Audit Interim Report

This is an interim Smart Contract Audit Report that is executed for proper communication between Saif Sghaier and its clients. This is not to be considered a final report.

Project Name - *Widcoin*
Project Platform - *EVM*
Project Language - *Solidity*

Project Contract Link -*<Enter CodeBase link here>*

Project CodeBase - *Provided as a file*
Project Commit - *<Enter Commit Hash for codebase here>*

File Details

<Enter Name of File and Give It A File ID>
*<File ID **Naming Conventio**n - File ID will contain 3 letters. The first two letters will be initials from the project name, the third letter will be the initial of the file name. If two or more files contain same initial, then a fourth letter might be added to distinguish between them>*
*<File ID **Example** -*
Project name - Lightning Works
File names - LW0-Contract.sol, LW0-Minter.sol, LW0-Simple.sol
File IDs - LWC, LWM, LWS
Issues IDs- LWC01, LWC02, LWC03 etc.>

File ID	**File Name**
PRSLE	PRSLE-Presale.sol

Audit Details

Report Submission Date - 6/10/2024
Result - Passed

Findings Details

Severity	Number Of Issues	Percentage
Critical	1	20%
High	0	0%
Medium	1	20%
Low	1	20%
Informational	2	40%

Finding Summary

<Keep all the issues from one file together while filling out this table. Once all issues from one file are done, then move on to next.>

*<Issues **Status** Details -*

__Reported__ - When Issue is first reported.

__Acknowledged__ - If client has seen the issues but not taken any action

__Resolved__ - If client has seen the issue and fixed it>

Issue ID	Type	Line	Severity	Status
PSRLE-01	Logic error in calculation in buyToken function	227	Critical Severity	Resolved
PSRLE-02	addressToHasClaimedPresale Tokens not used	84	Medium Severity	Resolved
PSRLE-03	Centralization Risk	-	Low Severity	Acknowledged
PSRLE-04	Unused Global Variable/Custom errors	45 / 101 / 104	Informational	Acknowledged
PSRLE-05	Unused Private/Internal Functions	452-458	Informational	Acknowledged

<Separate Issue Page - Copy this page for every issue and enter it's specific details. Do not write two issues in the same page>

Issue ID - *PRSLE-01*

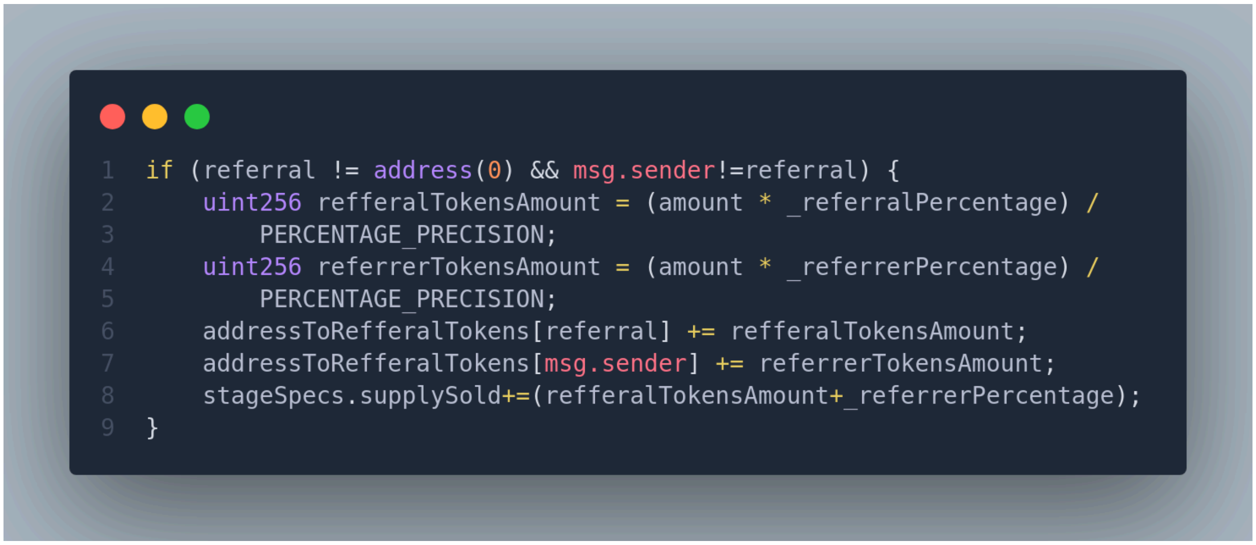
Type - Logic error in calculation in buyToken function
Severity - Critical Severity
File - Presale.sol
Line - 227
Status - Resolved

Description - In the buyToken function the supply sold is calculated by adding refferalTokensAmount and _referrerPercentage which seems to be the wrong way to do it. Logically in should be :

stageSpecs.supplySold+=(refferalTokensAmount+referrerTokensAmount);.

Remediation - Fix the logic unless it is intended to be that way.

SnapShot -



```
if (referral != address(0) && msg.sender!=referral) {
    uint256 refferalTokensAmount = (amount * _referralPercentage) /
        PERCENTAGE_PRECISION;
    uint256 referrerTokensAmount = (amount * _referrerPercentage) /
        PERCENTAGE_PRECISION;
    addressToRefferalTokens[referral] += refferalTokensAmount;
    addressToRefferalTokens[msg.sender] += referrerTokensAmount;
    stageSpecs.supplySold+=(refferalTokensAmount+_referrerPercentage);
}
```

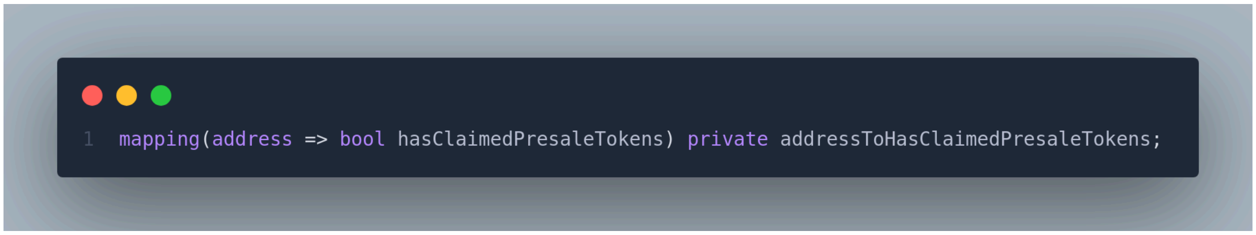
Issue ID - *PRSLE-02*

Type - addressToHasClaimedPresaleTokens not used
Severity - Medium Severity
File - Presale.sol
Line - 84
Status - Resolved

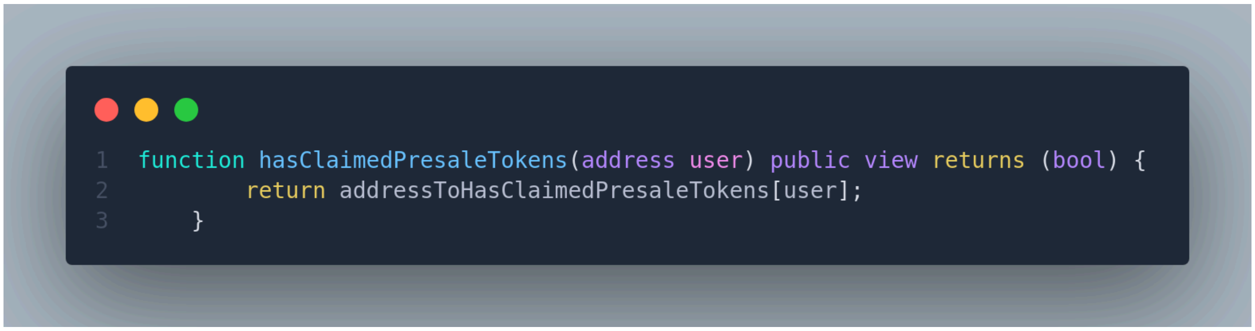
Description - The mapping addressToHasClaimedPresaleTokens is declared and has a getter function `hasClaimedPresaleTokens(address user)` but the mapping is not used at all in the contract which makes it impossible to know who has already claimed their tokens and it is also considered dead code.

Remediation - Populate the mapping when a certain user claims their presale tokens or remove it entirely to save storage and gas on deployment.

SnapShot -



```
mapping(address => bool hasClaimedPresaleTokens) private addressToHasClaimedPresaleTokens;
```



```
function hasClaimedPresaleTokens(address user) public view returns (bool) {
        return addressToHasClaimedPresaleTokens[user];
    }
```

Issue ID - *PRSLE-03*

Type - Centralization risk
Severity - Low Severity
File - Presale.sol
Line - -
Status - Acknowledged

Description - The Owner of the contract holds all the privileges. It is considered a bad practice and can lead to loss of funds or losing control of the protocol if the owner address is compromised.

Remediation - Consider adding more roles (admins) or using a multisignature.

SnapShot -
No snapshot required.

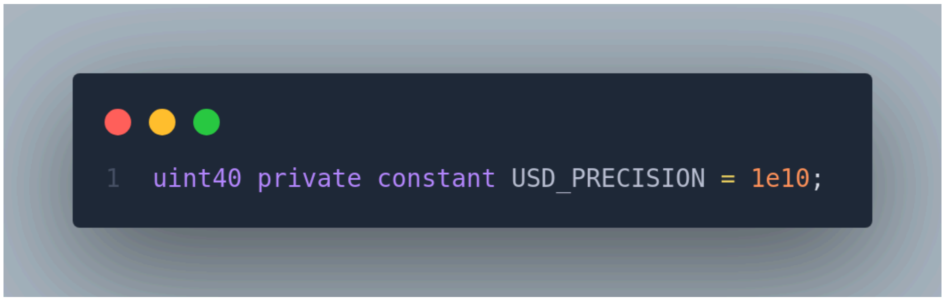
Issue ID - *PRSLE-04*

Type - Unused Global Variable/Custom errors
Severity - Informational
File - Presale.sol
Line - 45 / 101 / 104
Status - Acknowledged

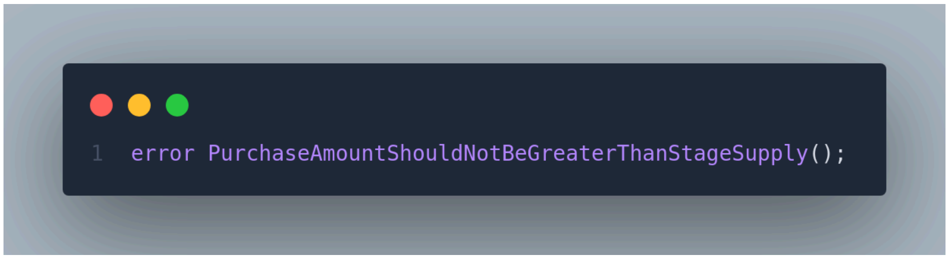
Description - The global variable and custom errors below are declared but are not used. PS: only USD_PRECISION is used. The errors are still unused!!!!

Remediation - Remove them or use them inside the functions.

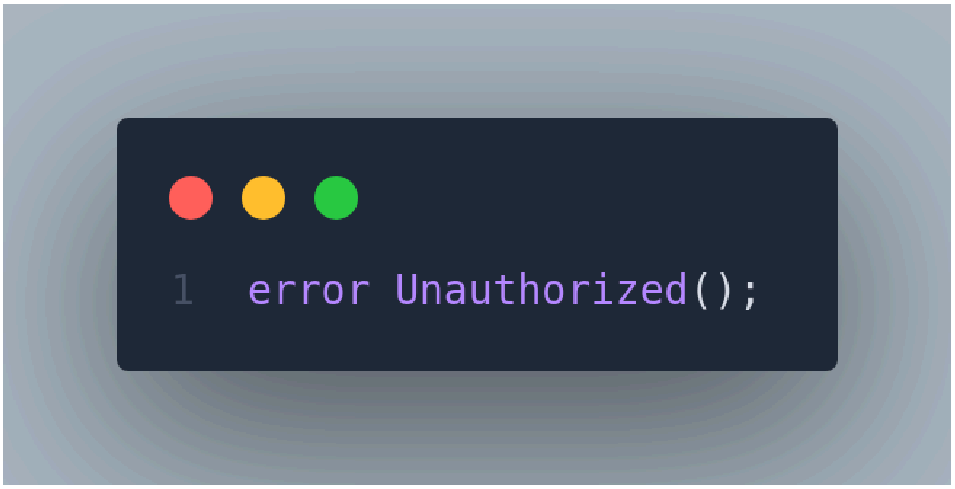
SnapShot -



```
uint40 private constant USD_PRECISION = 1e10;
```



```
error PurchaseAmountShouldNotBeGreaterThanStageSupply();
```



```
error Unauthorized();
```

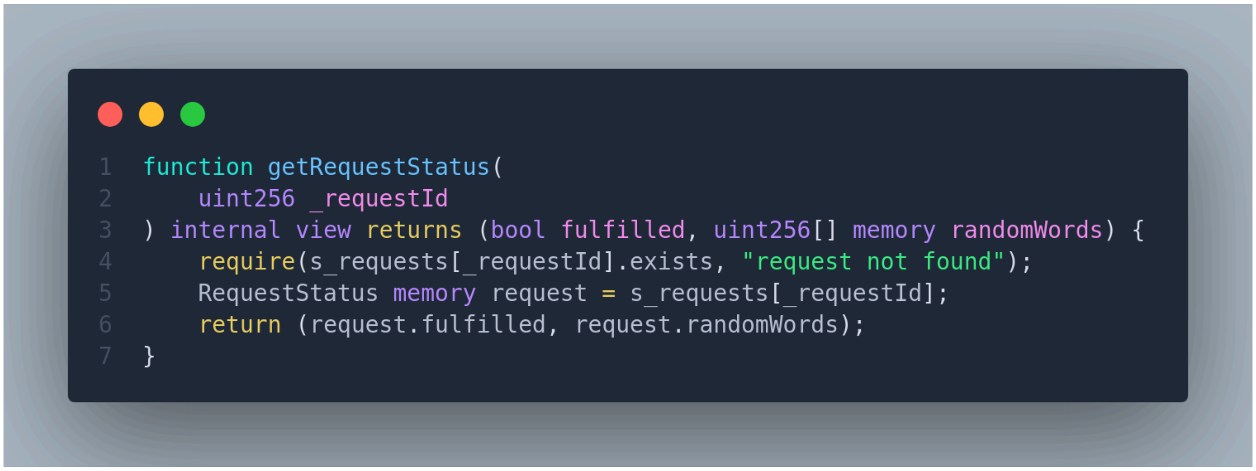
Issue ID - *PRSLE-05*

Type - Unused Private/Internal functions
Severity - Informational
File - Presale.sol
Line - 452-458
Status - Acknowledged

Description - getRequestStatus is internal, meaning it can be accessed only within another public/external function but it is not used.

Remediation - Remove it or use it inside public/external functions.

SnapShot -



```
function getRequestStatus(
    uint256 _requestId
) internal view returns (bool fulfilled, uint256[] memory randomWords) {
    require(s_requests[_requestId].exists, "request not found");
    RequestStatus memory request = s_requests[_requestId];
    return (request.fulfilled, request.randomWords);
}
```