**Why Storing amount withdrawn?**

**Referral commission BNB will be coming from Smart contract balance?**

**Audit**

Summary: StakingBNB is a token for users to stake their BNB and receive daily rewards. This also supports Referral mechanism.

|  |  |  |
| --- | --- | --- |
|  | Criteria | Grade |
| ✓ | Compiles Without Error | Pass |
| ✓ | Is Ownable | Pass |
| ✕ | Ownership is transferable / renounceable | Fail |
|  | Tokens are transferable |  |
|  | Contract avoids state changes after external calls |  |
| ✕ | Contract can receive native currency | Fail |
|  | Contract follows checks, effects, and interactions flow |  |
| ✓ | Not vulnerable to overflow (Solidity version 0.8.4) | Pass |
| ✓ | Properly emits relevant events |  |
| ✕ | All require/revert/assert statements have proper error. | Fail |
|  | Withdraw function safe for withdrawal |  |
|  | **Overall** | **Fail** |

Comments:

* Will revert safe transfer transactions.
* No option to change Ownership. If Owner’s wallet gets hacked everything can be looted.
* Presence of [*withdrawContractBalance()*]is a major Red-flag, can lead people to think contract is a rug-pull scheme. **Strongly suggest to remove this and have a refund mechanism, if needed.**
* The contract includes "hardhat/console.sol", which will have no use is main-net and just increases the contract size. Should be removed before deployment.
* Growth Per day and other Fees are hard-coded and cannot be changed once code is deployed.
* No BNB is transferred to Owner at all.
* Performance Fee is set as 25%, instead of 10%.
* Only Owner functions do not provide a reason when reverting.
* Expired Staked should be deleted, else required gas will increase overtime can also exceed maximum allowed value.
* Daily max withdrawal limit can be changes only by staking amount greater than current max limit. Total amount staked has no effect on daily withdrawal limit. Is this by design?
* No method to see current active stake.
* A user cannot register without having a referral. Is this by design?
* No read-only (view) function to get a user’s total Earning, growth, team earnings, team members. This will be required for the UI.
* The contract has separate withdraw functions for each type of earning, this is against the requirement where all earnings should be accumulated and a single withdraw function should be used.
* Contract contains many internal functions that are not ever used. These should be removed since their presence affects contract size and has deployment cost.
* ReferralLevels structure uses hard-coded max-referral-stack, It will not be reliable if max-referral-stack is increased.
* No BlackList/ Suspend functionality
* No mechanism to pause daily ROI when stake is paused.

Table 1.1 Major View functions contained in the contract

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Description | Access | Remark | Status |
| getDailyEarningsRate | Get Daily ROI % based on total amount staked \* 100 | Internal /pure |  | Pass |
| getUserFunds | Get all stakes by a user |  |  | Pass |
| checkMaxEarnings | Maximum possible user earnings |  |  | Pass |
| checkMaxWithdrawal | Tests withdrawals limit for 24hour period |  |  | Pass |
| checkAntiWhaleTaxes | Anti-Whale tax |  |  | Pass |
| getRelativePercentage | Division | Internal / pure |  | Pass |
| \_getMaximumReferralStack | Check how many referrals Level user is eligible to get commission from. | Internal | * Does not check is user has active stake or not * Cannot handle max\_referral\_Stack change | Fail |
| getContractBalance | Total contract balance |  | Not Required | Pass |
| \_isAdminOrNullAddress |  | internal |  | Pass |
| \_getInitialReferral | Users/ referrer | internal | Internal function never used | Must Be Removed |

Table 1.2 Major State Modifying functions contained in the contract

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Description | Access | Remarks | Status |
| pauseContract() | To pause the contract | onlyOwners | * Has no effect on contracts working. * Should be marked external to save gas. | Fail |
| activeContract() | To activate the contract | onlyOwner | * Has no effect on contracts working. * Should be marked external to save gas. | Fail |
| registerUser() |  | Public | * Should be marked external to save gas. | Acceptable |
| \_registerUser | To refer a user | Internal | * Referee is adding referrer, not the other way around, anyone can add anyone as their referee, though it might be useless, but an invite mechanism could have been better. * Though user level is defined while registering a user, it is never used. | Pass |
| stakeBnb | To stake BNBs, function receiving BNB. | Public | * Changing max-Withdrawal only when new staked amount is greater than previous value, not on increase of total staked amount. * Should be marked external to save gas. * Does not allow staking exactly 0.01 BNB or 500BNB; * This call calculateDailyEarnings, why is that needed? * 50% of deposit fee is not transferred to owner. * Referral Commission earning is timedependent, (line 184). It should be fixed percentage of deposited amount. | Fail |
| calculateDailyEarnings | Calculates Total Daily Interest Gained from last withdrawal |  | * Considering only the last stake. Fails when   multiple active stakes are present.   * Should not be payable * Not considering Performance Fees | Fail |
| \_updateRefBalance - R | Updates referral Bonus for all referrers | Internal | * Referral Commission earning is time dependent, (line 184). It should be a fixed percentage of deposited amount. * Does not follow check-effect-interact | Fail |
| calculateReferalEarnings - R | Updates total referral earnings for all levels eligible | Public/Registered user | * Should not be payable      * Cannot handle max\_refferal\_stack change. * Does not check presence of active stake. |  |
| withdrawOwnBonus | Withdraw own Earning(daily ROI) |  | * Should be marked external to save gas. * Scenario failing -> single initial Stake = 100BNB 50BNB withdrawn at 100th day. Final balance is getting reduced to 17.5 BNB, instead of 40BNB. | Fail |
| withdrawReferralBonus- R | Withdraw ReferralBonus |  | * Should be marked external to save gas. |  |
| withdrawReferralCommission - R |  |  | * Should be marked external to save gas. |  |
| WithdrawBalacne-R | To be used by user to withdraw tokens |  |  |  |
| updateRefAddrs - R | Used when a new user is added. | internal |  |  |
| deductWithdrawalFees - R | Deducts withdrawal fees and provided referral commission | Internal | * Does not send Fee share to admin | Fail |
| deductPerformanceFees | Deduct Performance Fee in Daily ROI |  | * Performance Fee set to 25% instead of 10%. * Does not send Fee share to admin | Fail |
| \_distributeReferralBonus -R | Distribute performance referral between referrers. |  | * Does not check Team stake to be greater than 100BNB * Use of multiple if statements inside for loop defeats the purpose of for loop. This should be corrected to reduce gas consumption. * Will fail if Max\_referral\_stack size is changed. | Fail |
| withdrawContractBalance-R | To withdraw all BNB stored in a contract | Public/OnlyOwner | * Major Red Flag, will discourage users to put money. | Alternate method should be use |
| checkTotalEarnings - R | Users total earning |  |  |  |
| getTotalActiveStake- R |  | internal | Internal function that is Never Used | Must be Removed |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| \_deductDepositFees-R | Deposit Fee calculation | Internal | * Does not transfer Fee share to owner | Fail |
| \_deductReferralFees- R | Referral Fee calculation | Internal | * Does not transfer Fee share to owner | Fail |

Use Case:

~~Single user , Single stake~~

~~Single User, multiple Stake,~~

Referral

All stake active throughout

Stake inactive in between

Parent no stake

Max withdrawal test

Pause – daily ROI should be stopped