KlayBank Audit Report - Staking

Audited By : Creative Carrot 🥕

Previously Audited : Claimswap, Kleva Protocol

Audit Target

Github : https://github.com/klaybank/klaybank-audit-shared

Commit Hash : 6b4045002602dee42a1eb3db30c1c88e8b109859

Files : `./klaybank-stake/contracts/**/*.sol`

CRITICAL : `KlaybankDistributionManager._getAssetIndex()` will drop last month's reward when updated after distribution ends

STATUS : FIXED ON 9322b504307b07c15f6dc720f08848c1c96f95a3

> contracts/stake/KlaybankDistributionManager.sol:293: vars.currentMonthTimeDelta = 0; // CC : CRITICAL - this has to be SECONDS_OF_ONE_MONTH

https://github.com/klaybank/klaybank-audit-shared/blob/main/klaybank-stake/contracts/stake/KlaybankDistributionManager.sol#L293

If asset gets updated "after" the distribution ends, asset will ignore the remaining part of the last month since it sets the "currentMonthTimeDelta" to zero.

change vars.currentMonthTimeDelta to be SECONDS_OF_ONE_MONTH instead of zero

MAJOR : use `IStakedToken(stakedToken).GOVERNANCE_TOKEN()` instead of `GOVERNANCE_TOKEN` to ensure code functions even if stakedToken is not address(this)

STATUS : FIXED ON 9322b504307b07c15f6dc720f08848c1c96f95a3

> contracts/stake/StakedToken.sol:131: // CC : MAJOR - use IStakedToken(stakedToken).GOVERNANCE_TOKEN instead of this.GOVERNANCE_TOKEN to ensure code functionality when staking to external StakedToken

https://github.com/klaybank/klaybank-audit-shared/blob/main/klaybank-stake/contracts/stake/StakedToken.sol#L131

StakedToken contract has stakedToken variable to allow chained stake to multiple vaults.

Since it is not guaranteed that StakedToken itself and stakedToken has same `GOVERNANCE_TOKEN`, there can be issue while calling `stakedToken.stake()` since StakedToken did not allowed spending `stakedToken.GOVERNANCE_TOKEN()`

Change to `IKIP7(IStakedToken(stakedToken).GOVERNANCE_TOKEN()).safeApprove( ~`

MAJOR : `StakedToken.stakeAndActivateHeatup()` can be used to block users from redeeming

STATUS : FIXED ON 9322b504307b07c15f6dc720f08848c1c96f95a3

> contracts/stake/StakedToken.sol:172: // CC : MAJOR - should add `require(amount > 0)` to reduce risk of heatup by thrid party

stakeAndActivateHeatup() can be called with amount == 0, so it is allowed to call it without staking any GOVERNANCE_TOKEN. And this function resets the heatup timestamp to block.timestamp which will be used to DoS the withdraw functions.

Add `require(amount>0)` in stakeAndActivateHeatup. Or, just remove `behalfOf` param in the functions

MINOR : `require(uint256(newIndex) == newIndex, 'Index overflow');` can be removed

STATUS : FIXED ON 9322b504307b07c15f6dc720f08848c1c96f95a3

> contracts/stake/KlaybankDistributionManager.sol:141: require(uint256(newIndex) == newIndex, 'Index overflow'); // CC : MINOR - this cannot work as overflow check

since `newIndex` is already uint256, this require statement is meaningless

Remove this line entirely.

INFO : Duplicate statement can be merged

> contracts/stake/KlaybankDistributionManager.sol:147: assetConfig.lastUpdateTimestamp = uint40(block.timestamp); // CC : INFO - `else` statement can be removed

STATUS : FIXED ON 9322b504307b07c15f6dc720f08848c1c96f95a3

`assetConfig.lastUpdateTimestamp = uint40(block.timestamp);` is written in both inside `if` / `else`, and `else` has nothing else than this update. So `else` can be removed if `assetConfig.lastUpdateTimestamp =`

`uint40(block.timestamp);` is written outside the `if` / `else`.

INFO : StakedKbt.constructor() parameter names should be changed

> contracts/stake/StakedKbt.sol:15: IKIP7 stakedToken, // CC : INFO - name should be governanceToken

STATUS : ACKNOWLEDGED

StakedKbt.constructor() has parameter named `stakedToken` which is quite misleading since StakedToken has `stakedToken` storage variable.

Change parameter name to `governanceToken`

INFO : meaningless typecasting on StakedToken.initialize()

STATUS : ACKNOWLEDGED

> contracts/stake/StakedToken.sol:132:
> IKIP7(GOVERNANCE_TOKEN).safeApprove(address(stakedTokenAddress), type(uint256).max); // CC : INFO - stakedTokenAddress is already address type

StakedToken.initialize() function approves `GOVERNANCE_TOKEN` to `stakedTokenAddress` but this is done through typecasting the `stakedTokenAddress` to `address` type. which is awkward since it is already `address` type.