Apr_15_2024 (Token.sol#364) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._May_15_2024 (Token.sol#365) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Jun_15_2024 (Token.sol#366) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Jul_15_2024 (Token.sol#367) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Aug_15_2024 (Token.sol#368) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Sep_15_2024 (Token.sol#369) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Oct_15_2024 (Token.sol#370) is not in UPPER_CASE_WITH_UNDERSCORES
Constant BipgoToken._Nov_15_2024 (Token.sol#371) is not in UPPER_CASE_WITH_UNDERSCORES
Variable BipgoToken._categories (Token.sol#373) is not in mixedCase
Variable BipgoToken._dates (Token.sol#374) is not in mixedCase
Variable BipgoToken._categoriesAmountCap (Token.sol#375) is not in mixedCase
Variable BipgoToken._categoriesAddress (Token.sol#376) is not in mixedCase
Variable BipgoToken._coinDistribution (Token.sol#377) is not in mixedCase
Variable BipgoToken._coinDistributionStatus (Token.sol#378) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
```



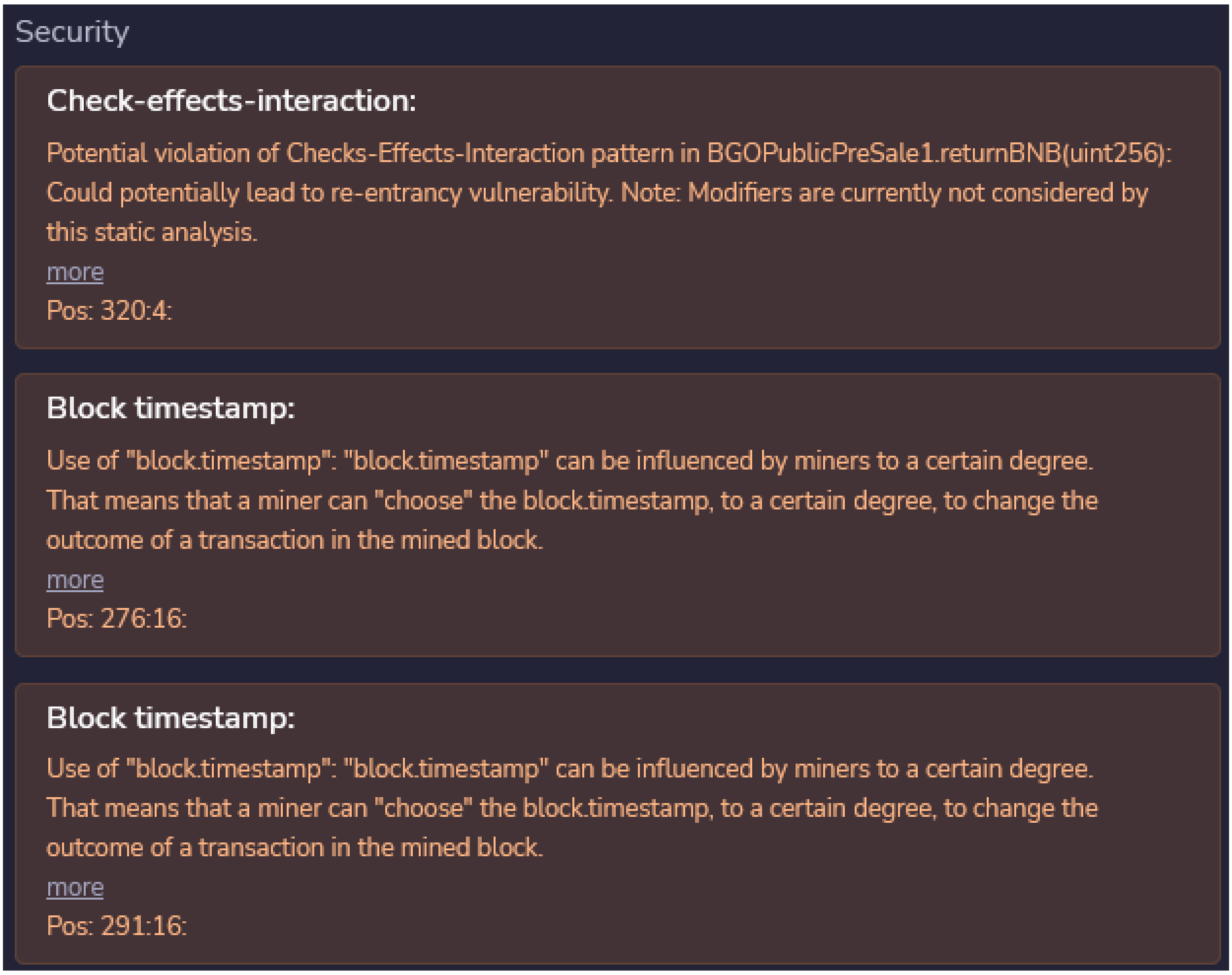
```
INFO:Detectors:
Variable BipgoToken._Apr_15_2022 (Token.sol#340) is too similar to BipgoToken._Apr_15_2024 (Token.sol#364)
Variable BipgoToken._Apr_15_2022 (Token.sol#340) is too similar to BipgoToken._Mar_15_2022 (Token.sol#339)
Variable BipgoToken._Apr_15_2022 (Token.sol#340) is too similar to BipgoToken._Apr_15_2023 (Token.sol#352)
Variable BipgoToken._Apr_15_2023 (Token.sol#352) is too similar to BipgoToken._Apr_15_2024 (Token.sol#364)
Variable BipgoToken._Aug_15_2022 (Token.sol#344) is too similar to BipgoToken._Aug_15_2023 (Token.sol#356)
Variable BipgoToken._Aug_15_2022 (Token.sol#344) is too similar to BipgoToken._Aug_15_2024 (Token.sol#368)
Variable BipgoToken._Aug_15_2023 (Token.sol#356) is too similar to BipgoToken._Aug_15_2024 (Token.sol#368)
Variable BipgoToken._Dec_15_2021 (Token.sol#336) is too similar to BipgoToken._Dec_15_2022 (Token.sol#348)
Variable BipgoToken._Dec_15_2022 (Token.sol#348) is too similar to BipgoToken._Dec_15_2023 (Token.sol#360)
Variable BipgoToken._Dec_15_2021 (Token.sol#336) is too similar to BipgoToken._Dec_15_2023 (Token.sol#360)
Variable BipgoToken._Feb_15_2022 (Token.sol#338) is too similar to BipgoToken._Feb_15_2023 (Token.sol#350)
Variable BipgoToken._Feb_15_2023 (Token.sol#350) is too similar to BipgoToken._Feb_15_2024 (Token.sol#362)
Variable BipgoToken._Feb_15_2022 (Token.sol#338) is too similar to BipgoToken._Feb_15_2024 (Token.sol#362)
Variable BipgoToken._Jan_15_2022 (Token.sol#337) is too similar to BipgoToken._Jan_15_2023 (Token.sol#349)
Variable BipgoToken._Jan_15_2023 (Token.sol#349) is too similar to BipgoToken._Jan_15_2024 (Token.sol#361)
Variable BipgoToken._Jan_15_2023 (Token.sol#349) is too similar to BipgoToken._Jun_15_2023 (Token.sol#354)
Variable BipgoToken._Jan_15_2022 (Token.sol#337) is too similar to BipgoToken._Jan_15_2024 (Token.sol#361)
Variable BipgoToken._Jan_15_2024 (Token.sol#361) is too similar to BipgoToken._Jun_15_2024 (Token.sol#366)
Variable BipgoToken._Jul_15_2022 (Token.sol#343) is too similar to BipgoToken._Jul_15_2023 (Token.sol#355)
Variable BipgoToken._Jul_15_2022 (Token.sol#343) is too similar to BipgoToken._Jul_15_2024 (Token.sol#367)
Variable BipgoToken._Jul_15_2023 (Token.sol#355) is too similar to BipgoToken._Jul_15_2024 (Token.sol#367)
Variable BipgoToken._Jan_15_2022 (Token.sol#337) is too similar to BipgoToken._Jun_15_2022 (Token.sol#342)
Variable BipgoToken._Jul_15_2022 (Token.sol#343) is too similar to BipgoToken._Jun_15_2022 (Token.sol#342)
Variable BipgoToken._Jun_15_2022 (Token.sol#342) is too similar to BipgoToken._Jun_15_2023 (Token.sol#354)
Variable BipgoToken._Jun_15_2022 (Token.sol#342) is too similar to BipgoToken._Jun_15_2024 (Token.sol#366)
Variable BipgoToken._Jul_15_2023 (Token.sol#355) is too similar to BipgoToken._Jun_15_2023 (Token.sol#354)
Variable BipgoToken._Jun_15_2023 (Token.sol#354) is too similar to BipgoToken._Jun_15_2024 (Token.sol#366)
Variable BipgoToken._Jul_15_2024 (Token.sol#367) is too similar to BipgoToken._Jun_15_2024 (Token.sol#366)
Variable BipgoToken._Apr_15_2023 (Token.sol#352) is too similar to BipgoToken._Mar_15_2023 (Token.sol#351)
Variable BipgoToken._Mar_15_2022 (Token.sol#339) is too similar to BipgoToken._Mar_15_2023 (Token.sol#351)
Variable BipgoToken._Mar_15_2023 (Token.sol#351) is too similar to BipgoToken._Mar_15_2024 (Token.sol#363)
Variable BipgoToken._Mar_15_2023 (Token.sol#351) is too similar to BipgoToken._May_15_2023 (Token.sol#353)
Variable BipgoToken._Apr_15_2024 (Token.sol#364) is too similar to BipgoToken._Mar_15_2024 (Token.sol#363)
Variable BipgoToken._Mar_15_2022 (Token.sol#339) is too similar to BipgoToken._Mar_15_2024 (Token.sol#363)
Variable BipgoToken._Mar_15_2024 (Token.sol#363) is too similar to BipgoToken._May_15_2024 (Token.sol#365)
Variable BipgoToken._Mar_15_2022 (Token.sol#339) is too similar to BipgoToken._May_15_2022 (Token.sol#341)
Variable BipgoToken._May_15_2022 (Token.sol#341) is too similar to BipgoToken._May_15_2023 (Token.sol#353)
```



```
Variable BipgoToken._Apr_15_2024 (Token.sol#364) is too similar to BipgoToken._Mar_15_2024 (Token.sol#363)
Variable BipgoToken._Mar_15_2022 (Token.sol#339) is too similar to BipgoToken._Mar_15_2024 (Token.sol#363)
Variable BipgoToken._Mar_15_2024 (Token.sol#363) is too similar to BipgoToken._May_15_2024 (Token.sol#365)
Variable BipgoToken._Mar_15_2022 (Token.sol#339) is too similar to BipgoToken._May_15_2022 (Token.sol#341)
Variable BipgoToken._May_15_2022 (Token.sol#341) is too similar to BipgoToken._May_15_2023 (Token.sol#353)
Variable BipgoToken._May_15_2022 (Token.sol#341) is too similar to BipgoToken._May_15_2024 (Token.sol#365)
Variable BipgoToken._May_15_2023 (Token.sol#353) is too similar to BipgoToken._May_15_2024 (Token.sol#365)
Variable BipgoToken._Nov_15_2021 (Token.sol#335) is too similar to BipgoToken._Nov_15_2022 (Token.sol#347)
Variable BipgoToken._Nov_15_2022 (Token.sol#347) is too similar to BipgoToken._Nov_15_2023 (Token.sol#359)
Variable BipgoToken._Nov_15_2021 (Token.sol#335) is too similar to BipgoToken._Nov_15_2023 (Token.sol#359)
Variable BipgoToken._Nov_15_2021 (Token.sol#335) is too similar to BipgoToken._Nov_15_2024 (Token.sol#371)
Variable BipgoToken._Nov_15_2022 (Token.sol#347) is too similar to BipgoToken._Nov_15_2024 (Token.sol#371)
Variable BipgoToken._Nov_15_2023 (Token.sol#359) is too similar to BipgoToken._Nov_15_2024 (Token.sol#371)
Variable BipgoToken._Oct_15_2022 (Token.sol#346) is too similar to BipgoToken._Oct_15_2023 (Token.sol#358)
Variable BipgoToken._Oct_15_2022 (Token.sol#346) is too similar to BipgoToken._Oct_15_2024 (Token.sol#370)
Variable BipgoToken._Oct_15_2023 (Token.sol#358) is too similar to BipgoToken._Oct_15_2024 (Token.sol#370)
Variable BipgoToken._Sep_15_2022 (Token.sol#345) is too similar to BipgoToken._Sep_15_2023 (Token.sol#357)
Variable BipgoToken._Sep_15_2022 (Token.sol#345) is too similar to BipgoToken._Sep_15_2024 (Token.sol#369)
Variable BipgoToken._Sep_15_2023 (Token.sol#357) is too similar to BipgoToken._Sep_15_2024 (Token.sol#369)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-are-too-similar
INFO:Detectors:
BipgoToken.slitherConstructorConstantVariables() (Token.sol#308-856) uses literals with too many digits:
        - _Sep_15_2022 = 1663200000 (Token.sol#345)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits
INFO:Detectors:
BipgoToken._totalSupply (Token.sol#314) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant
INFO:Detectors:
totalSupply() should be declared external:
        - BipgoToken.totalSupply() (Token.sol#700-702)
increaseAllowance(address,uint256) should be declared external:
        - BipgoToken.increaseAllowance(address,uint256) (Token.sol#740-743)
decreaseAllowance(address,uint256) should be declared external:
        - BipgoToken.decreaseAllowance(address,uint256) (Token.sol#759-762)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-function-that-could-be-declared-external
INFO:Slither:Token.sol analyzed (3 contracts with 75 detectors), 130 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and Github integration
```

Solidity static analysis

PublicPresale.sol



Security

Check-effects-interaction:

Potential violation of Checks-Effects-Interaction pattern in BGOPublicPreSale1.returnBNB(uint256): Could potentially lead to re-entrancy vulnerability. Note: Modifiers are currently not considered by this static analysis.

more

Pos: 320:4:

Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree. That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.

more

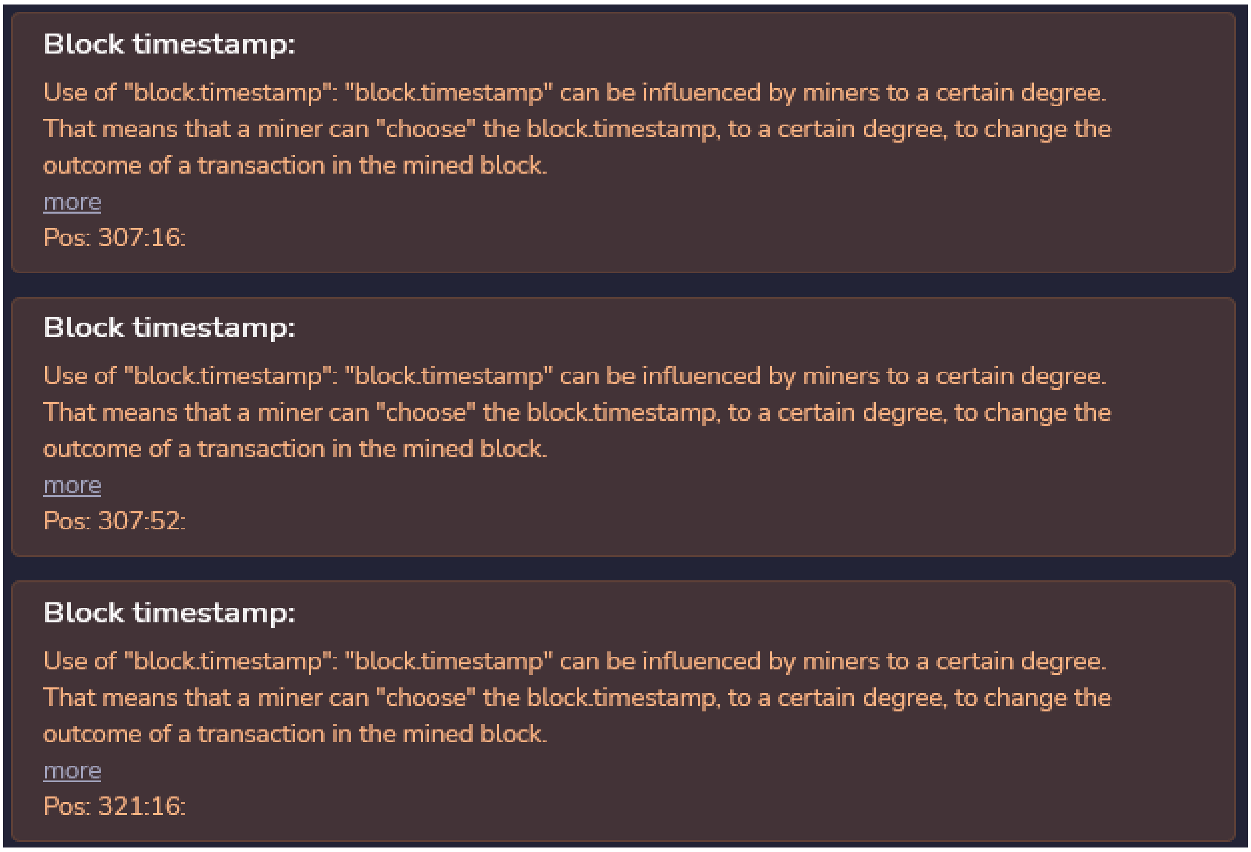
Pos: 276:16:

Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree. That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.

more

Pos: 291:16:



Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree. That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.

more

Pos: 307:16:

Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree. That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.

more

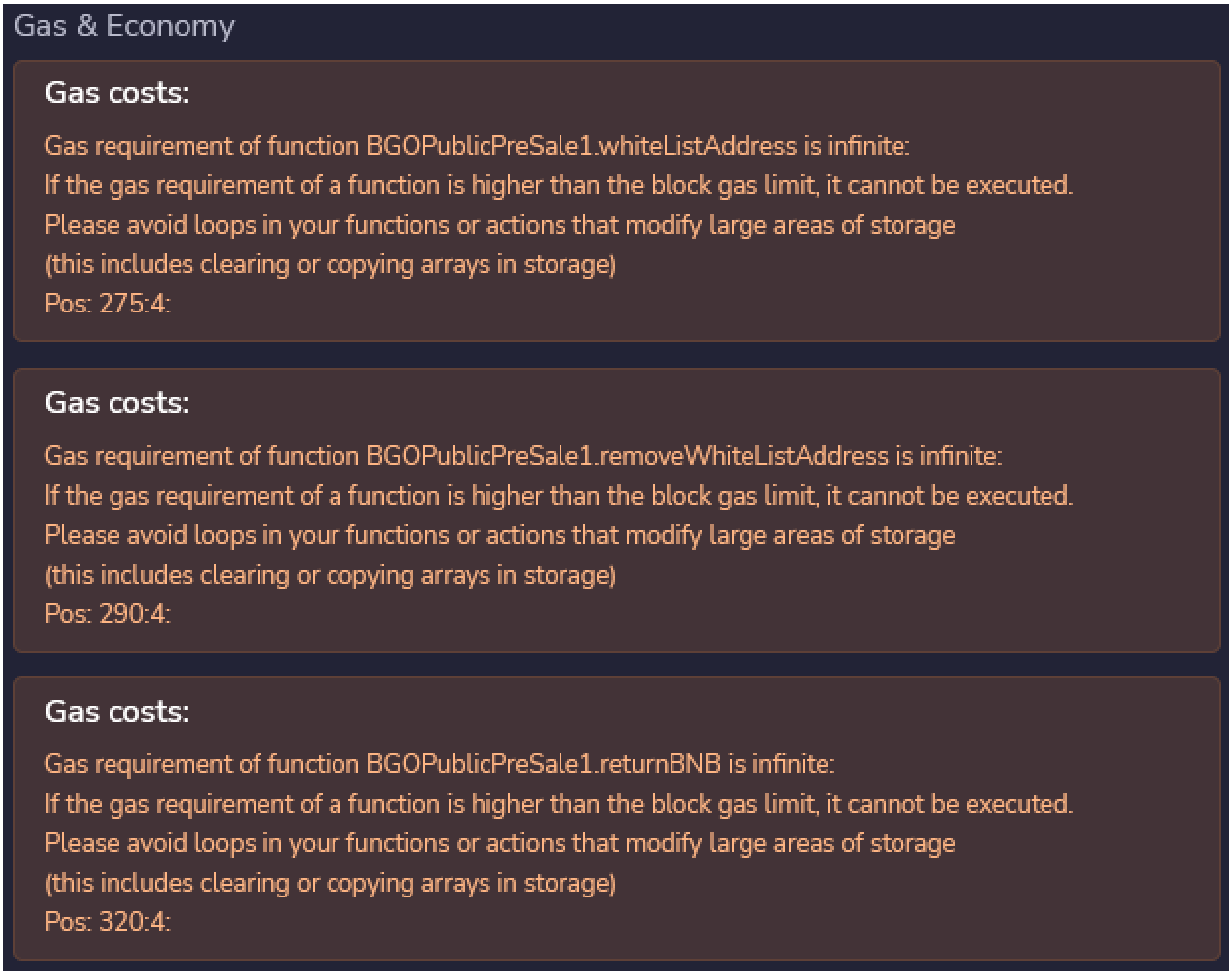
Pos: 307:52:

Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree. That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.

more

Pos: 321:16:



Gas & Economy

Gas costs:

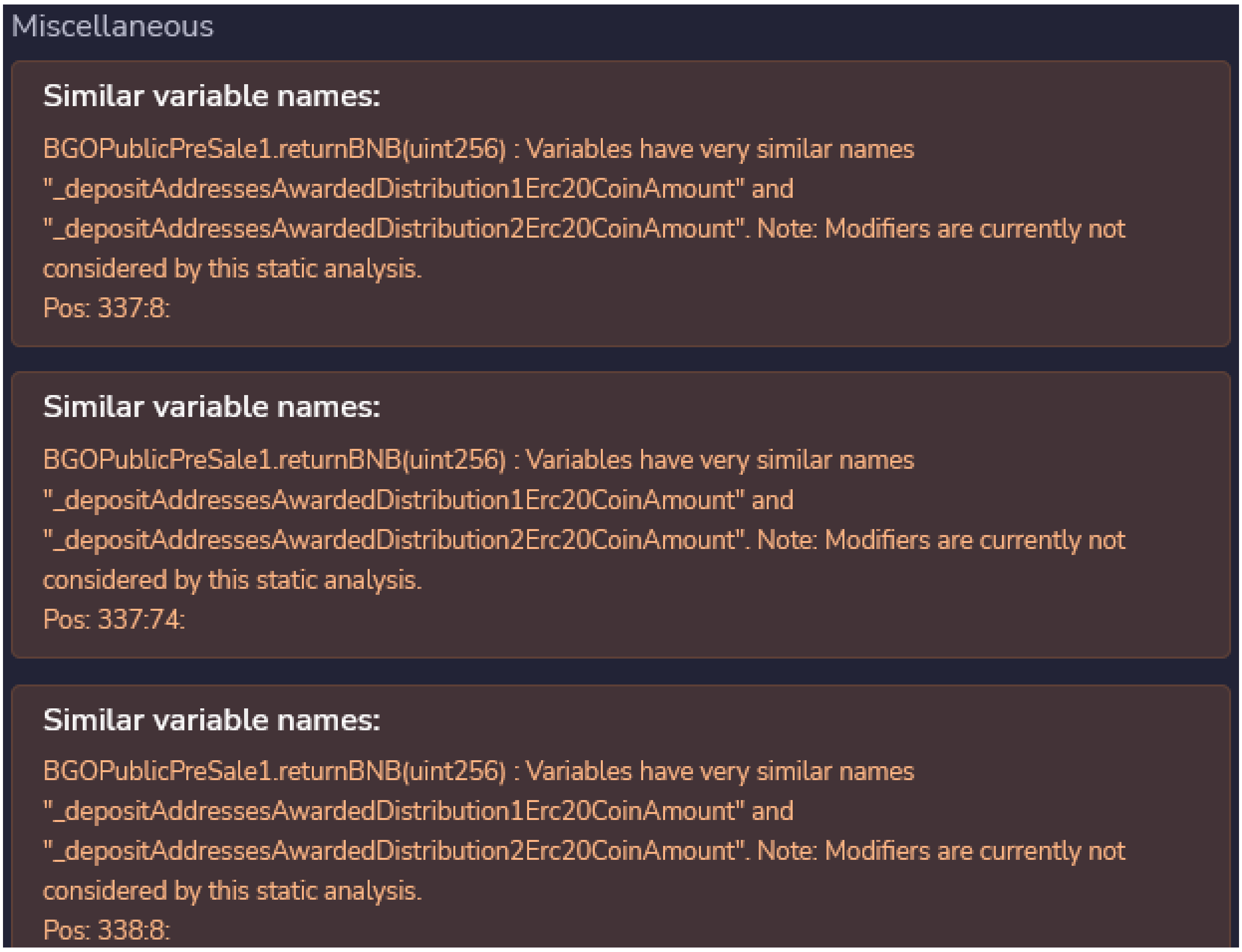
Gas requirement of function BGOPublicPreSale1.whiteListAddress is infinite:
If the gas requirement of a function is higher than the block gas limit, it cannot be executed.
Please avoid loops in your functions or actions that modify large areas of storage
(this includes clearing or copying arrays in storage)
Pos: 275:4:

Gas costs:

Gas requirement of function BGOPublicPreSale1.removeWhiteListAddress is infinite:
If the gas requirement of a function is higher than the block gas limit, it cannot be executed.
Please avoid loops in your functions or actions that modify large areas of storage
(this includes clearing or copying arrays in storage)
Pos: 290:4:

Gas costs:

Gas requirement of function BGOPublicPreSale1.returnBNB is infinite:
If the gas requirement of a function is higher than the block gas limit, it cannot be executed.
Please avoid loops in your functions or actions that modify large areas of storage
(this includes clearing or copying arrays in storage)
Pos: 320:4:



Miscellaneous

Similar variable names:

BGOPublicPreSale1.returnBNB(uint256) : Variables have very similar names "_depositAddressesAwardedDistribution1Erc20CoinAmount" and "_depositAddressesAwardedDistribution2Erc20CoinAmount". Note: Modifiers are currently not considered by this static analysis.
Pos: 337:8:

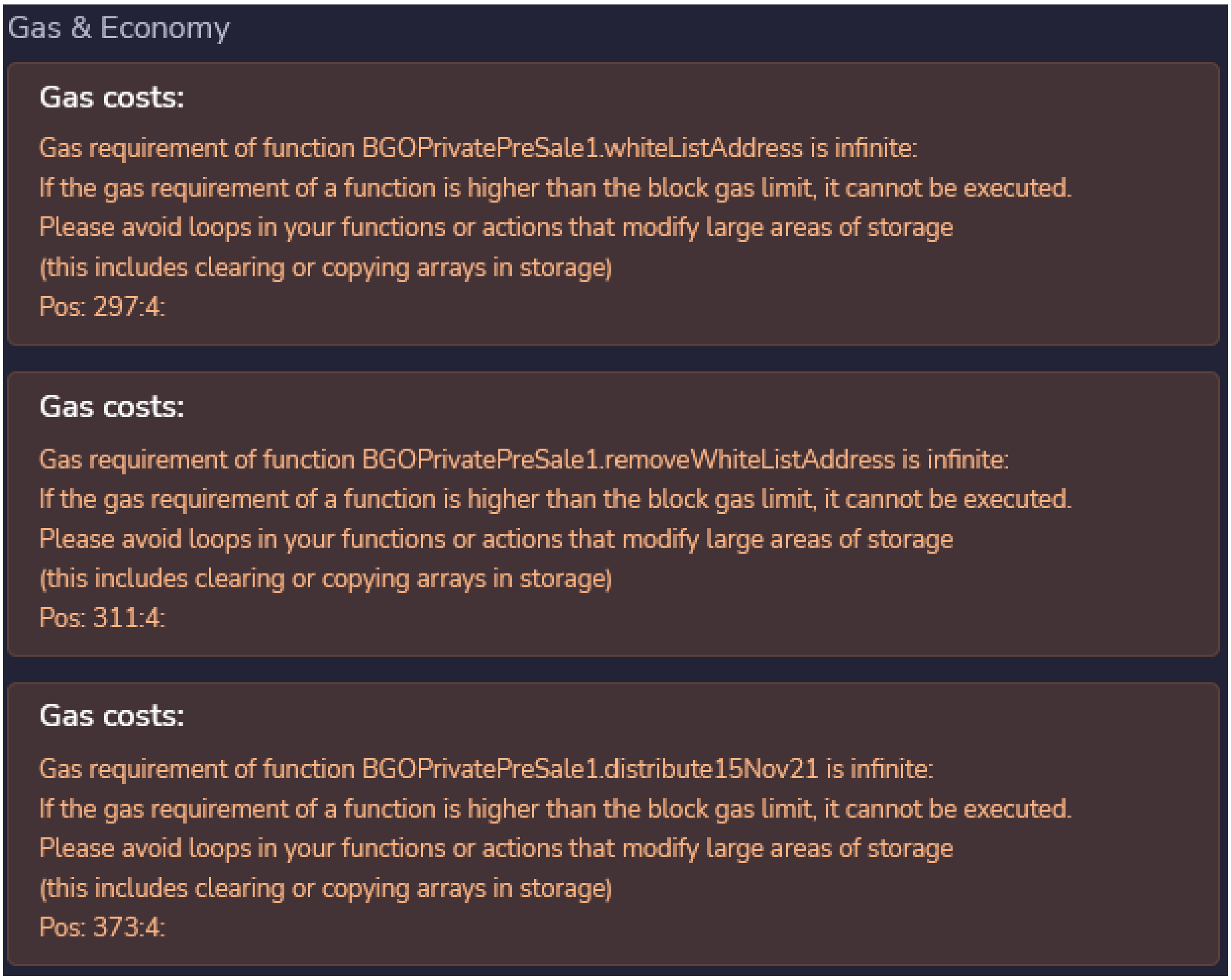
Similar variable names:

BGOPublicPreSale1.returnBNB(uint256) : Variables have very similar names "_depositAddressesAwardedDistribution1Erc20CoinAmount" and "_depositAddressesAwardedDistribution2Erc20CoinAmount". Note: Modifiers are currently not considered by this static analysis.
Pos: 337:74:

Similar variable names:

BGOPublicPreSale1.returnBNB(uint256) : Variables have very similar names "_depositAddressesAwardedDistribution1Erc20CoinAmount" and "_depositAddressesAwardedDistribution2Erc20CoinAmount". Note: Modifiers are currently not considered by this static analysis.
Pos: 338:8:

PrivatePreSale.sol



Gas & Economy

Gas costs:

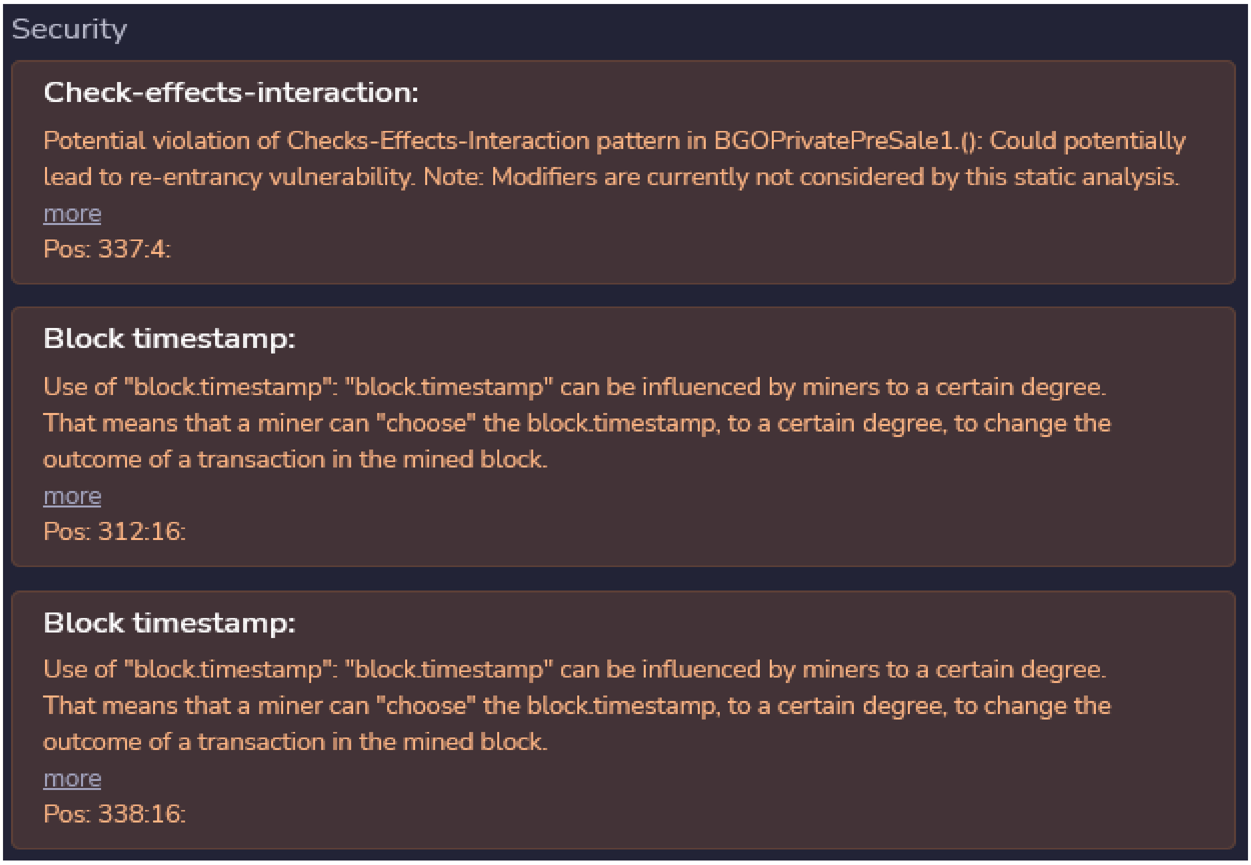
Gas requirement of function BGOPrivatePreSale1.whiteListAddress is infinite:
If the gas requirement of a function is higher than the block gas limit, it cannot be executed.
Please avoid loops in your functions or actions that modify large areas of storage
(this includes clearing or copying arrays in storage)
Pos: 297:4:

Gas costs:

Gas requirement of function BGOPrivatePreSale1.removeWhiteListAddress is infinite:
If the gas requirement of a function is higher than the block gas limit, it cannot be executed.
Please avoid loops in your functions or actions that modify large areas of storage
(this includes clearing or copying arrays in storage)
Pos: 311:4:

Gas costs:

Gas requirement of function BGOPrivatePreSale1.distribute15Nov21 is infinite:
If the gas requirement of a function is higher than the block gas limit, it cannot be executed.
Please avoid loops in your functions or actions that modify large areas of storage
(this includes clearing or copying arrays in storage)
Pos: 373:4:



Security

Check-effects-interaction:

Potential violation of Checks-Effects-Interaction pattern in BGOPrivatePreSale1.(): Could potentially lead to re-entrancy vulnerability. Note: Modifiers are currently not considered by this static analysis.
more
Pos: 337:4:

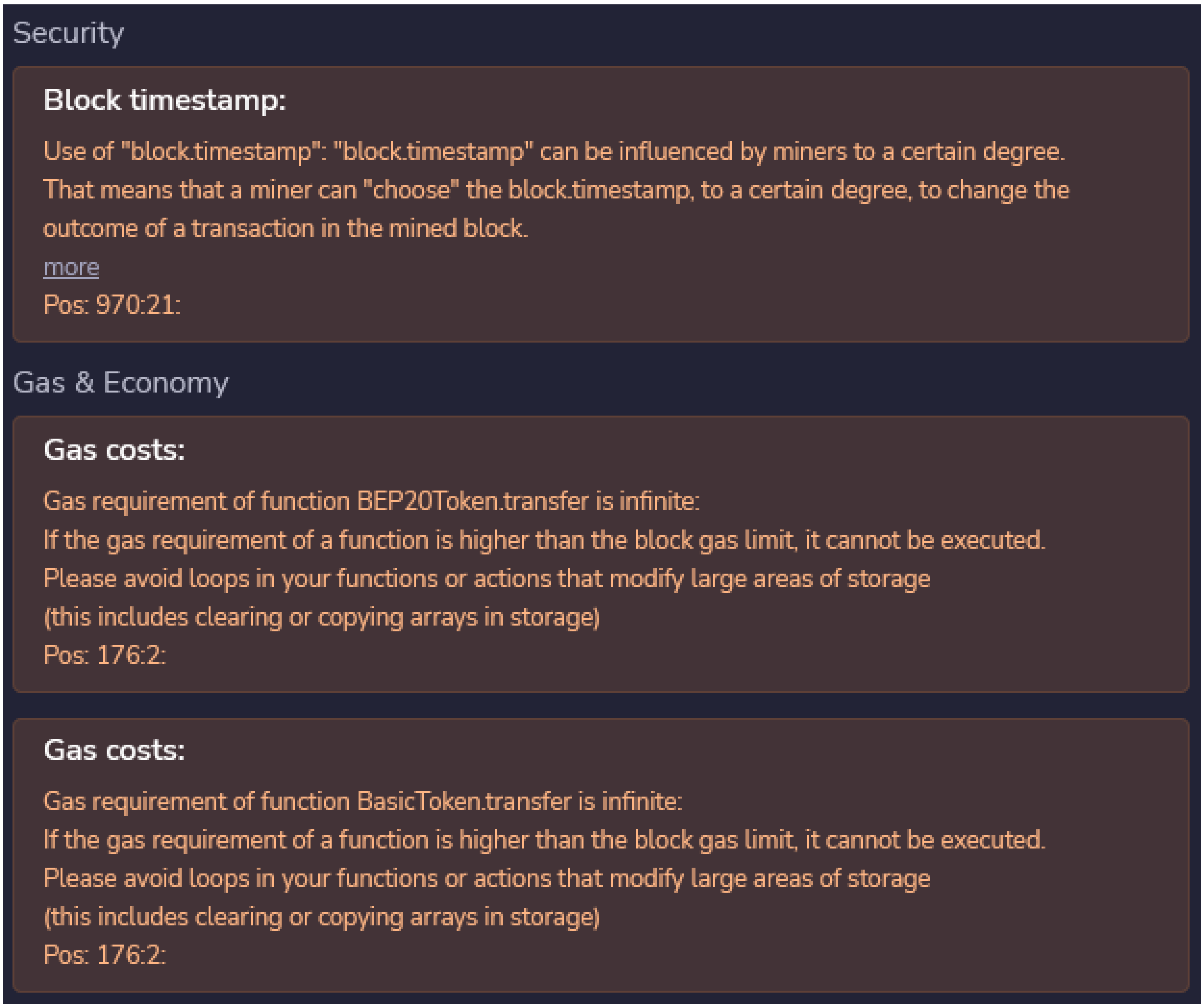
Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree.
That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.
more
Pos: 312:16:

Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree.
That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.
more
Pos: 338:16:

Token.sol



Security

Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree. That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.

Pos: 970:21:

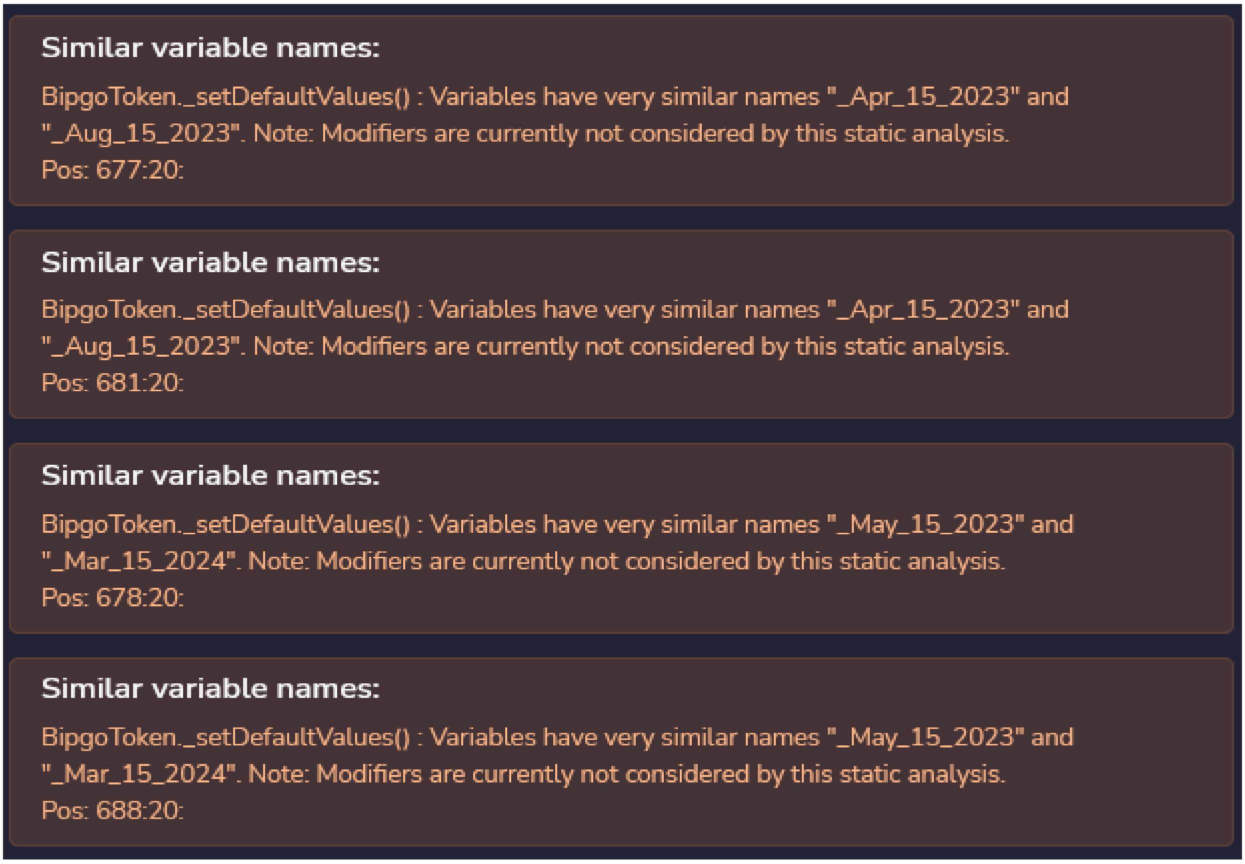
Gas & Economy

Gas costs:

Gas requirement of function BEP20Token.transfer is infinite:
If the gas requirement of a function is higher than the block gas limit, it cannot be executed.
Please avoid loops in your functions or actions that modify large areas of storage
(this includes clearing or copying arrays in storage)
Pos: 176:2:

Gas costs:

Gas requirement of function BasicToken.transfer is infinite:
If the gas requirement of a function is higher than the block gas limit, it cannot be executed.
Please avoid loops in your functions or actions that modify large areas of storage
(this includes clearing or copying arrays in storage)
Pos: 176:2:



Similar variable names:

BipgoToken._setDefaultValues() : Variables have very similar names "_Apr_15_2023" and "_Aug_15_2023". Note: Modifiers are currently not considered by this static analysis.
Pos: 677:20:

Similar variable names:

BipgoToken._setDefaultValues() : Variables have very similar names "_Apr_15_2023" and "_Aug_15_2023". Note: Modifiers are currently not considered by this static analysis.
Pos: 681:20:

Similar variable names:

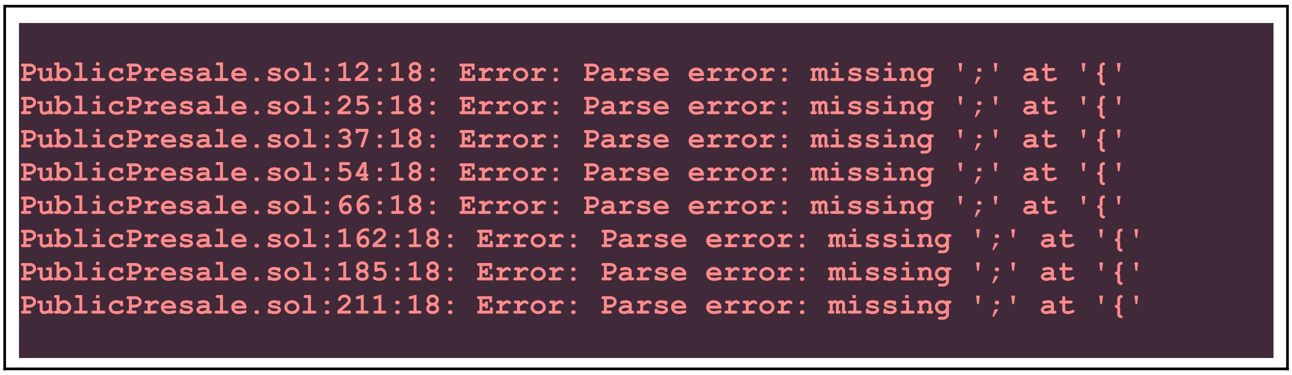
BipgoToken._setDefaultValues() : Variables have very similar names "_May_15_2023" and "_Mar_15_2024". Note: Modifiers are currently not considered by this static analysis.
Pos: 678:20:

Similar variable names:

BipgoToken._setDefaultValues() : Variables have very similar names "_May_15_2023" and "_Mar_15_2024". Note: Modifiers are currently not considered by this static analysis.
Pos: 688:20:

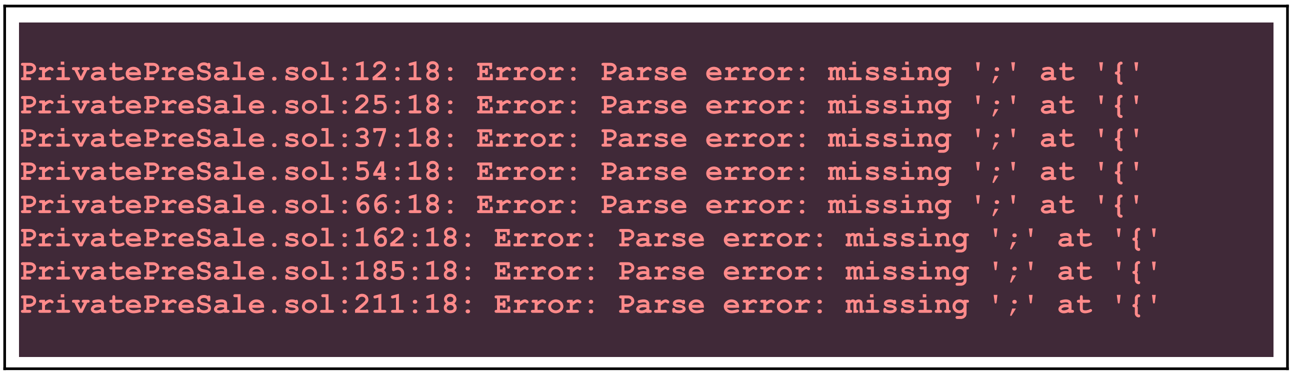
Solhint Linter

PublicPresale.sol



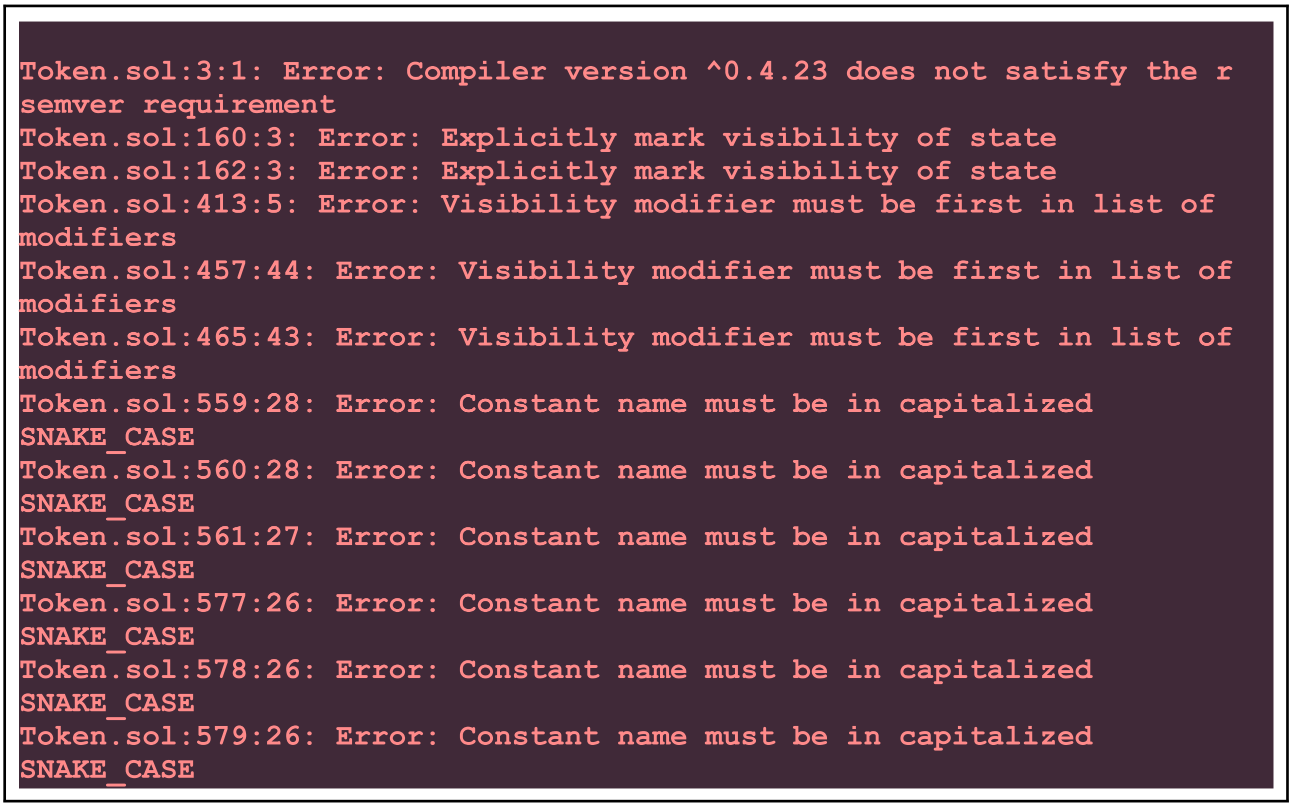
```
PublicPresale.sol:12:18: Error: Parse error: missing ';' at '{'
PublicPresale.sol:25:18: Error: Parse error: missing ';' at '{'
PublicPresale.sol:37:18: Error: Parse error: missing ';' at '{'
PublicPresale.sol:54:18: Error: Parse error: missing ';' at '{'
PublicPresale.sol:66:18: Error: Parse error: missing ';' at '{'
PublicPresale.sol:162:18: Error: Parse error: missing ';' at '{'
PublicPresale.sol:185:18: Error: Parse error: missing ';' at '{'
PublicPresale.sol:211:18: Error: Parse error: missing ';' at '{'
```

PrivatePreSale.sol

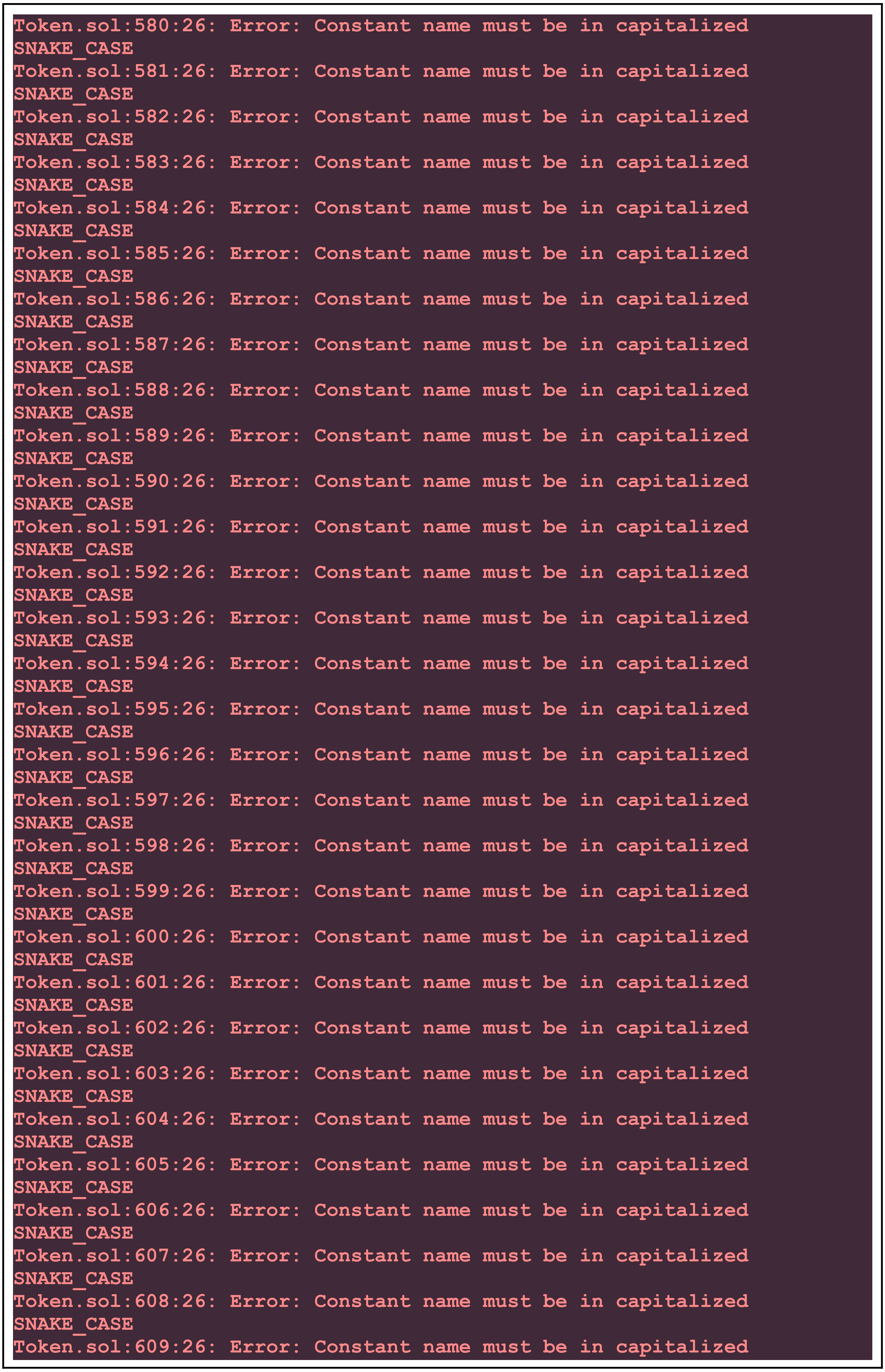


```
PrivatePreSale.sol:12:18: Error: Parse error: missing ';' at '{'
PrivatePreSale.sol:25:18: Error: Parse error: missing ';' at '{'
PrivatePreSale.sol:37:18: Error: Parse error: missing ';' at '{'
PrivatePreSale.sol:54:18: Error: Parse error: missing ';' at '{'
PrivatePreSale.sol:66:18: Error: Parse error: missing ';' at '{'
PrivatePreSale.sol:162:18: Error: Parse error: missing ';' at '{'
PrivatePreSale.sol:185:18: Error: Parse error: missing ';' at '{'
PrivatePreSale.sol:211:18: Error: Parse error: missing ';' at '{'
```

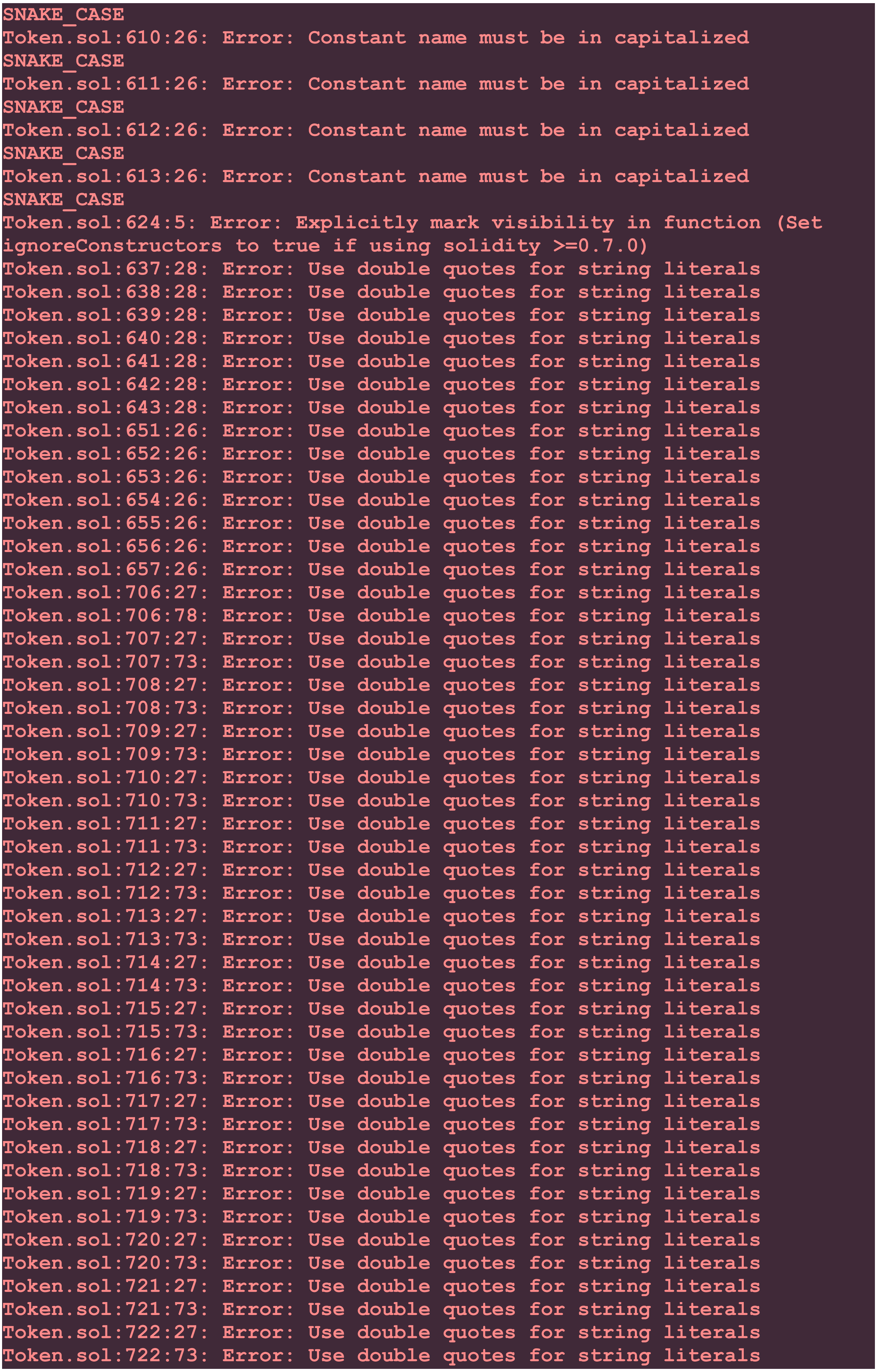
Token.sol



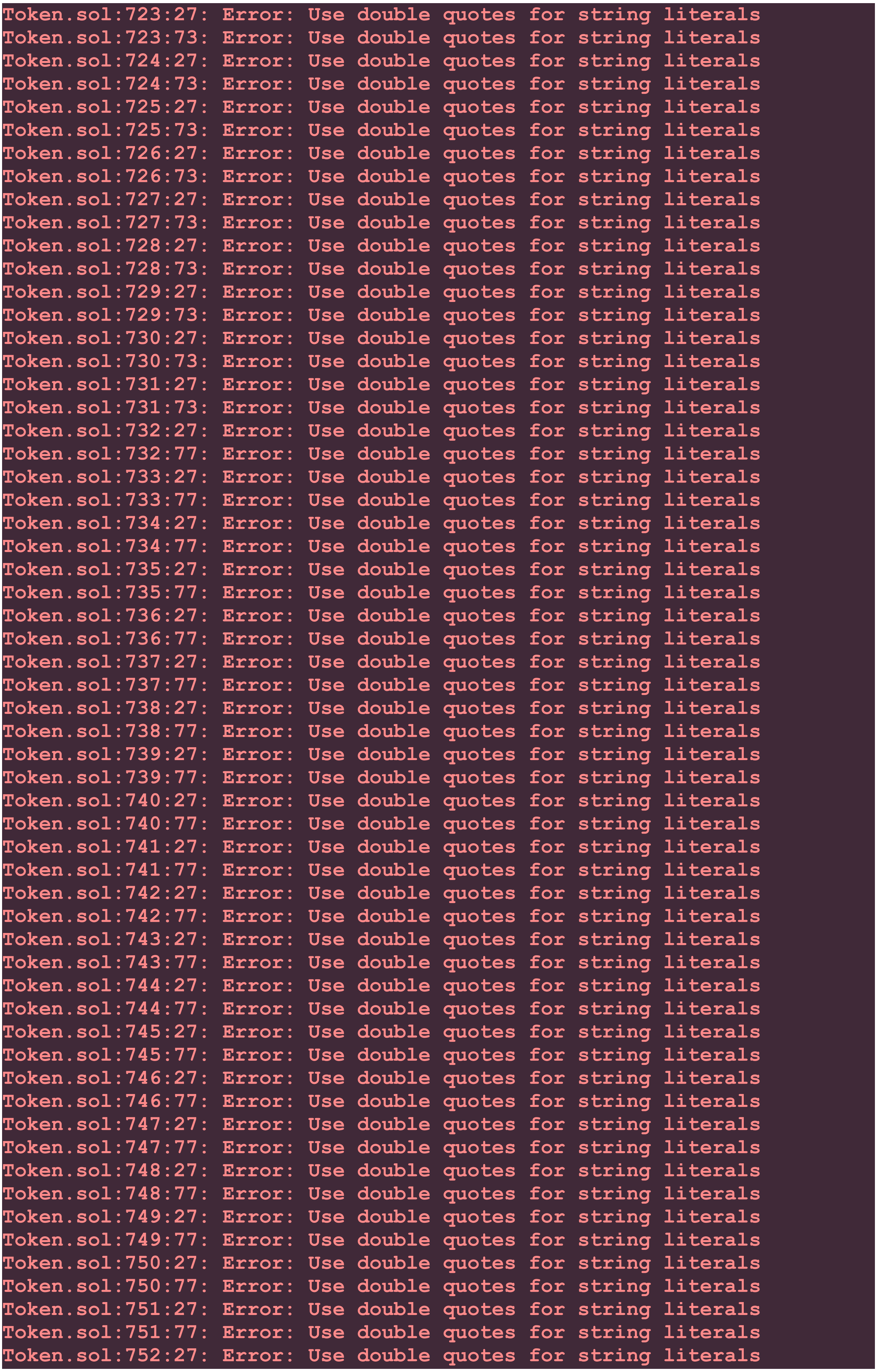
```
Token.sol:3:1: Error: Compiler version ^0.4.23 does not satisfy the r
semver requirement
Token.sol:160:3: Error: Explicitly mark visibility of state
Token.sol:162:3: Error: Explicitly mark visibility of state
Token.sol:413:5: Error: Visibility modifier must be first in list of
modifiers
Token.sol:457:44: Error: Visibility modifier must be first in list of
modifiers
Token.sol:465:43: Error: Visibility modifier must be first in list of
modifiers
Token.sol:559:28: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:560:28: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:561:27: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:577:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:578:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:579:26: Error: Constant name must be in capitalized
SNAKE_CASE
```



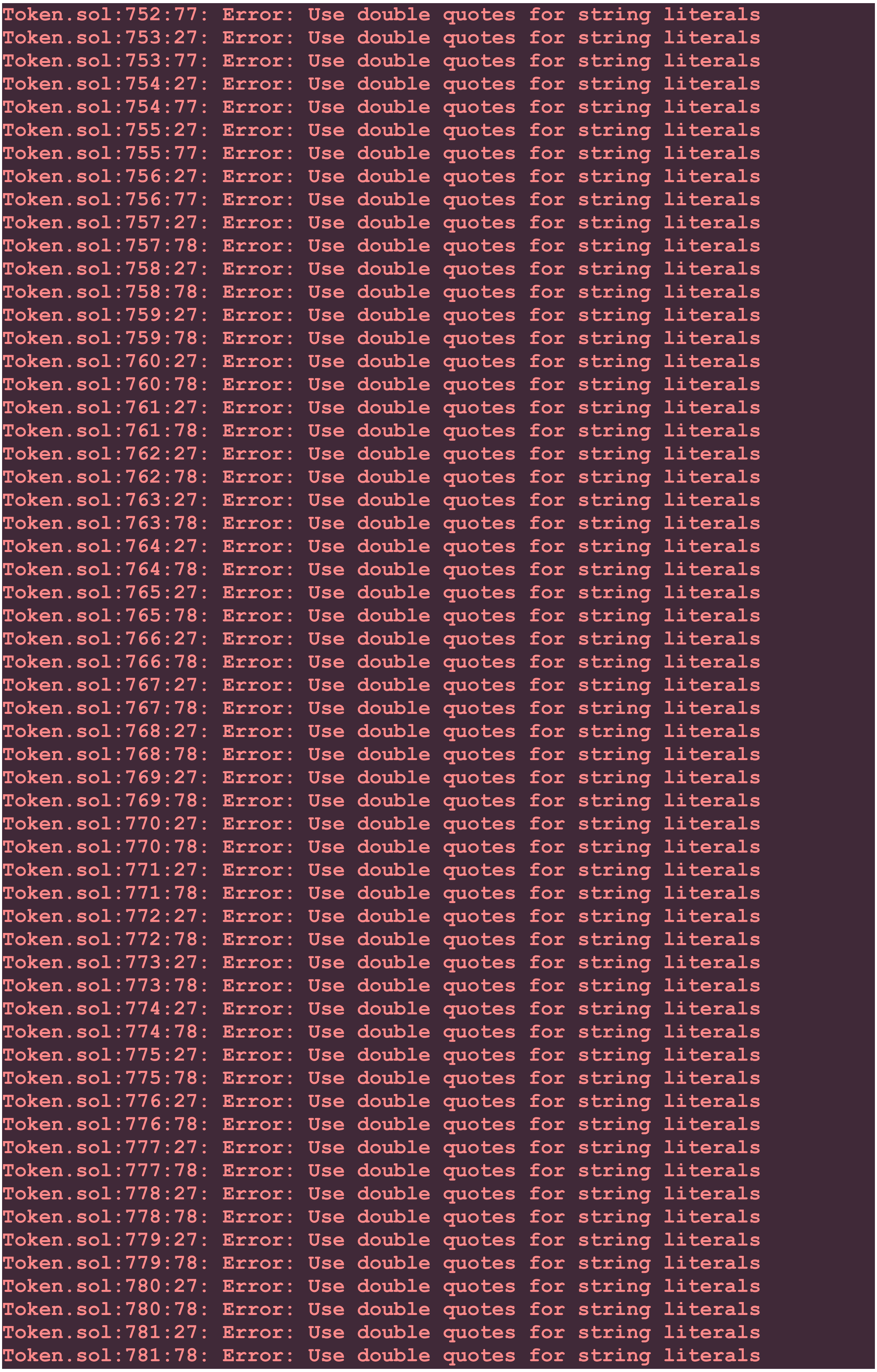
```
Token.sol:580:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:581:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:582:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:583:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:584:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:585:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:586:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:587:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:588:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:589:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:590:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:591:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:592:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:593:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:594:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:595:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:596:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:597:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:598:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:599:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:600:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:601:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:602:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:603:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:604:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:605:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:606:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:607:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:608:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:609:26: Error: Constant name must be in capitalized
```



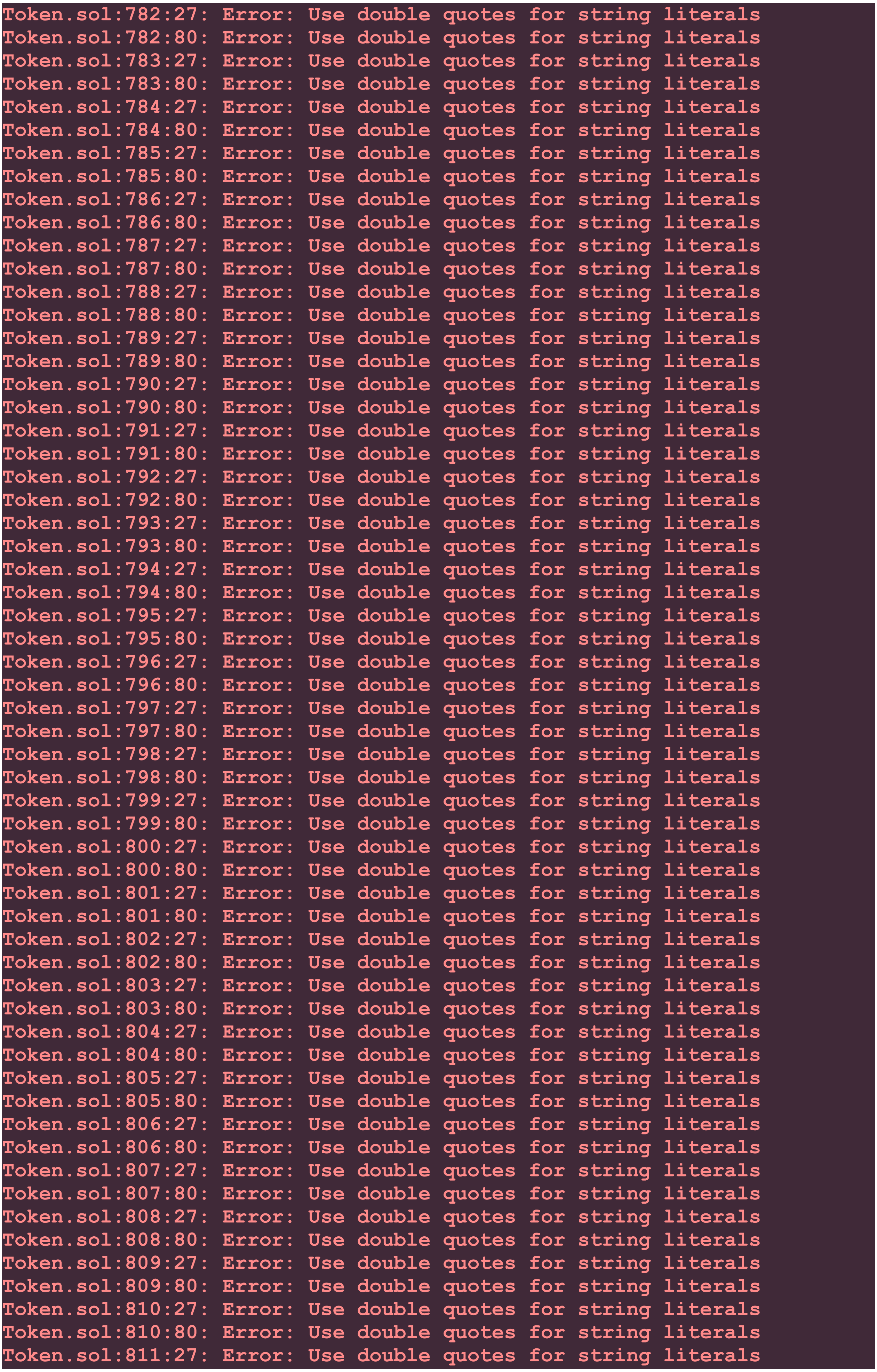
```
SNAKE_CASE
Token.sol:610:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:611:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:612:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:613:26: Error: Constant name must be in capitalized
SNAKE_CASE
Token.sol:624:5: Error: Explicitly mark visibility in function (Set
ignoreConstructors to true if using solidity >=0.7.0)
Token.sol:637:28: Error: Use double quotes for string literals
Token.sol:638:28: Error: Use double quotes for string literals
Token.sol:639:28: Error: Use double quotes for string literals
Token.sol:640:28: Error: Use double quotes for string literals
Token.sol:641:28: Error: Use double quotes for string literals
Token.sol:642:28: Error: Use double quotes for string literals
Token.sol:643:28: Error: Use double quotes for string literals
Token.sol:651:26: Error: Use double quotes for string literals
Token.sol:652:26: Error: Use double quotes for string literals
Token.sol:653:26: Error: Use double quotes for string literals
Token.sol:654:26: Error: Use double quotes for string literals
Token.sol:655:26: Error: Use double quotes for string literals
Token.sol:656:26: Error: Use double quotes for string literals
Token.sol:657:26: Error: Use double quotes for string literals
Token.sol:706:27: Error: Use double quotes for string literals
Token.sol:706:78: Error: Use double quotes for string literals
Token.sol:707:27: Error: Use double quotes for string literals
Token.sol:707:73: Error: Use double quotes for string literals
Token.sol:708:27: Error: Use double quotes for string literals
Token.sol:708:73: Error: Use double quotes for string literals
Token.sol:709:27: Error: Use double quotes for string literals
Token.sol:709:73: Error: Use double quotes for string literals
Token.sol:710:27: Error: Use double quotes for string literals
Token.sol:710:73: Error: Use double quotes for string literals
Token.sol:711:27: Error: Use double quotes for string literals
Token.sol:711:73: Error: Use double quotes for string literals
Token.sol:712:27: Error: Use double quotes for string literals
Token.sol:712:73: Error: Use double quotes for string literals
Token.sol:713:27: Error: Use double quotes for string literals
Token.sol:713:73: Error: Use double quotes for string literals
Token.sol:714:27: Error: Use double quotes for string literals
Token.sol:714:73: Error: Use double quotes for string literals
Token.sol:715:27: Error: Use double quotes for string literals
Token.sol:715:73: Error: Use double quotes for string literals
Token.sol:716:27: Error: Use double quotes for string literals
Token.sol:716:73: Error: Use double quotes for string literals
Token.sol:717:27: Error: Use double quotes for string literals
Token.sol:717:73: Error: Use double quotes for string literals
Token.sol:718:27: Error: Use double quotes for string literals
Token.sol:718:73: Error: Use double quotes for string literals
Token.sol:719:27: Error: Use double quotes for string literals
Token.sol:719:73: Error: Use double quotes for string literals
Token.sol:720:27: Error: Use double quotes for string literals
Token.sol:720:73: Error: Use double quotes for string literals
Token.sol:721:27: Error: Use double quotes for string literals
Token.sol:721:73: Error: Use double quotes for string literals
Token.sol:722:27: Error: Use double quotes for string literals
Token.sol:722:73: Error: Use double quotes for string literals
```



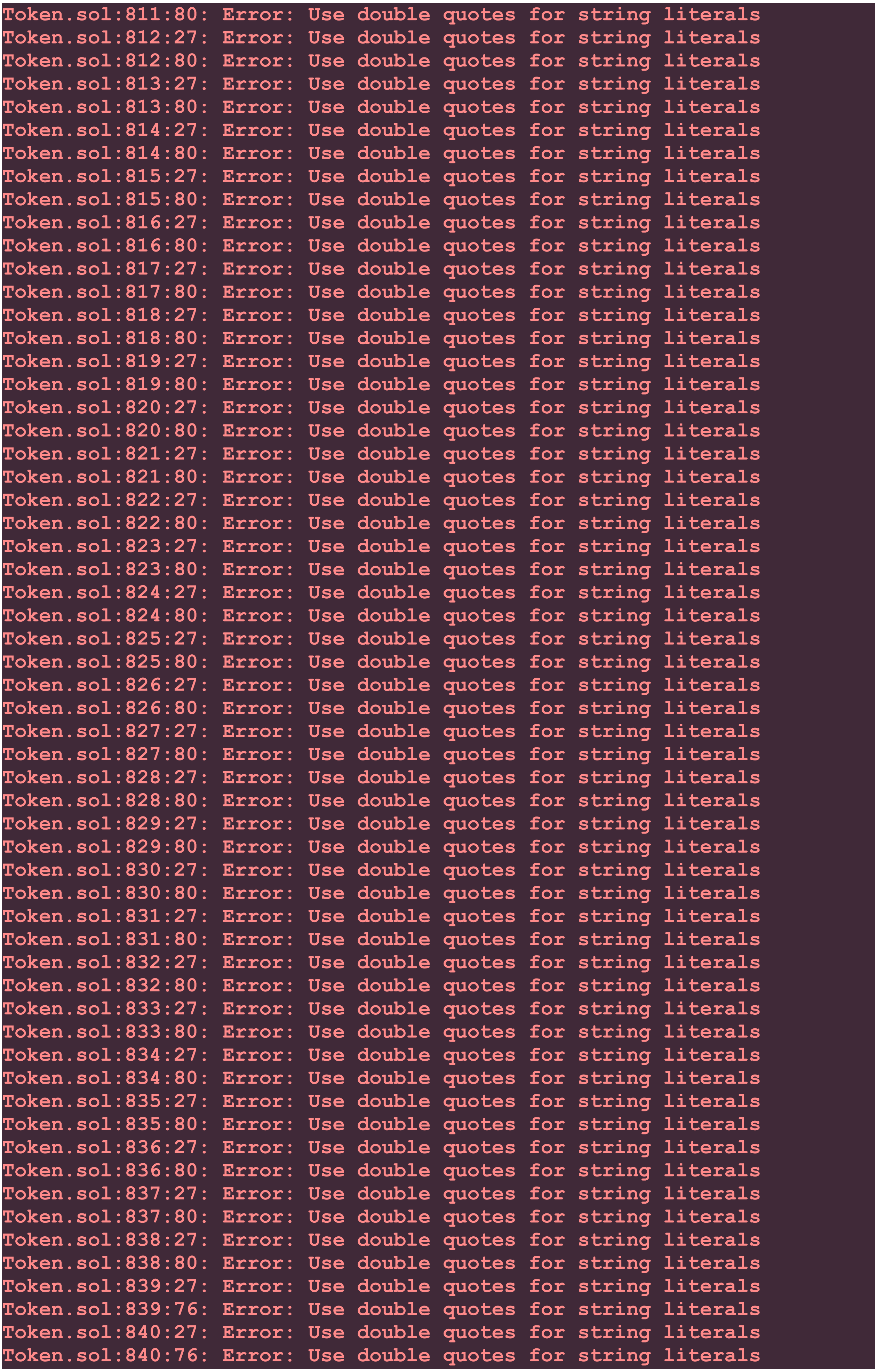
```
Token.sol:723:27: Error: Use double quotes for string literals
Token.sol:723:73: Error: Use double quotes for string literals
Token.sol:724:27: Error: Use double quotes for string literals
Token.sol:724:73: Error: Use double quotes for string literals
Token.sol:725:27: Error: Use double quotes for string literals
Token.sol:725:73: Error: Use double quotes for string literals
Token.sol:726:27: Error: Use double quotes for string literals
Token.sol:726:73: Error: Use double quotes for string literals
Token.sol:727:27: Error: Use double quotes for string literals
Token.sol:727:73: Error: Use double quotes for string literals
Token.sol:728:27: Error: Use double quotes for string literals
Token.sol:728:73: Error: Use double quotes for string literals
Token.sol:729:27: Error: Use double quotes for string literals
Token.sol:729:73: Error: Use double quotes for string literals
Token.sol:730:27: Error: Use double quotes for string literals
Token.sol:730:73: Error: Use double quotes for string literals
Token.sol:731:27: Error: Use double quotes for string literals
Token.sol:731:73: Error: Use double quotes for string literals
Token.sol:732:27: Error: Use double quotes for string literals
Token.sol:732:77: Error: Use double quotes for string literals
Token.sol:733:27: Error: Use double quotes for string literals
Token.sol:733:77: Error: Use double quotes for string literals
Token.sol:734:27: Error: Use double quotes for string literals
Token.sol:734:77: Error: Use double quotes for string literals
Token.sol:735:27: Error: Use double quotes for string literals
Token.sol:735:77: Error: Use double quotes for string literals
Token.sol:736:27: Error: Use double quotes for string literals
Token.sol:736:77: Error: Use double quotes for string literals
Token.sol:737:27: Error: Use double quotes for string literals
Token.sol:737:77: Error: Use double quotes for string literals
Token.sol:738:27: Error: Use double quotes for string literals
Token.sol:738:77: Error: Use double quotes for string literals
Token.sol:739:27: Error: Use double quotes for string literals
Token.sol:739:77: Error: Use double quotes for string literals
Token.sol:740:27: Error: Use double quotes for string literals
Token.sol:740:77: Error: Use double quotes for string literals
Token.sol:741:27: Error: Use double quotes for string literals
Token.sol:741:77: Error: Use double quotes for string literals
Token.sol:742:27: Error: Use double quotes for string literals
Token.sol:742:77: Error: Use double quotes for string literals
Token.sol:743:27: Error: Use double quotes for string literals
Token.sol:743:77: Error: Use double quotes for string literals
Token.sol:744:27: Error: Use double quotes for string literals
Token.sol:744:77: Error: Use double quotes for string literals
Token.sol:745:27: Error: Use double quotes for string literals
Token.sol:745:77: Error: Use double quotes for string literals
Token.sol:746:27: Error: Use double quotes for string literals
Token.sol:746:77: Error: Use double quotes for string literals
Token.sol:747:27: Error: Use double quotes for string literals
Token.sol:747:77: Error: Use double quotes for string literals
Token.sol:748:27: Error: Use double quotes for string literals
Token.sol:748:77: Error: Use double quotes for string literals
Token.sol:749:27: Error: Use double quotes for string literals
Token.sol:749:77: Error: Use double quotes for string literals
Token.sol:750:27: Error: Use double quotes for string literals
Token.sol:750:77: Error: Use double quotes for string literals
Token.sol:751:27: Error: Use double quotes for string literals
Token.sol:751:77: Error: Use double quotes for string literals
Token.sol:752:27: Error: Use double quotes for string literals
```



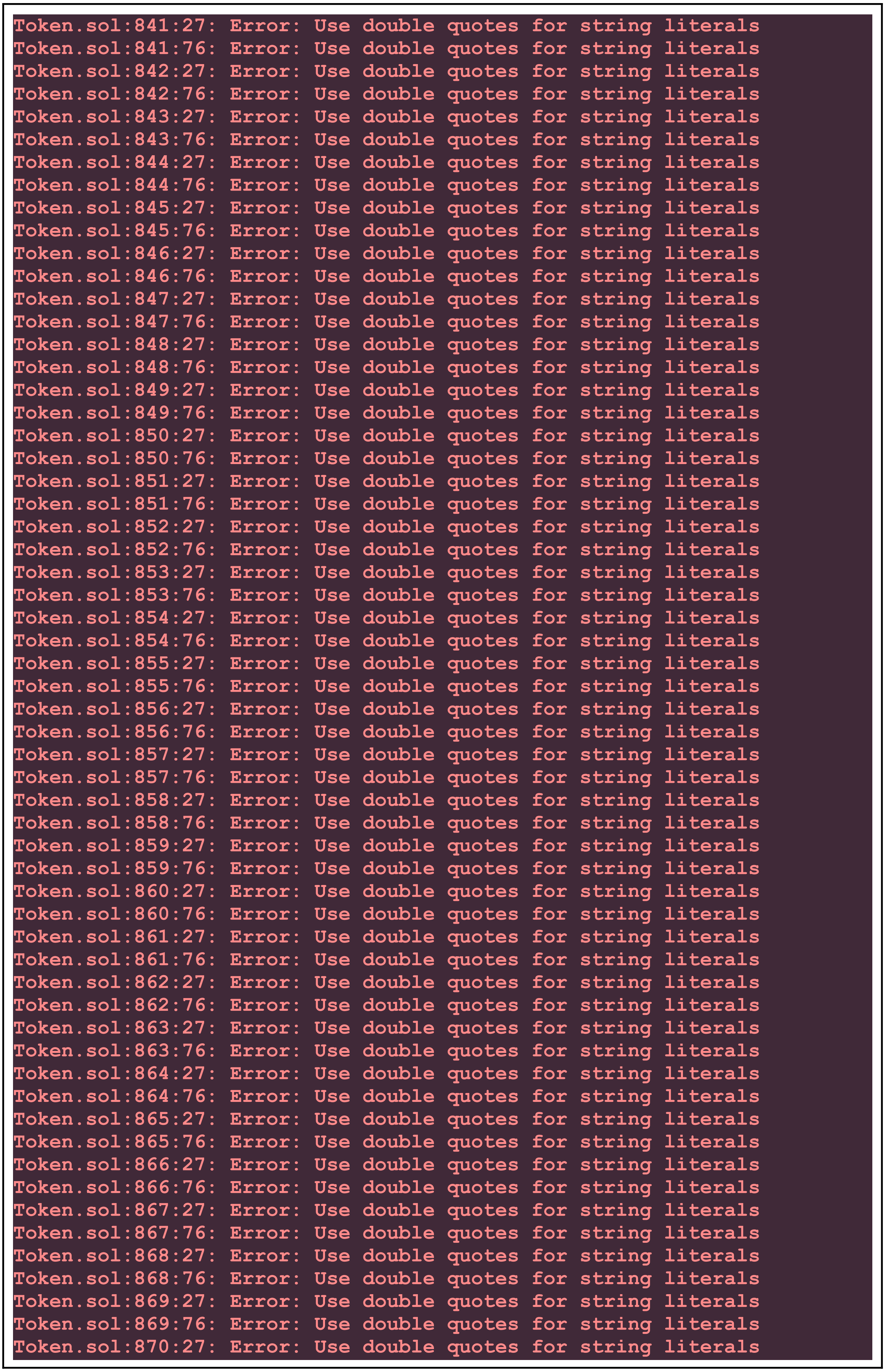
```
Token.sol:752:77: Error: Use double quotes for string literals
Token.sol:753:27: Error: Use double quotes for string literals
Token.sol:753:77: Error: Use double quotes for string literals
Token.sol:754:27: Error: Use double quotes for string literals
Token.sol:754:77: Error: Use double quotes for string literals
Token.sol:755:27: Error: Use double quotes for string literals
Token.sol:755:77: Error: Use double quotes for string literals
Token.sol:756:27: Error: Use double quotes for string literals
Token.sol:756:77: Error: Use double quotes for string literals
Token.sol:757:27: Error: Use double quotes for string literals
Token.sol:757:78: Error: Use double quotes for string literals
Token.sol:758:27: Error: Use double quotes for string literals
Token.sol:758:78: Error: Use double quotes for string literals
Token.sol:759:27: Error: Use double quotes for string literals
Token.sol:759:78: Error: Use double quotes for string literals
Token.sol:760:27: Error: Use double quotes for string literals
Token.sol:760:78: Error: Use double quotes for string literals
Token.sol:761:27: Error: Use double quotes for string literals
Token.sol:761:78: Error: Use double quotes for string literals
Token.sol:762:27: Error: Use double quotes for string literals
Token.sol:762:78: Error: Use double quotes for string literals
Token.sol:763:27: Error: Use double quotes for string literals
Token.sol:763:78: Error: Use double quotes for string literals
Token.sol:764:27: Error: Use double quotes for string literals
Token.sol:764:78: Error: Use double quotes for string literals
Token.sol:765:27: Error: Use double quotes for string literals
Token.sol:765:78: Error: Use double quotes for string literals
Token.sol:766:27: Error: Use double quotes for string literals
Token.sol:766:78: Error: Use double quotes for string literals
Token.sol:767:27: Error: Use double quotes for string literals
Token.sol:767:78: Error: Use double quotes for string literals
Token.sol:768:27: Error: Use double quotes for string literals
Token.sol:768:78: Error: Use double quotes for string literals
Token.sol:769:27: Error: Use double quotes for string literals
Token.sol:769:78: Error: Use double quotes for string literals
Token.sol:770:27: Error: Use double quotes for string literals
Token.sol:770:78: Error: Use double quotes for string literals
Token.sol:771:27: Error: Use double quotes for string literals
Token.sol:771:78: Error: Use double quotes for string literals
Token.sol:772:27: Error: Use double quotes for string literals
Token.sol:772:78: Error: Use double quotes for string literals
Token.sol:773:27: Error: Use double quotes for string literals
Token.sol:773:78: Error: Use double quotes for string literals
Token.sol:774:27: Error: Use double quotes for string literals
Token.sol:774:78: Error: Use double quotes for string literals
Token.sol:775:27: Error: Use double quotes for string literals
Token.sol:775:78: Error: Use double quotes for string literals
Token.sol:776:27: Error: Use double quotes for string literals
Token.sol:776:78: Error: Use double quotes for string literals
Token.sol:777:27: Error: Use double quotes for string literals
Token.sol:777:78: Error: Use double quotes for string literals
Token.sol:778:27: Error: Use double quotes for string literals
Token.sol:778:78: Error: Use double quotes for string literals
Token.sol:779:27: Error: Use double quotes for string literals
Token.sol:779:78: Error: Use double quotes for string literals
Token.sol:780:27: Error: Use double quotes for string literals
Token.sol:780:78: Error: Use double quotes for string literals
Token.sol:781:27: Error: Use double quotes for string literals
Token.sol:781:78: Error: Use double quotes for string literals
```



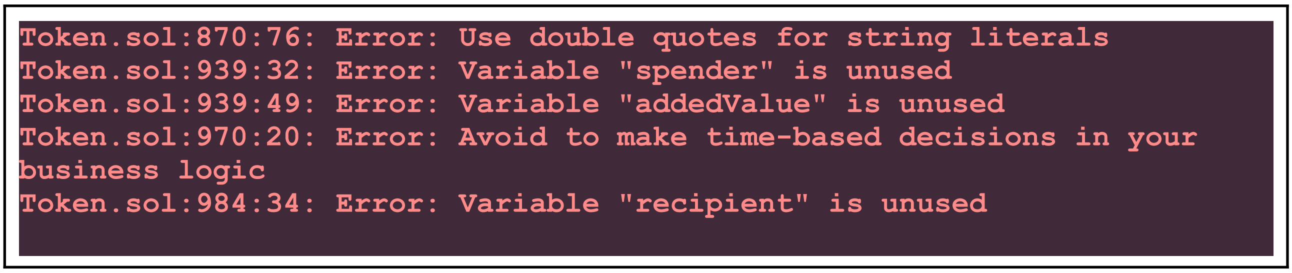
```
Token.sol:782:27: Error: Use double quotes for string literals
Token.sol:782:80: Error: Use double quotes for string literals
Token.sol:783:27: Error: Use double quotes for string literals
Token.sol:783:80: Error: Use double quotes for string literals
Token.sol:784:27: Error: Use double quotes for string literals
Token.sol:784:80: Error: Use double quotes for string literals
Token.sol:785:27: Error: Use double quotes for string literals
Token.sol:785:80: Error: Use double quotes for string literals
Token.sol:786:27: Error: Use double quotes for string literals
Token.sol:786:80: Error: Use double quotes for string literals
Token.sol:787:27: Error: Use double quotes for string literals
Token.sol:787:80: Error: Use double quotes for string literals
Token.sol:788:27: Error: Use double quotes for string literals
Token.sol:788:80: Error: Use double quotes for string literals
Token.sol:789:27: Error: Use double quotes for string literals
Token.sol:789:80: Error: Use double quotes for string literals
Token.sol:790:27: Error: Use double quotes for string literals
Token.sol:790:80: Error: Use double quotes for string literals
Token.sol:791:27: Error: Use double quotes for string literals
Token.sol:791:80: Error: Use double quotes for string literals
Token.sol:792:27: Error: Use double quotes for string literals
Token.sol:792:80: Error: Use double quotes for string literals
Token.sol:793:27: Error: Use double quotes for string literals
Token.sol:793:80: Error: Use double quotes for string literals
Token.sol:794:27: Error: Use double quotes for string literals
Token.sol:794:80: Error: Use double quotes for string literals
Token.sol:795:27: Error: Use double quotes for string literals
Token.sol:795:80: Error: Use double quotes for string literals
Token.sol:796:27: Error: Use double quotes for string literals
Token.sol:796:80: Error: Use double quotes for string literals
Token.sol:797:27: Error: Use double quotes for string literals
Token.sol:797:80: Error: Use double quotes for string literals
Token.sol:798:27: Error: Use double quotes for string literals
Token.sol:798:80: Error: Use double quotes for string literals
Token.sol:799:27: Error: Use double quotes for string literals
Token.sol:799:80: Error: Use double quotes for string literals
Token.sol:800:27: Error: Use double quotes for string literals
Token.sol:800:80: Error: Use double quotes for string literals
Token.sol:801:27: Error: Use double quotes for string literals
Token.sol:801:80: Error: Use double quotes for string literals
Token.sol:802:27: Error: Use double quotes for string literals
Token.sol:802:80: Error: Use double quotes for string literals
Token.sol:803:27: Error: Use double quotes for string literals
Token.sol:803:80: Error: Use double quotes for string literals
Token.sol:804:27: Error: Use double quotes for string literals
Token.sol:804:80: Error: Use double quotes for string literals
Token.sol:805:27: Error: Use double quotes for string literals
Token.sol:805:80: Error: Use double quotes for string literals
Token.sol:806:27: Error: Use double quotes for string literals
Token.sol:806:80: Error: Use double quotes for string literals
Token.sol:807:27: Error: Use double quotes for string literals
Token.sol:807:80: Error: Use double quotes for string literals
Token.sol:808:27: Error: Use double quotes for string literals
Token.sol:808:80: Error: Use double quotes for string literals
Token.sol:809:27: Error: Use double quotes for string literals
Token.sol:809:80: Error: Use double quotes for string literals
Token.sol:810:27: Error: Use double quotes for string literals
Token.sol:810:80: Error: Use double quotes for string literals
Token.sol:811:27: Error: Use double quotes for string literals
```



```
Token.sol:811:80: Error: Use double quotes for string literals
Token.sol:812:27: Error: Use double quotes for string literals
Token.sol:812:80: Error: Use double quotes for string literals
Token.sol:813:27: Error: Use double quotes for string literals
Token.sol:813:80: Error: Use double quotes for string literals
Token.sol:814:27: Error: Use double quotes for string literals
Token.sol:814:80: Error: Use double quotes for string literals
Token.sol:815:27: Error: Use double quotes for string literals
Token.sol:815:80: Error: Use double quotes for string literals
Token.sol:816:27: Error: Use double quotes for string literals
Token.sol:816:80: Error: Use double quotes for string literals
Token.sol:817:27: Error: Use double quotes for string literals
Token.sol:817:80: Error: Use double quotes for string literals
Token.sol:818:27: Error: Use double quotes for string literals
Token.sol:818:80: Error: Use double quotes for string literals
Token.sol:819:27: Error: Use double quotes for string literals
Token.sol:819:80: Error: Use double quotes for string literals
Token.sol:820:27: Error: Use double quotes for string literals
Token.sol:820:80: Error: Use double quotes for string literals
Token.sol:821:27: Error: Use double quotes for string literals
Token.sol:821:80: Error: Use double quotes for string literals
Token.sol:822:27: Error: Use double quotes for string literals
Token.sol:822:80: Error: Use double quotes for string literals
Token.sol:823:27: Error: Use double quotes for string literals
Token.sol:823:80: Error: Use double quotes for string literals
Token.sol:824:27: Error: Use double quotes for string literals
Token.sol:824:80: Error: Use double quotes for string literals
Token.sol:825:27: Error: Use double quotes for string literals
Token.sol:825:80: Error: Use double quotes for string literals
Token.sol:826:27: Error: Use double quotes for string literals
Token.sol:826:80: Error: Use double quotes for string literals
Token.sol:827:27: Error: Use double quotes for string literals
Token.sol:827:80: Error: Use double quotes for string literals
Token.sol:828:27: Error: Use double quotes for string literals
Token.sol:828:80: Error: Use double quotes for string literals
Token.sol:829:27: Error: Use double quotes for string literals
Token.sol:829:80: Error: Use double quotes for string literals
Token.sol:830:27: Error: Use double quotes for string literals
Token.sol:830:80: Error: Use double quotes for string literals
Token.sol:831:27: Error: Use double quotes for string literals
Token.sol:831:80: Error: Use double quotes for string literals
Token.sol:832:27: Error: Use double quotes for string literals
Token.sol:832:80: Error: Use double quotes for string literals
Token.sol:833:27: Error: Use double quotes for string literals
Token.sol:833:80: Error: Use double quotes for string literals
Token.sol:834:27: Error: Use double quotes for string literals
Token.sol:834:80: Error: Use double quotes for string literals
Token.sol:835:27: Error: Use double quotes for string literals
Token.sol:835:80: Error: Use double quotes for string literals
Token.sol:836:27: Error: Use double quotes for string literals
Token.sol:836:80: Error: Use double quotes for string literals
Token.sol:837:27: Error: Use double quotes for string literals
Token.sol:837:80: Error: Use double quotes for string literals
Token.sol:838:27: Error: Use double quotes for string literals
Token.sol:838:80: Error: Use double quotes for string literals
Token.sol:839:27: Error: Use double quotes for string literals
Token.sol:839:76: Error: Use double quotes for string literals
Token.sol:840:27: Error: Use double quotes for string literals
Token.sol:840:76: Error: Use double quotes for string literals
```



```
Token.sol:841:27: Error: Use double quotes for string literals
Token.sol:841:76: Error: Use double quotes for string literals
Token.sol:842:27: Error: Use double quotes for string literals
Token.sol:842:76: Error: Use double quotes for string literals
Token.sol:843:27: Error: Use double quotes for string literals
Token.sol:843:76: Error: Use double quotes for string literals
Token.sol:844:27: Error: Use double quotes for string literals
Token.sol:844:76: Error: Use double quotes for string literals
Token.sol:845:27: Error: Use double quotes for string literals
Token.sol:845:76: Error: Use double quotes for string literals
Token.sol:846:27: Error: Use double quotes for string literals
Token.sol:846:76: Error: Use double quotes for string literals
Token.sol:847:27: Error: Use double quotes for string literals
Token.sol:847:76: Error: Use double quotes for string literals
Token.sol:848:27: Error: Use double quotes for string literals
Token.sol:848:76: Error: Use double quotes for string literals
Token.sol:849:27: Error: Use double quotes for string literals
Token.sol:849:76: Error: Use double quotes for string literals
Token.sol:850:27: Error: Use double quotes for string literals
Token.sol:850:76: Error: Use double quotes for string literals
Token.sol:851:27: Error: Use double quotes for string literals
Token.sol:851:76: Error: Use double quotes for string literals
Token.sol:852:27: Error: Use double quotes for string literals
Token.sol:852:76: Error: Use double quotes for string literals
Token.sol:853:27: Error: Use double quotes for string literals
Token.sol:853:76: Error: Use double quotes for string literals
Token.sol:854:27: Error: Use double quotes for string literals
Token.sol:854:76: Error: Use double quotes for string literals
Token.sol:855:27: Error: Use double quotes for string literals
Token.sol:855:76: Error: Use double quotes for string literals
Token.sol:856:27: Error: Use double quotes for string literals
Token.sol:856:76: Error: Use double quotes for string literals
Token.sol:857:27: Error: Use double quotes for string literals
Token.sol:857:76: Error: Use double quotes for string literals
Token.sol:858:27: Error: Use double quotes for string literals
Token.sol:858:76: Error: Use double quotes for string literals
Token.sol:859:27: Error: Use double quotes for string literals
Token.sol:859:76: Error: Use double quotes for string literals
Token.sol:860:27: Error: Use double quotes for string literals
Token.sol:860:76: Error: Use double quotes for string literals
Token.sol:861:27: Error: Use double quotes for string literals
Token.sol:861:76: Error: Use double quotes for string literals
Token.sol:862:27: Error: Use double quotes for string literals
Token.sol:862:76: Error: Use double quotes for string literals
Token.sol:863:27: Error: Use double quotes for string literals
Token.sol:863:76: Error: Use double quotes for string literals
Token.sol:864:27: Error: Use double quotes for string literals
Token.sol:864:76: Error: Use double quotes for string literals
Token.sol:865:27: Error: Use double quotes for string literals
Token.sol:865:76: Error: Use double quotes for string literals
Token.sol:866:27: Error: Use double quotes for string literals
Token.sol:866:76: Error: Use double quotes for string literals
Token.sol:867:27: Error: Use double quotes for string literals
Token.sol:867:76: Error: Use double quotes for string literals
Token.sol:868:27: Error: Use double quotes for string literals
Token.sol:868:76: Error: Use double quotes for string literals
Token.sol:869:27: Error: Use double quotes for string literals
Token.sol:869:76: Error: Use double quotes for string literals
Token.sol:870:27: Error: Use double quotes for string literals
```



```
Token.sol:870:76: Error: Use double quotes for string literals
Token.sol:939:32: Error: Variable "spender" is unused
Token.sol:939:49: Error: Variable "addedValue" is unused
Token.sol:970:20: Error: Avoid to make time-based decisions in your
business logic
Token.sol:984:34: Error: Variable "recipient" is unused
```

Software analysis result:

These software reported many false positive results and some are informational issues. So, those issues can be safely ignored.

EtherAuthority

Email: audit@EtherAuthority.io