Assure DeFi®

THE VERIFICATION GOLD STANDARD

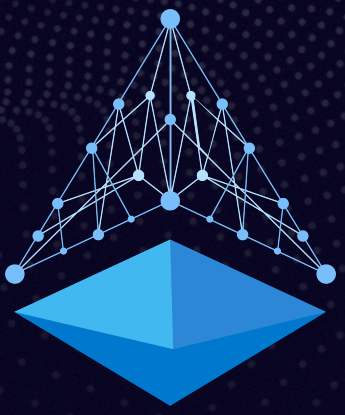
Security Assessment

EVA Farm

Date: 28/09/2024

Audit Status: PASS

Audit Edition: Advanced



ASSURE DEFI®

THE VERIFICATION GOLD STANDARD

Risk Analysis

Vulnerability summary

Classification	Description
High	High-level vulnerabilities can result in the loss of assets or manipulation of data.
Medium	Medium-level vulnerabilities can be challenging to exploit, but they still have a considerable impact on smart contract execution, such as allowing public access to critical functions.
Low	Low-level vulnerabilities are primarily associated with outdated or unused code snippets that generally do not significantly impact execution, sometimes they can be ignored.
Informational	Informational vulnerabilities, code style violations, and informational statements do not affect smart contract execution and can typically be disregarded.

Executive Summary

According to the Assure assessment, the Customer's smart contract is **Well Secured.**

Insecure	Poorly Secured	Secured	Well Secured

Scope

Target Code And Revision

For this audit, we performed research, investigation, and review of the EVA Farm contracts followed by issue reporting, along with mitigation and remediation instructions outlined in this report.

Target Code And Revision

Project	Assure
Language	Solidity
Codebase	EvaFarming.sol - [SHA256] d246c7e08b466d4e648a4a19a345be45378db120ce1519e206c35857e87d2662 EvaFarming_v2.sol [SHA256] 82b516da739ed7e777d5ce147ee94fdfae60fb5aec742b6e8d99bd3e2c7c0311
Audit Methodology	Static, Manual

Attacks made to the contract

In order to check for the security of the contract, we tested several attacks in order to make sure that the contract is secure and follows best practices.

Category	Item
Code review & Functional Review	• Compiler warnings. • Race conditions and Reentrancy. Cross-function race conditions. • Possible delays in data delivery. • Oracle calls. • Front running. • Timestamp dependence. • Integer Overflow and Underflow. • DoS with Revert. • DoS with block gas limit. • Methods execution permissions. • Economy model. • Private user data leaks. • Malicious Event log. • Scoping and Declarations. • Uninitialized storage pointers. • Arithmetic accuracy. • Design Logic. • Cross-function race conditions. • Safe Zeppelin module. • Fallback function security. • Overpowered functions / Owner privileges

AUDIT OVERVIEW

HIGH

No high severity issues were found.

MEDIUM

1. Potential Denial of Service (DOS) Due to Large Arrays [FIXED ✅]

Function: getPools()

Issue: If the pools array grows significantly, iterating over the entire array in the getPools() function may cause the transaction to run out of gas, leading to a denial of service.

In the function itself it is detailed in a comment that it should not be used onchain (@dev THIS FUNCTION IS NOT MEANT TO BE USED ONCHAIN.)*

```
address[] public pools;

/**
 * @notice Retrieves the list of active and inactive pools.
 * @dev THIS FUNCTION IS NOT MEANT TO BE USED ONCHAIN.
 * @dev This function iterates through all pools and determines which are active (last block is greater than current block)
 * and which are inactive (last block is less than or equal to current block).
 * @return active The list of active pools.
 * @return inactive The list of inactive pools.
 */
```

Recommendation: Implement pagination or batching in the getPools() function to avoid gas limitations. For example, return only a subset of pools per call, determined by start and end indices.

Fix: The vulnerability was fixed by introducing pagination

1. Lack of Input Validation for deployFarming Parameters [FIXED ✅]

Function: deployFarming()

Issue: The parameters _depositToken, _rewardToken, _rewardPerBlock, _startBlock, _bonusEndBlock, and _limitAmount are not validated in the deployFarming function. This could result in the deployment of farming contracts with erroneous or malicious configurations.

Recommendation: Validate the input parameters for logical consistency.

Fix: Input validation has been added.

2. Gas Optimization: Inefficient Array Resizing [FIXED ✅]

Function: getPools()

Issue: The getPools() function resizes the active and inactive arrays multiple times, which is an inefficient use of gas.

Recommendation: Use a single memory array and append to it instead of resizing. Alternatively, initialize the active and inactive arrays with exact sizes when possible.

Fix: The issue is solved eliminating unnecessary memory operations.

1. No Events for Critical Operations [FIXED ✅]

Function: deployFarming()

Issue: There are no events emitted in critical functions like deployFarming. This makes it difficult to track actions like deployments and ownership transfers in the blockchain history.

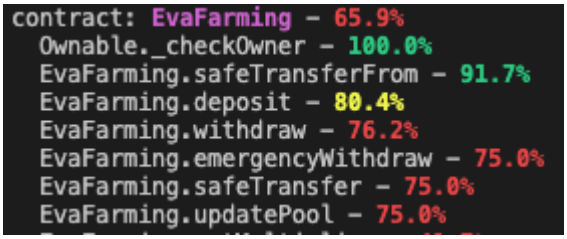
Recommendation: Add events for deployFarming and other critical actions to improve transparency and monitoring capabilities.

Fix: Events were added improving transparency and tracking.

Testing coverage

During the testing phase, custom use cases were written to cover all the logic of contracts. *Check “Annexes” to see the testing code.*

EVA Farm contract tests:



```
tests/test_eva_farming.py::test_deposit RUNNING
Transaction sent: 0x13ff6a351a7deb147a46e61aca43b90ac6a0585519e1516fb2d7bcc030c99a1c
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 0
  WETH9.constructor confirmed   Block: 1   Gas used: 476546 (3.97%)
  WETH9 deployed at: 0x3194cBDC3dbcd3E11a07892e7bA5c3394048Cc87

Transaction sent: 0x65d2c4daf2d03adae4ed9fef448f6af66e8002f9a110dafa4fec4e6040023466
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 1
  ERC20Mock.constructor confirmed   Block: 2   Gas used: 523810 (4.37%)
  ERC20Mock deployed at: 0x602C71e4DAC47a042Ee7f46E0aee17F94A3bA0B6

Transaction sent: 0x2c1f2257ebc8b19f8af081e3c6cfbe0e49ca6e180489d811987c6e569f09a815
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 2
  ERC20Mock.constructor confirmed   Block: 3   Gas used: 523810 (4.37%)
  ERC20Mock deployed at: 0xE7eD6747FaC5360f88a2EFC03E00d25789F69291

Transaction sent: 0xe7a699c27a4df1c9f4db2744211049f62ea27f7a7fd3416d4c27f4a094e5c969
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 3
  ERC20Mock.constructor confirmed   Block: 4   Gas used: 523834 (4.37%)
  ERC20Mock deployed at: 0x6951b5Bd815043E3F842c1b026b0Fa888Cc2DD85

Transaction sent: 0xe52bc72a3430e1d926a6dc459598a6bd8adc33dea4be5de426a6ea3f9eb5b60c
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 4
  UniswapV2Factory.constructor confirmed   Block: 5   Gas used: 2412742 (20.11%)
  UniswapV2Factory deployed at: 0xe0aA552A10d7EC8760Fc6c246D391E698a82dDf9

Transaction sent: 0xde66f60f31e51a5a320a7acccbd286c10067cd27e6e96a158374269d64108523
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 5
  UniswapV2Router02.constructor confirmed   Block: 6   Gas used: 3895430 (32.46%)
  UniswapV2Router02 deployed at: 0x6b4BDe1086912A6Cb24ce3dB43b3466e6c72AFd3

Transaction sent: 0xc7226bab0166f80725fd21cf70efe64d4320867d74c997a248620bc01cf520c6
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 6
  UniswapV2Factory.createPair confirmed   Block: 7   Gas used: 2020039 (16.83%)

Transaction sent: 0x70ab5231e9caaca4bc7d9073ec63b2df1cd20709af958ae675b2c353adbed19c
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 7
  EvaFarming.constructor confirmed   Block: 8   Gas used: 1401429 (11.68%)
  EvaFarming deployed at: 0xcCB53c9429d32594F404d01fbe9E65ED1DCda8D9

Transaction sent: 0x5360656245eef53f5b9f9b0b6029f2657e1275435d8665d453a757f248bc7e93
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 8
  ERC20Mock.mint confirmed   Block: 9   Gas used: 65821 (0.55%)

Transaction sent: 0x8e74a0d51c3a3737fc7f2e1b43a0d42392cdb15ad76537b5ce7afe83eca49189
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 9
  ERC20Mock.mint confirmed   Block: 10   Gas used: 65821 (0.55%)

Transaction sent: 0x77c028c8e0cab6820e428369e5cdb1da71e8603b4936edcce9d9675c57262804
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 10
  ERC20Mock.mint confirmed   Block: 11   Gas used: 65821 (0.55%)

Transaction sent: 0x73d71523fc9d4faee898fcd79454498475111e4e7f53397a7d2646ba558b12d
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 0
  ERC20Mock.approve confirmed   Block: 12   Gas used: 44271 (0.37%)

Transaction sent: 0xc8447b4da4e6d11a7a3e3ce6e1f272f6ac8ebeec579a6c785fb7635773c84723
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 1
  ERC20Mock.approve confirmed   Block: 13   Gas used: 44271 (0.37%)

Transaction sent: 0x18529a6b999ec1016f1829290db311920615643161b67ff63d05d4acd5578568
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 2
  UniswapV2Router02.addLiquidity confirmed   Block: 14   Gas used: 201146 (1.68%)

Transaction sent: 0xd200899624694aaa78ffc49d2735eb39f6a59a1f9b7413183600b39641c18f74
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 3
  Transaction confirmed   Block: 15   Gas used: 44042 (0.37%)

Transaction sent: 0x85d1260f003312e7b363c0dedd84943698a3b1f99a62d175d713b550c32528ea
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 11
  WETH9.deposit confirmed   Block: 16   Gas used: 43706 (0.36%)

Transaction sent: 0x6d736f72063200d43001ca53123a467bd3a98309f064efebbf0c4b9b564e563a
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 12
  WETH9.transfer confirmed   Block: 17   Gas used: 36904 (0.31%)

Transaction sent: 0xc3193f6b868907d4d87c6a4843a6744456c7e6be1df652886a626de12e8cc15e
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 4
  EvaFarming.deposit confirmed (reverted)   Block: 18   Gas used: 24313 (0.20%)
```

```
Transaction sent: 0x14ffd65ea35b9b922ad4cc65ba7abc5cf26a846d18d8583b42dadd4857264358
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 5
  EvaFarming.deposit confirmed (reverted)   Block: 19   Gas used: 59975 (0.50%)

Transaction sent: 0xc5079bd5926866854333694f81df85496da685e640cef248f2309933400ce871
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 6
  EvaFarming.deposit confirmed   Block: 20   Gas used: 112985 (0.94%)

Transaction sent: 0xb289f62b549ed68a0e02b9506051d8c0ac08cdeb8f0e8abb979fe5c83c18b79e
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 7
  EvaFarming.deposit confirmed   Block: 21   Gas used: 77728 (0.65%)

tests/test_eva_farming.py::test_deposit PASSED
tests/test_eva_farming.py::test_withdraw RUNNING
Transaction sent: 0x1bd7ab17a7743b4fcb72f0d6072e511803683897d75c70930a3e19d2ffda25f2
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 13
  WETH9.constructor confirmed   Block: 22   Gas used: 476546 (3.97%)
  WETH9 deployed at: 0xe692Cf21B12e0B2717C4bF647F9768Fa58861c8b

Transaction sent: 0x26f3fe796d6aef62a1111eb3a1370ae2a1f802f3e4761d57322a26ecdf78061l2
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 14
  ERC20Mock.constructor confirmed   Block: 23   Gas used: 523810 (4.37%)
  ERC20Mock deployed at: 0xe65A7a341978d59d40d30FC23F5014FACB4f575A

Transaction sent: 0xa474b58ada8e6c9ab388f1c877fda44a8194203469d8b456e94a500dabc214d9
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 15
  ERC20Mock.constructor confirmed   Block: 24   Gas used: 523810 (4.37%)
  ERC20Mock deployed at: 0x30375B532345B01cB8c2AD12541b09E9Aa53A93d

Transaction sent: 0xaec91638073195846bdbff2063aa160675dda0a2053ade9a5c278d7bf15c9afc
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 16
  ERC20Mock.constructor confirmed   Block: 25   Gas used: 523834 (4.37%)
  ERC20Mock deployed at: 0x26f15335BB1C6a4C0B660eDd694a0555A9F1cce3

Transaction sent: 0x6dc2a8c1097c7934aa95ee9059c837660babadd07c65ef1802b73dd7e668561c
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 17
  UniswapV2Factory.constructor confirmed   Block: 26   Gas used: 2412742 (20.11%)
  UniswapV2Factory deployed at: 0xFbD588c72B438faD4Cf7cD879c8F730Faa213Da0

Transaction sent: 0x58d508c61d2229d6ddfbb6214a963a32c651e6f2f07c486e59e4b0884a433b99
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 18
  UniswapV2Router02.constructor confirmed   Block: 27   Gas used: 3895430 (32.46%)
  UniswapV2Router02 deployed at: 0xed00238F9A0F7b4d93842033cdF56cCB32C781c2

Transaction sent: 0x77454d8b8ed6b39d9082c5f42dbc8b620fe27be0b7293b6b02d128a5eb51c992
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 19
  UniswapV2Factory.createPair confirmed   Block: 28   Gas used: 2020041 (16.83%)

Transaction sent: 0x8c97007ca96cde6ca228220029a8a2f0642a7b7d1196a64747ceb6d1b37d2aa3
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 20
  EvaFarming.constructor confirmed   Block: 29   Gas used: 1401417 (11.68%)
  EvaFarming deployed at: 0xdCF93F11ef216cEC9C07fd31dD801c9b2b39Afb4

Transaction sent: 0x54fe55126459f2bbf2b79e736e65d237d8b6dcfcc482a436cdd5f50c367e2767
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 21
  ERC20Mock.mint confirmed   Block: 30   Gas used: 65821 (0.55%)

Transaction sent: 0x36f888acc19c39394037d8e042fcdc9f9eb319ae1f3829683bc11eea8f9f8e48
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 22
  ERC20Mock.mint confirmed   Block: 31   Gas used: 65821 (0.55%)

Transaction sent: 0xa406c6d71522893939dcda7b4412f001c7a2314e13ac3a9eb807bcd5824bac33
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 8
  ERC20Mock.approve confirmed   Block: 32   Gas used: 44259 (0.37%)

Transaction sent: 0x2a1b88e3218416e0d2cce1f587ec780f13280c6604761da6d61710f88bc14f05
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 9
  ERC20Mock.approve confirmed   Block: 33   Gas used: 44259 (0.37%)

Transaction sent: 0x8d355d084d542faa686fdc290da46368cdea63dfcf4e77df36b5171b934ac6f8
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 10
  UniswapV2Router02.addLiquidity confirmed   Block: 34   Gas used: 201198 (1.68%)

Transaction sent: 0x727565dcb5561f2f0ea678726fcb086255245fc389870eaa6b5e74fde9967338
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 11
  Transaction confirmed   Block: 35   Gas used: 44042 (0.37%)

Transaction sent: 0x561886ld8f3a7c832630b882bf0c9822066811dc14d5c104464fae1530ca3bd9
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 23
  WETH9.deposit confirmed   Block: 36   Gas used: 43706 (0.36%)
```

```
Transaction sent: 0x05d46fff79d427d0797c5b771b310a8f36882f2f8f8d040fb6670ff0cd26e0e0
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 24
  WETH9.transfer confirmed   Block: 37   Gas used: 36904 (0.31%)

Transaction sent: 0x9dbfe87203006d34acd0c94a952ffa159adb915057901659df952b9b473e04d2
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 12
  EvaFarming.deposit confirmed   Block: 38   Gas used: 112985 (0.94%)

Transaction sent: 0xd4e88785ca0d0aclef2d0fa761c36f3f3da871a5c8c8349d2b8c32b40f377bbc
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 13
  EvaFarming.withdraw confirmed (reverted)   Block: 39   Gas used: 23389 (0.19%)

Transaction sent: 0xddaaa1967008d9a5ffc674d1ae7f95f89980d42ddb65eba7c7b2d16a4e2c351b
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 14
  EvaFarming.withdraw confirmed   Block: 40   Gas used: 68736 (0.57%)

Transaction sent: 0x7a98afd784c78a05ffb6eebba90ebb945001b7b948828f964508eb0bd8082db2
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 15
  EvaFarming.withdraw confirmed (reverted)   Block: 41   Gas used: 23389 (0.19%)

Transaction sent: 0xf4f8b29c4c2491a3f2e147eeaa0a6cc440578429552367ba702f36a62a388e22
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 16
  EvaFarming.pause confirmed (reverted)   Block: 42   Gas used: 22178 (0.18%)

Transaction sent: 0x6f18507e67a29812ba6e858e3a2135631e8ee41458b42fae030be2a578ac0a93
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 25
  EvaFarming.pause confirmed   Block: 43   Gas used: 29936 (0.25%)

Transaction sent: 0x59945d1839d1e79c1af589d3030171ea3d08f96cc372e68574bf474c6d8dbcce
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 17
  EvaFarming.emergencyWithdraw confirmed (reverted)   Block: 44   Gas used: 22129 (0.18%)

Transaction sent: 0x00d5e9ed2e8531da2204ed60a2e4ee7777a85e30cee8b963034608ff5f99e15d
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 26
  EvaFarming.unpause confirmed   Block: 45   Gas used: 29974 (0.25%)

Transaction sent: 0x3cc4a28f6ce4cdbe8003e4e65c8d7b14bf2577b83bc080da4adbaef9e2686abe
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 18
  EvaFarming.emergencyWithdraw confirmed   Block: 46   Gas used: 23973 (0.20%)

Transaction sent: 0x15414ef23fd443e10581b000dfcde5ba7bfa0c7afc5f64d7d59a93b4585c7a13
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 19
  EvaFarming.emergencyRewardsWithdraw confirmed (reverted)   Block: 47   Gas used: 22199 (0.18%)

Transaction sent: 0x6cac752c617d021c5ecd1a1ff2d02c947a09b49c0ea4a018e1611f063aa542a3
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 27
  ERC20Mock.mint confirmed   Block: 48   Gas used: 65821 (0.55%)

Transaction sent: 0x0f2e7caf0298fbeaec9bb52590623249ce792d1e53d44a9823499b49a9de6c5b
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 28
  EvaFarming.emergencyRewardsWithdraw confirmed   Block: 49   Gas used: 43584 (0.36%)

Transaction sent: 0x60d1c2f7697ea27ba8541a4ae168d7b9e136923fc347d63cbec3a2dc17df25c2
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 29
  EvaFarmingFactory.constructor confirmed   Block: 50   Gas used: 1801636 (15.01%)
  EvaFarmingFactory deployed at: 0x42E8D004c84E6B5Bad559D3b5CE7947AADb9E0bc

Transaction sent: 0x67b75457db3a21ee697423fb782fa3019e1439d6a6249124ed74a0d803c000b8
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 20
  EvaFarmingFactory.deployFarming confirmed (reverted)   Block: 51   Gas used: 23911 (0.20%)

Transaction sent: 0x818f1081c0d28b6dcad82691fab4306e94dff5d0653dda4bd02878afaa882add
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 30
  EvaFarmingFactory.deployFarming confirmed   Block: 52   Gas used: 1358693 (11.32%)

tests/test_eva_farming.py::test_withdraw PASSED
```

Annexes

Testing code:

Testing_EVA Farm:

```
from brownie import (
    reverts, accounts, UniswapV2Pair
)
from brownie.network.contract import Contract


from scripts.helpful_scripts import (
    ZERO_ADDRESS,
    DAY_TIMESTAMP,
    get_account,
    get_timestamp,
    increase_timestamp,
    get_chain_number,
)


from scripts.deploy import (
    deploy_weth,
    deploy_erc,
    deploy_factory,
    deploy_router,
    deploy_eva_farming,
    deploy_eva_farming_factory,
)


def test_deposit(only_local):
```

```
    # Arrange
    owner = get_account(0)
    other = get_account(1)
    extra = get_account(2)

    weth = deploy_weth(owner)
    usd = deploy_erc(owner, "USD", "USD")
    eva = deploy_erc(owner, "Eva", "EVA")
    reward = deploy_erc(owner, "Token", "TKN")
    factory = deploy_factory(owner, owner)
    router = deploy_router(owner, factory.address, weth.address)

    tx = factory.createPair(eva.address, usd.address)
    pair_addr = tx.events['PairCreated'][0]['pair']

    farming = deploy_eva_farming(owner, pair_addr, reward.address)

    # Mint some tokens
    usd.mint(other, 10e18)
    eva.mint(other, 10e18)
    reward.mint(farming.address, 10e18)
    # Add allowance to router
    eva.approve(router.address, 2e18, {"from": other})
    usd.approve(router.address, 2e18, {"from": other})
    # Add liquidity
    router.addLiquidity(usd.address, eva.address, 2e18, 2e18, 1, 1, other,
get_timestamp(1), {"from": other})
```

```
    pair = Contract.from_abi("UniswapV2Pair", pair_addr, UniswapV2Pair.abi)
    pair.approve(farming.address, 10e18, {"from": other})
    #print(pair.decimals())
    #print(pair.balanceOf(other))

    weth.deposit({"from": owner, "value": 5e18})
    weth.transfer(farming.address, 5e18, {"from": owner})

    with reverts("EVAFarming_ExceedTheTop: "):
        farming.deposit(6e18, {"from": other})
    with reverts("EVAFarming_ErrorWhileTransferring: "):
        farming.deposit(3e18, {"from": other})

    tx = farming.deposit(0.5e18, {"from": other})
    assert tx.events['Transfer'][0]['from'] == other
    assert tx.events['Transfer'][0]['to'] == farming.address
    assert tx.events['Transfer'][0]['value'] == 0.5e18

    tx = farming.deposit(0.5e18, {"from": other})
    assert tx.events['Transfer'][0]['from'] == other
    assert tx.events['Transfer'][0]['to'] == farming.address
    assert tx.events['Transfer'][0]['value'] == 0.5e18

def test_withdraw(only_local):
    # Arrange
    owner = get_account(0)
    other = get_account(1)
    extra = get_account(2)
```

```
    weth = deploy_weth(owner)
    usd = deploy_erc(owner, "USD", "USD")
    eva = deploy_erc(owner, "Eva", "EVA")
    reward = deploy_erc(owner, "Token", "TKN")
    factory = deploy_factory(owner, owner)
    router = deploy_router(owner, factory.address, weth.address)

    tx = factory.createPair(eva.address, usd.address)
    pair_addr = tx.events['PairCreated'][0]['pair']

    farming = deploy_eva_farming(owner, pair_addr, reward.address)

    # Mint some tokens
    usd.mint(other, 10e18)
    eva.mint(other, 10e18)
    # Add allowance to router
    eva.approve(router.address, 2e18, {"from": other})
    usd.approve(router.address, 2e18, {"from": other})
    # Add liquidity
    router.addLiquidity(usd.address, eva.address, 2e18, 2e18, 1, 1, other,
get_timestamp(1), {"from": other})

    pair = Contract.from_abi("UniswapV2Pair", pair_addr, UniswapV2Pair.abi)
    pair.approve(farming.address, 10e18, {"from": other})
    weth.deposit({"from": owner, "value": 5e18})
    weth.transfer(farming.address, 5e18, {"from": owner})
```

```
    farming.deposit(1e18, {"from": other})

    # Assert

    with reverts("EVAFarming_InvalidWithdrawAmount: "):

        farming.withdraw(2e18, {"from": other})

    tx = farming.withdraw(0.5e18, {"from": other})

    assert tx.events['Transfer'][0]['from'] == farming.address

    assert tx.events['Transfer'][0]['to'] == other

    assert tx.events['Transfer'][0]['value'] == 0.5e18


    with reverts("EVAFarming_InvalidWithdrawAmount: "):

        farming.withdraw(1e18, {"from": other})

    with reverts():

        farming.pause({"from": other})

    farming.pause({"from": owner})

    with reverts("EnforcedPause: "):

        farming.emergencyWithdraw({"from": other})

    farming.unpause({"from": owner})

    tx = farming.emergencyWithdraw({"from": other})

    assert tx.events['Transfer'][0]['from'] == farming.address

    assert tx.events['Transfer'][0]['to'] == other

    assert tx.events['Transfer'][0]['value'] == 0.5e18


    with reverts():

        farming.emergencyRewardsWithdraw({"from": other})

    reward.mint(farming.address, 1e18)

    tx = farming.emergencyRewardsWithdraw({"from": owner})

    assert tx.events['Transfer'][0]['from'] == farming.address

    assert tx.events['Transfer'][0]['to'] == owner
```

```
    assert tx.events['Transfer'][0]['value'] == 1e18


    farm_factory = deploy_eva_farming_factory(owner)
    with reverts():
        farm_factory.deployFarming(
            pair_addr, reward.address,
            1e16, 0, 0, 5e18,
            {"from": other})
    tx = farm_factory.deployFarming(
            pair_addr, reward.address,
            1e16, 0, 0, 5e18,
            {"from": owner})
    assert tx.events['OwnershipTransferred'][1]['previousOwner'] == farm_factory.address
    assert tx.events['OwnershipTransferred'][1]['newOwner'] == owner
```

Technical Findings Summary

Findings

Vulnerability Level	Total	Pending	Not Apply	Acknowledged	Partially Fixed	Fixed
High	0					
Medium	1					1
Low	2					2
Informational	1					1

Assessment Results

Score Results

Review	Score
Global Score	**95/100**
Assure KYC	https://www.assuredefi.com/projects/eva-intelligence/
Audit Score	95/100

The Following Score System Has been Added to this page to help understand the value of the audit, the maximum score is 100, however to attain that value the project must pass and provide all the data needed for the assessment. Our Passing Score has been changed to 84 Points for a higher standard, if a project does not attain 85% is an automatic failure. Read our notes and final assessment below. The Global Score is a combination of the evaluations obtained between having or not having KYC and the type of contract audited together with its manual audit.

Audit PASS

Following our comprehensive security audit of the token contract for the EVA Farm project, we inform you that the project has met the necessary security standards. We recommend reviewing the reported issues and fixing them to the extent possible.
V2: All issues presented in the report were resolved.

Disclaimer

Assure Defi has conducted an independent security assessment to verify the integrity of and highlight any vulnerabilities or errors, intentional or unintentional, that may be present in the reviewed code for the scope of this assessment. This report does not constitute agreement, acceptance, or advocating for the Project, and users relying on this report should not consider this as having any merit for financial adEVA Farm in any shape, form, or nature. The contracts audited do not account for any economic developments that the Project in question may pursue, and the veracity of the findings thus presented in this report relate solely to the proficiency, competence, aptitude, and discretion of our independent auditors, who make no guarantees nor assurance that the contracts are entirely free of exploits, bugs, vulnerabilities or deprecation of technologies.

All information provided in this report does not constitute financial or investment adEVA Farm, nor should it be used to signal that any person reading this report should invest their funds without sufficient individual due diligence, regardless of the findings presented. Information is provided ‘as is, and Assure Defi is under no covenant to audit completeness, accuracy, or solidity of the contracts. In no event will Assure Defi or its partners, employees, agents, or parties related to the provision of this audit report be liable to any parties for, or lack thereof, decisions or actions with regards to the information provided in this audit report.

The assessment serEVA Farms provided by Assure Defi are subject to dependencies and are under continuing development. You agree that your access or use, including but not limited to any serEVA Farms, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies with high levels of technical risk and uncertainty. The assessment reports could include false positives, negatives, and unpredictable results. The serEVA Farms may access, and depend upon, multiple layers of third parties.