Security Audit of everdome

Conclusion

In the final contract were not found:

- Backdoors for investor funds withdrawal by anyone.
- Bugs allowing to steal money from the contract.
- Other security problems.

Obvious errors or backdoors were not found in the contract.

The client was acknowledged about all secutiry notes below.

SECURED 100% SECURED

Scope

everdome-io-master.zip

md5sum: 75e9ef58730e39fcb21f4b8d0c7556df

a2c2f8714afb9626a8616eaa1fa70579

contracts/Everdome.sol

0c30c76cef347340d498d66710157c34

contracts/Tools.sol

d5b46b7ce3b344ef5374ad30ddcc7198

contracts/VestingContract.sol

7c842b121c97144cd39dc033dee29bd1

contracts/VestingController.sol

2953b7aad83bf6d2317306b6c2f017d7

contracts/interfaces.sol

Methodology

1. Blind audit. Try to understand the structure of the code.
2. Find info in internet.
3. Ask quiestions to developers.
4. Draw the scheme of cross-contracts interactions.
5. Write user-stories, usage cases.

6. Run static analyzers

Find problems with:

- backdoors
- bugs
- math
- potential leaking of funds
- potential locking of the contract
- validate arguments and events
- others

Result

Critical

1. OWNER UN-INITIALIZED LEADS TO BROKEN CONTRACT.

At:

- contracts/VestingController.sol:8
  you have inhereted from Ownable, but this is never initialized.
  That's why passing modifier checks onlyOwner will never be possible.

Recommendation

Set the owner.

Major

Not found.

Warning

1. OLD-STYLE SAFEMATH

- contracts/Everdome.sol:147

it will not work since solidity 0.8.0 already include it under-the-hood (with no revert message)
https://docs.soliditylang.org/en/v0.8.11/080-breaking-changes.html (https://docs.soliditylang.org/en/v0.8.11/080-breaking-changes.html)
you should remove SafeMath or use modern version for 0.8.0+
e.g.
https://github.com/OpenZeppelin/openzeppelin-

contracts/blob/v4.4.1/contracts/utils/math/SafeMath.sol (https://github.com/OpenZeppelin/openzeppelin-contracts/blob/v4.4.1/contracts/utils/math/SafeMath.sol)

Recommendation

Remove or upgrade.

2. Missing zero-address checks.

At:

- contracts/Everdome.sol:389
- contracts/Everdome.sol:408

they is no check for zero-address.

Recommendation

Add the check.

3. Declare methods external.

Use `external` modifier instead of `public` for methods which are not going to be used internaly.

Recommendation

Use `external`.

4. Wrong content of interfaces.sol

In the file interfaces.sol there are actually not only interfaces.

Recommendation

1. Split the contracts/libraries/interfaces one per file.
2. Use imports from Openzeppelin, you don't need to copy-paste their libraries into your repository, you can import them.

5. Extensive deploys lead to excess gas consumption

At:

- contracts/VestingController.sol:63
  you deploy new contract with the same implementation for every vault.
  Deployment is very expensive.

Recommendation

Use MinimalProxy just to allocate new storage, but use delegatecall-s to the single implementation.

6. Lack of zero-address validation

At:

- contracts/VestingContract.sol:16
  you set the address variable, without check for zero address. This is a very common community accepted practice.

Recommendation

Do require(value != address(0), "ZERO_ADDRESS");

7. Lack of IBEP20 success status

At:

- contracts/VestingContract.sol:8
  you have everdomeToken is IBEP20
  however at:
- contracts/VestingContract.sol:36
- contracts/VestingContract.sol:52
- contracts/VestingController.sol#39
  you don't check the success status of the transfer.

Recommendation

Use SafeERC20 library

8. Tools should be library or abstract contract.

At:

- contracts/Tools.sol:4
  you define Tools as a contract, but it's never used as a real contract.

Recommendation

Tools should be library or abstract contract.

9. Unclear logic of Merkle Tree.

At:

- contracts/Tools.sol
  Merkle Tree is not the easiest code and the usage is not 100% clear.

Recommendation

Add more comments inside the code to make the logic 100% transparent, verifieble and bug-free.

10. Confusing method name.

At:

- contracts/VestingContract.sol:34
  it's actually a push, not pull. Tokens are transfered from sender to the contract.

Recommendation

Rename, add more comments about how this method will be used.

11. Unclear logic of amountAvailableToWithdraw.

At:

- contracts/VestingContract.sol:27
  you use some not realy clear algorithm to calculate available tokens to withdraw, you sum up, divide and subsctract some numbers, but the high-level idea is not obvious from the first sight.

Recommendation

Add more comments inside the code to make the logic 100% transparent, verifieble and bug-free.

12. Unclear remainingBNB calculation.

At:

- contracts/VestingContract.sol:75
  some unclear code
  uint remainingBNB = (boughtAmount- amountAvailable) * tokenPriceInBNB / (10**everdomeToken.decimals()) ;

Recommendation

uint remainingBNB = msg.value - amountAvailable * tokenPriceInBNB / (10**everdomeToken.decimals()) ;

13. USE NONREENTRANT EVERYWHERE.

There are a lot of external contracts calls and eth transfering. There is a risk of unnoticed reentry attack.

Recommendation

Use **https://github.com/OpenZeppelin/openzeppelin-contracts/blob/master/contracts/security/ReentrancyGuard.sol** (https://github.com/OpenZeppelin/openzeppelin-contracts/blob/master/contracts/security/ReentrancyGuard.sol) everywhere.

Comment

1. WRONG SPDX-LICENSE-IDENTIFIER

- contracts/Everdome.sol:1
  it should be Unlicensed

Recommendation

Fix misspelling.

2. _MSGSENDER IS NOT NEED

You dont really need to use this function.
There are no metatransactions or context switching in the code. It's just wasting of gas.

Recommendation

Replace with msg.sender

3. USE REUSABLE MODIFIERS

In a lot of places in the code there is a check:

```
require(isAdmin[msg.sender],"only-admin");
```

Recommendation

Do not copy-paste the same check again and again.
Just make a modifier.

4. EMIT EVENT IN IMPORTANT SETTING CHANGE.

At:

- contracts/Everdome.sol:408
- contracts/Everdome.sol:413
- contracts/Everdome.sol:419

- contracts/Everdome.sol:424

there is no event emited.

Recommendation

Emit event in important setting change.

5. Name private variables prefixed with underscore.

At:

- contracts/Everdome.sol:386

there is no underscore at the beginning of the name.

Recommendation

Rename.

6. Use constants.

At:

- contracts/Everdome.sol:394
- contracts/Everdome.sol:395

they are constant but set inside constructor.

Recommendation

Use class level constants or immutable declaration.

7. Follow the language style.

Spaces are missed before `{`, and after `,`.

Recommendation

Follow the language style.

8. Redundant brackets.

At:

- contracts/Everdome.sol:32
- contracts/Everdome.sol:33
  you don't need brackets.

Recommendation

It's not necessary to use brackets there.

9. Exclude double-calculation.

At:

- contracts/Everdome.sol:32
- contracts/Everdome.sol:33
  you calculate the same value twice.

Recommendation

Calculate this value on the front-end side, and pass __totalSupply already with calculated value, instead of:

```
    _totalSupply = __totalSupply * (10**__decimals);
    _balances[owner] = __totalSupply * (10**__decimals);
```

do

```
    _totalSupply = __totalSupply;
    _balances[owner] = __totalSupply;
```

it’s also the common way to pass totalSupply in constructor already in the right scale, multiplication inside the constructor is confusing.

10. Avoid reading storage.

At:

- contracts/Everdome.sol:36

you read the value from the storage, it’s expensive.

Recommendation

Take the value form calldata or from memory variables.

11. Uneccesary set.

At:

- contracts/VestingController.sol:23
  no need to set it false, since the default value is already false.

Recommendation

Remove the set.

Slither static-analyzer log

```
Compilation warnings/errors on contracts/VestingController.sol:
Warning: SPDX license identifier not provided in source file. Before publishing,
--> contracts/VestingContract.sol


Compilation warnings/errors on contracts/VestingContract.sol:
Warning: SPDX license identifier not provided in source file. Before publishing,
--> contracts/VestingContract.sol



[91m
VestingContract.pull(uint256) (contracts/VestingContract.sol#34-38) ignores retu
VestingContract.widthdrawAvailable() (contracts/VestingContract.sol#49-54) ignore
VestingController.withdrawAll(address) (contracts/VestingController.sol#34-41) i
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unchecke
[93m
VestingContract.getLegalPercentage() (contracts/VestingContract.sol#40-47) perfo
        -percentage = STARTING_PERCENTAGE + WEEK_PERCENTAGE * ((block.timestamp -
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#divide-b
[93m
Reentrancy in VestingController.claimTokens(uint256,uint256,bytes32[]) (contracts
        External calls:
        - IWhitelisted(address(everdomeToken)).setWhitelisted(address(usersVault)
        State variables written after the call(s):
        - vestingContracts[_msgSender()] = address(usersVault) (contracts/Vesting
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentra
[93m
VestingController.claimTokens(uint256,uint256,bytes32[]) (contracts/VestingContr
VestingController.claimTokens(uint256,uint256,bytes32[]) (contracts/VestingContr
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-r
[92m
VestingContract.constructor(uint256,IBEP20,address)._owner (contracts/VestingCon
        - Ownable._owner (contracts/interfaces.sol#308) (state variable)
VestingController.getVestingContract(address)._owner (contracts/VestingControlle
        - Ownable._owner (contracts/interfaces.sol#308) (state variable)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#local-va
[92m
Reentrancy in VestingContract.pull(uint256) (contracts/VestingContract.sol#34-38)
        External calls:
        - everdomeToken.transferFrom(msg.sender,address(this),amount) (contracts/
        Event emitted after the call(s):
        - PullCalled(amount) (contracts/VestingContract.sol#37)
Reentrancy in VestingContract.widthdrawAvailable() (contracts/VestingContract.so
        External calls:
        - everdomeToken.transfer(owner(),available) (contracts/VestingContract.so
        Event emitted after the call(s):
        - WithdrawAvailableCalled(available) (contracts/VestingContract.sol#53)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentra
[92m
VestingContract.pull(uint256) (contracts/VestingContract.sol#34-38) uses timesta
        Dangerous comparisons:
        - require(bool,string)(amount <= amountAvailableToBuy(),over-buy-limit) (
VestingContract.getLegalPercentage() (contracts/VestingContract.sol#40-47) uses t
        Dangerous comparisons:
        - percentage > 100 (contracts/VestingContract.sol#42)
VestingController.withdrawAll(address) (contracts/VestingController.sol#34-41) u
        Dangerous comparisons:
        - require(bool,string)(isInitialized == false || block.timestamp > start
VestingController.claimTokens(uint256,uint256,bytes32[]) (contracts/VestingContr
        Dangerous comparisons:
        - require(bool,string)(block.timestamp < start + ICO_TIMESPAN,ico-ended)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-t
[92m
VestingController.withdrawAll(address) (contracts/VestingController.sol#34-41) c
        -require(bool,string)(isInitialized == false || block.timestamp > start +
VestingController.claimTokens(uint256,uint256,bytes32[]) (contracts/VestingContr
        -hasVestingContract() == false (contracts/VestingController.sol#62)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#boolean-
[92m
Context._msgData() (contracts/interfaces.sol#288-291) is never used and should be
SafeMath.add(uint256,uint256) (contracts/interfaces.sol#131-136) is never used a
SafeMath.div(uint256,uint256) (contracts/interfaces.sol#205-207) is never used a
SafeMath.div(uint256,uint256,string) (contracts/interfaces.sol#220-231) is never
SafeMath.mod(uint256,uint256) (contracts/interfaces.sol#244-246) is never used a
SafeMath.mod(uint256,uint256,string) (contracts/interfaces.sol#259-266) is never
SafeMath.mul(uint256,uint256) (contracts/interfaces.sol#180-192) is never used a
SafeMath.sub(uint256,uint256) (contracts/interfaces.sol#147-149) is never used a
SafeMath.sub(uint256,uint256,string) (contracts/interfaces.sol#160-169) is never
```

```
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-coc
[92m
Pragma version^0.8.0 (contracts/Tools.sol#2) allows old versions
Pragma version^0.8.0 (contracts/VestingContract.sol#1) allows old versions
Pragma version^0.8.0 (contracts/VestingController.sol#2) allows old versions
Pragma version^0.8.0 (contracts/interfaces.sol#2) allows old versions
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrec
[92m
Low level call in VestingController.claimTokens(uint256,uint256,bytes32[]) (contr
       - (sent) = msg.sender.call{value: remainingBNB}() (contracts/VestingContr
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-leve
[92m
Parameter Tools.hashLeaf(uint256,uint256,address)._sender (contracts/Tools.sol#7)
Parameter Tools.nodeVerificationPasses(uint256,uint256,address,bytes32[],bytes32)
Parameter Tools.nodeVerificationPasses(uint256,uint256,address,bytes32[],bytes32)
Variable VestingContract.WEEK_PERIOD (contracts/VestingContract.sol#10) is not in
Variable VestingContract.STARTING_PERCENTAGE (contracts/VestingContract.sol#11) i
Variable VestingContract.WEEK_PERCENTAGE (contracts/VestingContract.sol#12) is no
Parameter VestingController.getVestingContract(address)._owner (contracts/Vesting
Variable VestingController.ICO_TIMESPAN (contracts/VestingController.sol#11) is n
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conforma
[92m
Redundant expression "this (contracts/interfaces.sol#289)" inContext (contracts/i
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#redundar
[92m
hashSingle(bytes32) should be declared external:
       - Tools.hashSingle(bytes32) (contracts/Tools.sol#19-21)
pull(uint256) should be declared external:
       - VestingContract.pull(uint256) (contracts/VestingContract.sol#34-38)
widthdrawAvailable() should be declared external:
       - VestingContract.widthdrawAvailable() (contracts/VestingContract.sol#49-
initialize() should be declared external:
       - VestingController.initialize() (contracts/VestingController.sol#29-32)
withdrawAll(address) should be declared external:
       - VestingController.withdrawAll(address) (contracts/VestingController.sol
availableToBuy() should be declared external:
       - VestingController.availableToBuy() (contracts/VestingController.sol#47-
claimTokens(uint256,uint256,bytes32[]) should be declared external:
       - VestingController.claimTokens(uint256,uint256,bytes32[]) (contracts/Ves
getVestingContract(address) should be declared external:
       - VestingController.getVestingContract(address) (contracts/VestingControl
renounceOwnership() should be declared external:
       - Ownable.renounceOwnership() (contracts/interfaces.sol#346-349)
transferOwnership(address) should be declared external:
       - Ownable.transferOwnership(address) (contracts/interfaces.sol#355-357)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-f
[92m
Pragma version^0.8.0 (contracts/Tools.sol#2) allows old versions
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrec
[92m
Parameter Tools.hashLeaf(uint256,uint256,address)._sender (contracts/Tools.sol#7)
Parameter Tools.nodeVerificationPasses(uint256,uint256,address,bytes32[],bytes32)
Parameter Tools.nodeVerificationPasses(uint256,uint256,address,bytes32[],bytes32)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conforma
[92m
hashSingle(bytes32) should be declared external:
       - Tools.hashSingle(bytes32) (contracts/Tools.sol#19-21)
nodeVerificationPasses(uint256,uint256,address,bytes32[],bytes32) should be decla
       - Tools.nodeVerificationPasses(uint256,uint256,address,bytes32[],bytes32)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-f
[91m
VestingContract.pull(uint256) (contracts/VestingContract.sol#34-38) ignores retur
VestingContract.widthdrawAvailable() (contracts/VestingContract.sol#49-54) ignore
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unchecke
[93m
VestingContract.getLegalPercentage() (contracts/VestingContract.sol#40-47) perfor
       -percentage = STARTING_PERCENTAGE + WEEK_PERCENTAGE * ((block.timestamp -
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#divide-b
[92m
VestingContract.constructor(uint256,IBEP20,address)._owner (contracts/VestingCont
       - Ownable._owner (contracts/interfaces.sol#308) (state variable)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#local-va
[92m
Reentrancy in VestingContract.pull(uint256) (contracts/VestingContract.sol#34-38)
       External calls:
       - everdomeToken.transferFrom(msg.sender,address(this),amount) (contracts/
       Event emitted after the call(s):
       - PullCalled(amount) (contracts/VestingContract.sol#37)
Reentrancy in VestingContract.widthdrawAvailable() (contracts/VestingContract.sol
        External calls:
        - everdomeToken.transfer(owner(),available) (contracts/VestingContract.so
        Event emitted after the call(s):
        - WithdrawAvailableCalled(available) (contracts/VestingContract.sol#53)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentran
[92m
VestingContract.pull(uint256) (contracts/VestingContract.sol#34-38) uses timestam
        Dangerous comparisons:
        - require(bool,string)(amount <= amountAvailableToBuy(),over-buy-limit) (
VestingContract.getLegalPercentage() (contracts/VestingContract.sol#40-47) uses t
        Dangerous comparisons:
        - percentage > 100 (contracts/VestingContract.sol#42)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-ti
[92m
Context._msgData() (contracts/interfaces.sol#288-291) is never used and should be
SafeMath.add(uint256,uint256) (contracts/interfaces.sol#131-136) is never used an
SafeMath.div(uint256,uint256) (contracts/interfaces.sol#205-207) is never used an
SafeMath.div(uint256,uint256,string) (contracts/interfaces.sol#220-231) is never
SafeMath.mod(uint256,uint256) (contracts/interfaces.sol#244-246) is never used an
SafeMath.mod(uint256,uint256,string) (contracts/interfaces.sol#259-266) is never
SafeMath.mul(uint256,uint256) (contracts/interfaces.sol#180-192) is never used an
SafeMath.sub(uint256,uint256) (contracts/interfaces.sol#147-149) is never used an
SafeMath.sub(uint256,uint256,string) (contracts/interfaces.sol#160-169) is never
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-cod
[92m
Pragma version^0.8.0 (contracts/VestingContract.sol#1) allows old versions
Pragma version^0.8.0 (contracts/interfaces.sol#2) allows old versions
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrec
[92m
Variable VestingContract.WEEK_PERIOD (contracts/VestingContract.sol#10) is not in
Variable VestingContract.STARTING_PERCENTAGE (contracts/VestingContract.sol#11) i
Variable VestingContract.WEEK_PERCENTAGE (contracts/VestingContract.sol#12) is no
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conforma
[92m
Redundant expression "this (contracts/interfaces.sol#289)" inContext (contracts/i
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#redundan
[92m
pull(uint256) should be declared external:
        - VestingContract.pull(uint256) (contracts/VestingContract.sol#34-38)
widthdrawAvailable() should be declared external:
        - VestingContract.widthdrawAvailable() (contracts/VestingContract.sol#49-
renounceOwnership() should be declared external:
        - Ownable.renounceOwnership() (contracts/interfaces.sol#346-349)
transferOwnership(address) should be declared external:
        - Ownable.transferOwnership(address) (contracts/interfaces.sol#355-357)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-f
[92m
Everdome.constructor(address,uint256,uint8).owner (contracts/Everdome.sol#24) sha
        - Ownable.owner() (contracts/interfaces.sol#327-329) (function)
Everdome.allowance(address,address).owner (contracts/Everdome.sol#132) shadows:
        - Ownable.owner() (contracts/interfaces.sol#327-329) (function)
Everdome._approve(address,address,uint256).owner (contracts/Everdome.sol#284) sha
        - Ownable.owner() (contracts/interfaces.sol#327-329) (function)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#local-va
[92m
Context._msgData() (contracts/interfaces.sol#288-291) is never used and should be
SafeMath.div(uint256,uint256) (contracts/interfaces.sol#205-207) is never used an
SafeMath.div(uint256,uint256,string) (contracts/interfaces.sol#220-231) is never
SafeMath.mod(uint256,uint256) (contracts/interfaces.sol#244-246) is never used an
SafeMath.mod(uint256,uint256,string) (contracts/interfaces.sol#259-266) is never
SafeMath.mul(uint256,uint256) (contracts/interfaces.sol#180-192) is never used an
SafeMath.sub(uint256,uint256) (contracts/interfaces.sol#147-149) is never used an
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-cod
[92m
Pragma version^0.8.0 (contracts/Everdome.sol#2) allows old versions
Pragma version^0.8.0 (contracts/interfaces.sol#2) allows old versions
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrec
[92m
Everdome (contracts/Everdome.sol#7-294) should inherit from IWhitelisted (contrac
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-
[92m
Variable Everdome._totalSupply (contracts/Everdome.sol#17) is not in mixedCase
Variable Everdome._decimals (contracts/Everdome.sol#18) is not in mixedCase
Variable Everdome._symbol (contracts/Everdome.sol#19) is not in mixedCase
Variable Everdome._name (contracts/Everdome.sol#20) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conforma
[92m
Redundant expression "this (contracts/interfaces.sol#289)" inContext (contracts/i
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#redundan
```

```
[92m
clearLocked() should be declared external:
	- Everdome.clearLocked() (contracts/Everdome.sol#39-41)
setAdmin(address) should be declared external:
	- Everdome.setAdmin(address) (contracts/Everdome.sol#43-46)
removeAdmin(address) should be declared external:
	- Everdome.removeAdmin(address) (contracts/Everdome.sol#48-52)
setWhitelisted(address) should be declared external:
	- Everdome.setWhitelisted(address) (contracts/Everdome.sol#54-57)
removeWhitelisted(address) should be declared external:
	- Everdome.removeWhitelisted(address) (contracts/Everdome.sol#59-63)
increaseAllowance(address,uint256) should be declared external:
	- Everdome.increaseAllowance(address,uint256) (contracts/Everdome.sol#198
decreaseAllowance(address,uint256) should be declared external:
	- Everdome.decreaseAllowance(address,uint256) (contracts/Everdome.sol#224
renounceOwnership() should be declared external:
	- Ownable.renounceOwnership() (contracts/interfaces.sol#346-349)
transferOwnership(address) should be declared external:
	- Ownable.transferOwnership(address) (contracts/interfaces.sol#355-357)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-f
[92m
Context._msgData() (contracts/interfaces.sol#288-291) is never used and should be
SafeMath.add(uint256,uint256) (contracts/interfaces.sol#131-136) is never used an
SafeMath.div(uint256,uint256) (contracts/interfaces.sol#205-207) is never used an
SafeMath.div(uint256,uint256,string) (contracts/interfaces.sol#220-231) is never
SafeMath.mod(uint256,uint256) (contracts/interfaces.sol#244-246) is never used an
SafeMath.mod(uint256,uint256,string) (contracts/interfaces.sol#259-266) is never
SafeMath.mul(uint256,uint256) (contracts/interfaces.sol#180-192) is never used an
SafeMath.sub(uint256,uint256) (contracts/interfaces.sol#147-149) is never used an
SafeMath.sub(uint256,uint256,string) (contracts/interfaces.sol#160-169) is never
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-coc
[92m
Pragma version^0.8.0 (contracts/interfaces.sol#2) allows old versions
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrec
[92m
Redundant expression "this (contracts/interfaces.sol#289)" inContext (contracts/i
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#redundar
[92m
owner() should be declared external:
	- Ownable.owner() (contracts/interfaces.sol#327-329)
renounceOwnership() should be declared external:
	- Ownable.renounceOwnership() (contracts/interfaces.sol#346-349)
transferOwnership(address) should be declared external:
	- Ownable.transferOwnership(address) (contracts/interfaces.sol#355-357)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-f
contracts analyzed (26 contracts with 77 detectors), 124 result(s) found
```