**Alpha Shark Token Audit**

View functions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Description | Access | Remark | Status |
| ifBoostTypeAvailable | Check if a particular boost type is available | Public | * Why public?   Using for loop, might consume too much gas if multiple boost types are introduced in future. Also adds gas to internal functions. |  |
| returnSingleSharkData | Get Data for a shark | Public |  |  |
| returnOwnerAllSharkData |  | Public | Revisit |  |
| getOwnerList | To view all tokens owned by user. | External /public |  |  |
| getActiveBoost | Get active Boost for a particular shark | Public/External |  |  |
| CheckFixedBooster | Check fixed claimable reward percentage |  | * Adding 30 and 10 units, issue when converted to wei. By metamask and others * Not considering time period of tokenID<1000 && oddTokenID |  |
| calculateTotalRewards | Total claimable reward percentage | Public | * Dependent on CheckFizedBooster. * All members of Shiver get 100% boost. * S3 boosts claimable only post expiry, by design? * Not checking if claim is stopped or not. This will create a discrepancy when claims are stopped because users will be able to see how much they should be able to claim, but they will be denied claim only when they actually try to claim. |  |
| availableTokens | Total available tokens | Public | Not checking if totalTokens< lockedTokens, might be possible because of DutchRaffle |  |
|  |  |  |  |  |
|  |  |  |  |  |

State Modifying functions:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Description | Access | Remarks | Status |
| Initialize() | To initialize the contract | Alpha\_shark\_factory/ deployer | does not affect Default\_admin\_role,only ownership is transferred. |  |
| updateRewardAddress() | Change reward token address | onlyOwner |  |  |
| toggleClaimReward | Allow/restrict award claims | onlyOwner |  |  |
| addBooster | Add new booster type. | Only booster |  |  |
| assignBooster | Assign booster to a user. | Only Booster | * Check if assigned boosts are removed post expiry. This will add gas for longer terms users. |  |
| userBoostAvailabitiy | Check if user has particular boost Type alloted | Internal | * Removes boost type if found. This will might be troublesome when a user has multiple units of particular boost Type,say junior shark. Not valid if availableBoosts is storing only active boosts. * Removal is being done from local copy of available boosts, not in actual mapped array. CHECK |  |
| activateBooster | Activate available boost on a shark. | Public | * Dependent on userBoostAvailability, so possessing multiple boost will be an issue. * User might able to activate multiple boosts at the same time, since push operation is not checking active boosts. * Using storage variable for \_shark, more gas. * Only caller can use owned boosts, no approve() mechanism. |  |
| claimRewards | Claim rewards. | Public/External | * Maximum claimable reward is 400 due ti extra 100 added in formula (line 297) |  |
| claimAll | Claim rewards for all tokens | Public/external | * Any claimer can claim rewards for any token ID |  |
| updateAddLockTokens | Updated locked tokens | Only NFT |  |  |
| updateTotalLockTokens | deca | Only DutchRaffle | * Updating only total tokens, not locked tokens |  |
| withdawTokens | Withdraw tokens | Public | * Not following check-effect-interact, susceptible to rentrancy attack.   (not following recommended practice, safe since only interacting with our token), But should be avoided. |  |
| depositTokens | ?????? |  | Anyone can send tokens to this contract, can they retrieve? |  |
| stakeNFT |  | Staker Only |  |  |
| checkShiverBreak |  | Internal |  |  |
| removeTokenFromOwnerList |  | internal |  |  |
| unStakeNFT |  | Staker Only |  |  |
| makeShiver |  | tokenOwner/ Alpha shark factory / Booster. |  |  |
| emergencyRewardWithdraw |  | onlyOwner | safeTransfer? |  |
| stopBooster | Stop existing boosterc | Only Booster | Check implementation because of storage usage. |  |
| updateAllRewards | Update per day rewards for multiple tokens |  | For loop, high gas. |  |
| updateSingleReward | Update single token per day reward |  |  |  |
|  |  |  |  |  |

ReentrancyGuard constructor redundant -> line 91;

---------------------------

Boost activated by the smart contract. -> explanation required. -> user to activate boost.

Boost -> “*There are 3 category of these boosters. Only one of each type of boost can be activated on a Shark*” -> Can another boost be applied on same shark post expiry of previous shark.

**Can users transfer Shark tokens/Available Boosts/Junior Sharks to non-Shark holders/other holders?**

**S1 boost-> minting first 1000 & minting odd -> mutually exclusive ? ,i.e., odd sharks minted within first 1000 to receive 30% or (30% + 10%) boost.**

**Boost from Junior shark to have expiry or to be valid until junior shark is unstaked?**

*If a bid is lost,* ***tokens are reclaimable*** *and can be used for bidding on* ***any other whitelist. ->*  can reclaimed tokens be used only for raffle, or other purposed too?**

**.**

Test Conditions:

1. Deployment with correct variables set.
2. Check use of admin only methods before initialization.
3. Initialization does not affect Default\_admin\_role,only ownership is transferred.

--------------------------------------------------------------------------------------

Making a Shiver will boost all Sharks by 100%, is this by design?

Check role assignment

Multiple for loops used. -> This will increase gas costs and possibility of required gas exceeding maximum allowed. Better to limit number of boosters that can be collected by a user to some fixed value.