Assure DeFi®

THE VERIFICATION GOLD STANDARD

VERIFIED BY ASSURE DEFI ®

INTEGRITY ✦ TRUST ✦ CREDIBILITY

Security Assessment

AITAX

Date: 28/04/2024

Audit Status: FAILED

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🟡 Low	Low-level vulnerabilities are primarily associated with outdated or unused code snippets that generally do not significantly impact execution, sometimes they can be ignored.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 **Insecure.**

Insecure	**Poorly Secured**	**Secured**	**Well Secured**

Scope

Target Code And Revision

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

Target Code And Revision

Project	Assure
Language	Solidity
Codebase	AITaxEthSplitter.sol - [SHA256] *412980afde1b84f28cc1f112f4d96a1b98f797743e560a077441d06b2b92b44a* AITaxStaking.sol - [SHA256] *6c5e50aa17d0673f90b4f52c86dd3d3ca00ff97766e085e076e879368c1415fa*
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

1. Ensuring Code Compilation by Managing Unused Variables

Contract: AITaxEthSplitter

Function: Not applicable

Issue: The contract fails to compile due to the unused public variable distributionPercs, which lacks any associated import.

Mitigation: Either remove the unused variable or ensure its related import is correctly set.

2. Consistent Variable Naming in distributeETH Function

Contract: AITaxEthSplitter

Function: distributeETH()

Issue: The contract does not compile because the variable buyBackAmount is inconsistently named in conditions as buybackAmount.

Mitigation: Harmonize the variable name throughout the function to ensure consistent naming.

◆ ◆ ◆ MEDIUM

1. Validating Transaction Success in distributeETH Function

Contract: AITaxEthSplitter

Function: distributeETH()

Issue: The function does not check the boolean success after executing payable().call, potentially missing errors.

Mitigation: Introduce a require() statement to verify success after the transaction, ensuring error handling.

2. Managing High Stake Percentages in distributeETH Function

Contract: AITaxEthSplitter

Function: distributeETH()

Issue: If percentStaked is excessively high, it causes the buybackamount calculation to fail.

Mitigation: Validate percentStaked prior to executing the token transfer to prevent failures.

3. Ensuring Successful ETH Withdrawal in withdrawStuckETH Function

Contract: AITaxEthSplitter

Function: withdrawStuckETH()

Issue: Similar to distributeETH(), this function does not verify the success boolean after payable().call, risking unhandled errors.

Mitigation: Implement a require() check to confirm success, ensuring the robustness of the withdrawal process.

LOW

1. Avoiding Naming Conflicts in Constructor Arguments

Contract: AITaxEthSplitter

Function: constructor()

Issue: The constructor uses function argument names that conflict with contract names, which can lead to confusion.

Mitigation: Rename aiTaxToken to aiTaxToken_ to avoid naming conflicts and enhance code clarity.

INFORMATIONAL

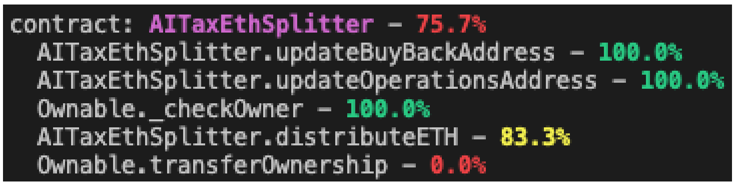
No informational severity issues were found.

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.*

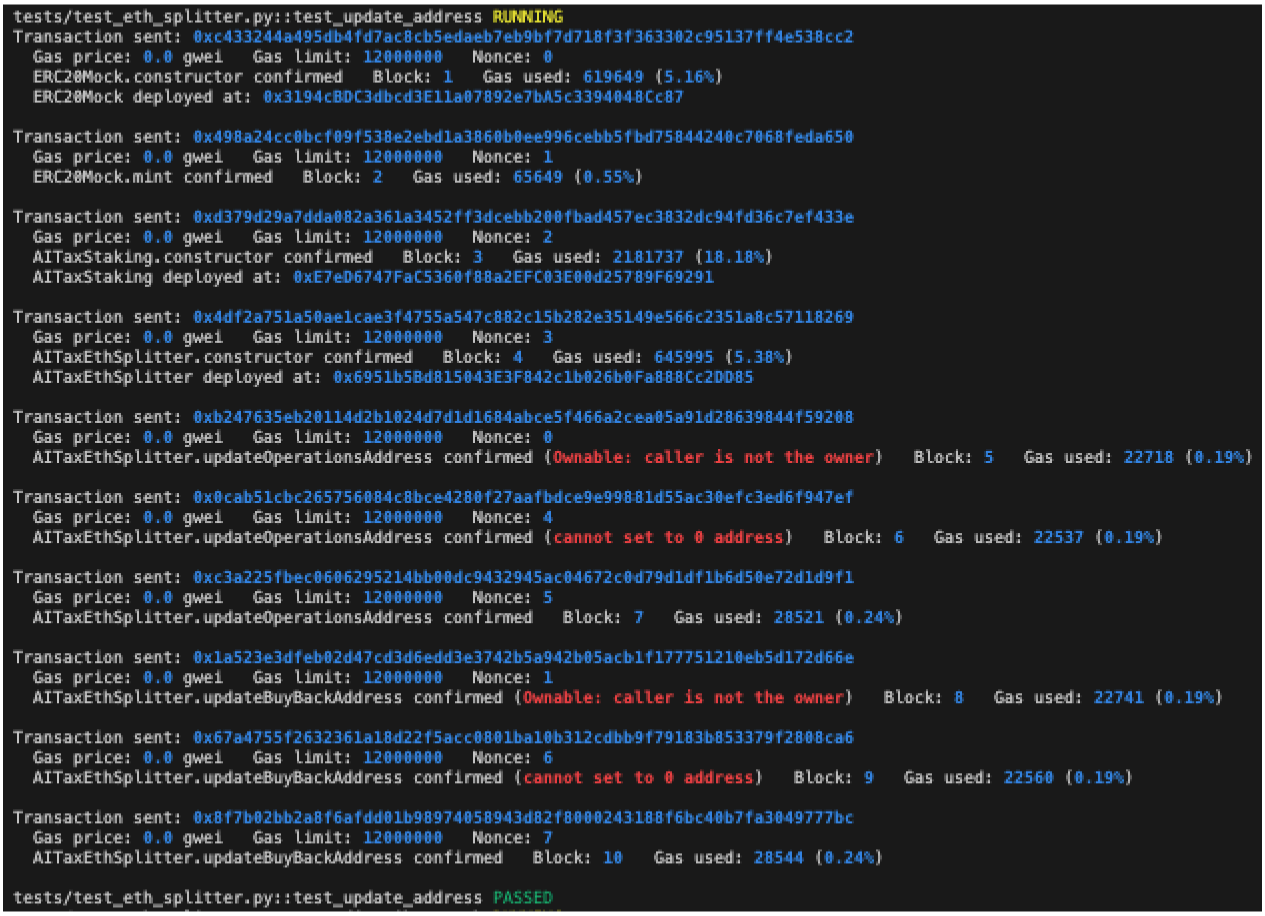
AITAX test:

Coverages:

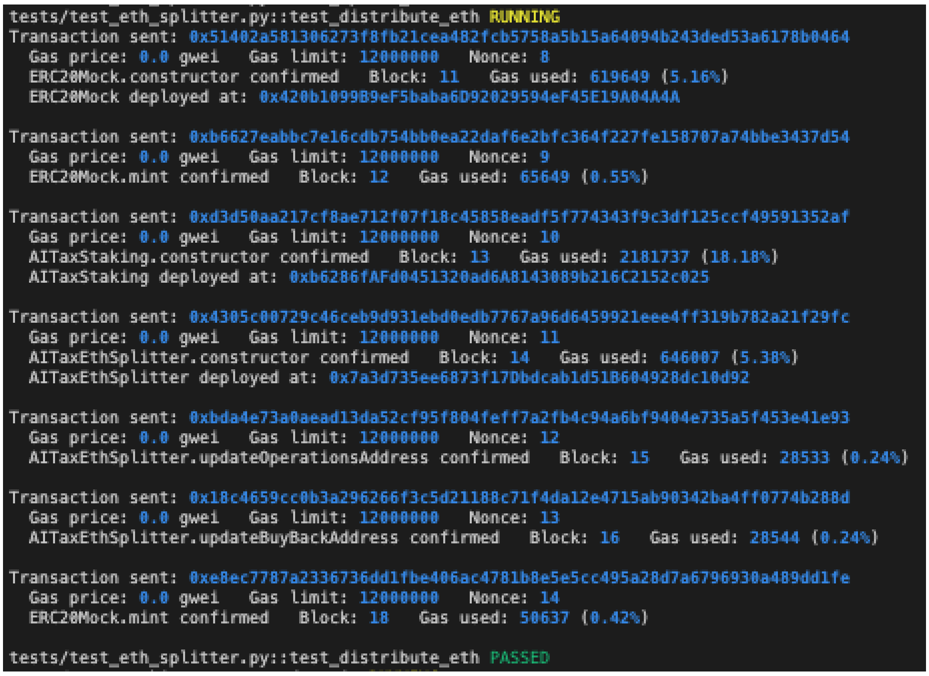


```
contract: AITaxEthSplitter - 75.7%
  AITaxEthSplitter.updateBuyBackAddress - 100.0%
  AITaxEthSplitter.updateOperationsAddress - 100.0%
  Ownable._checkOwner - 100.0%
  AITaxEthSplitter.distributeETH - 83.3%
  Ownable.transferOwnership - 0.0%
```

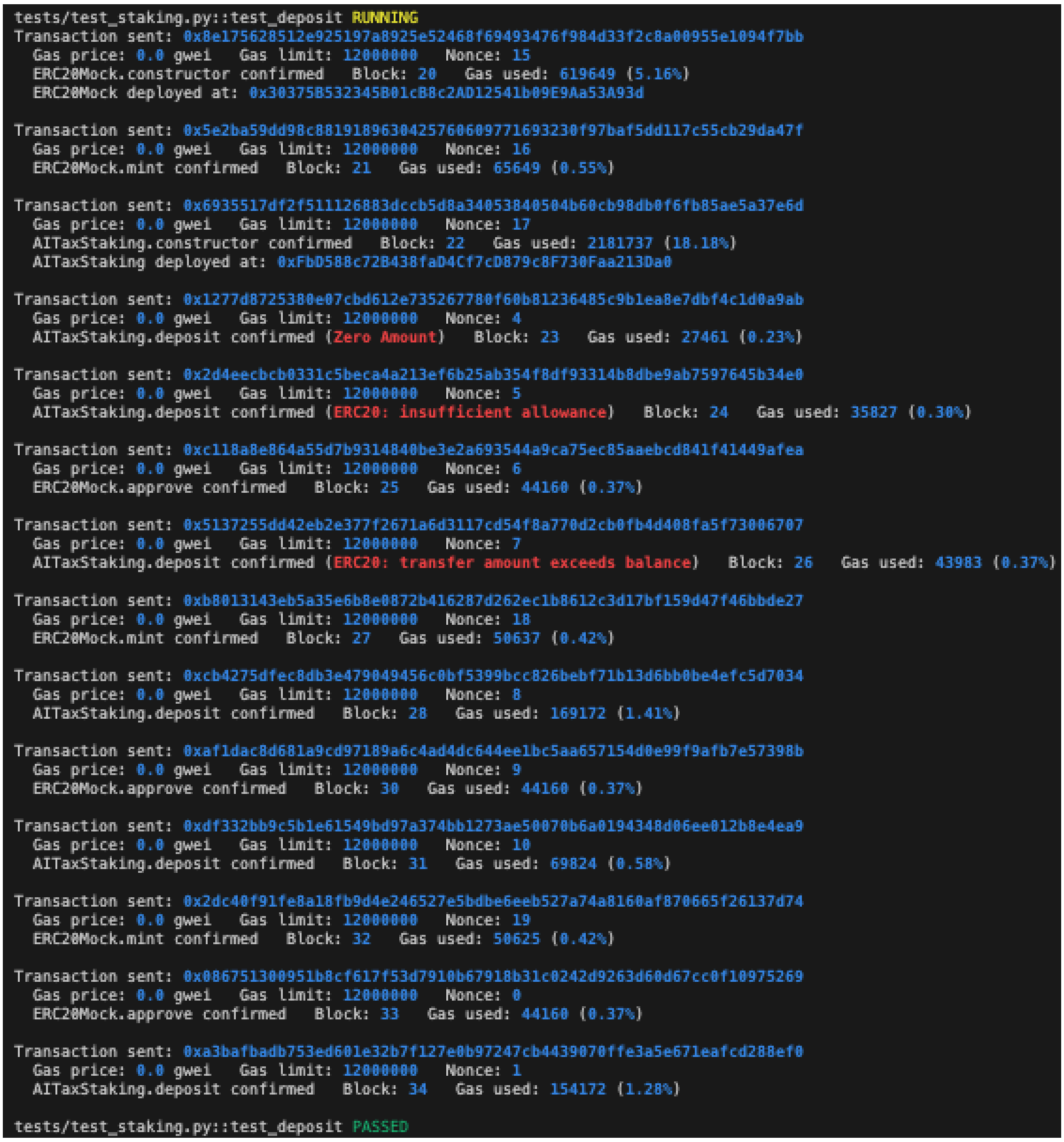
Testing AITAX contracts:



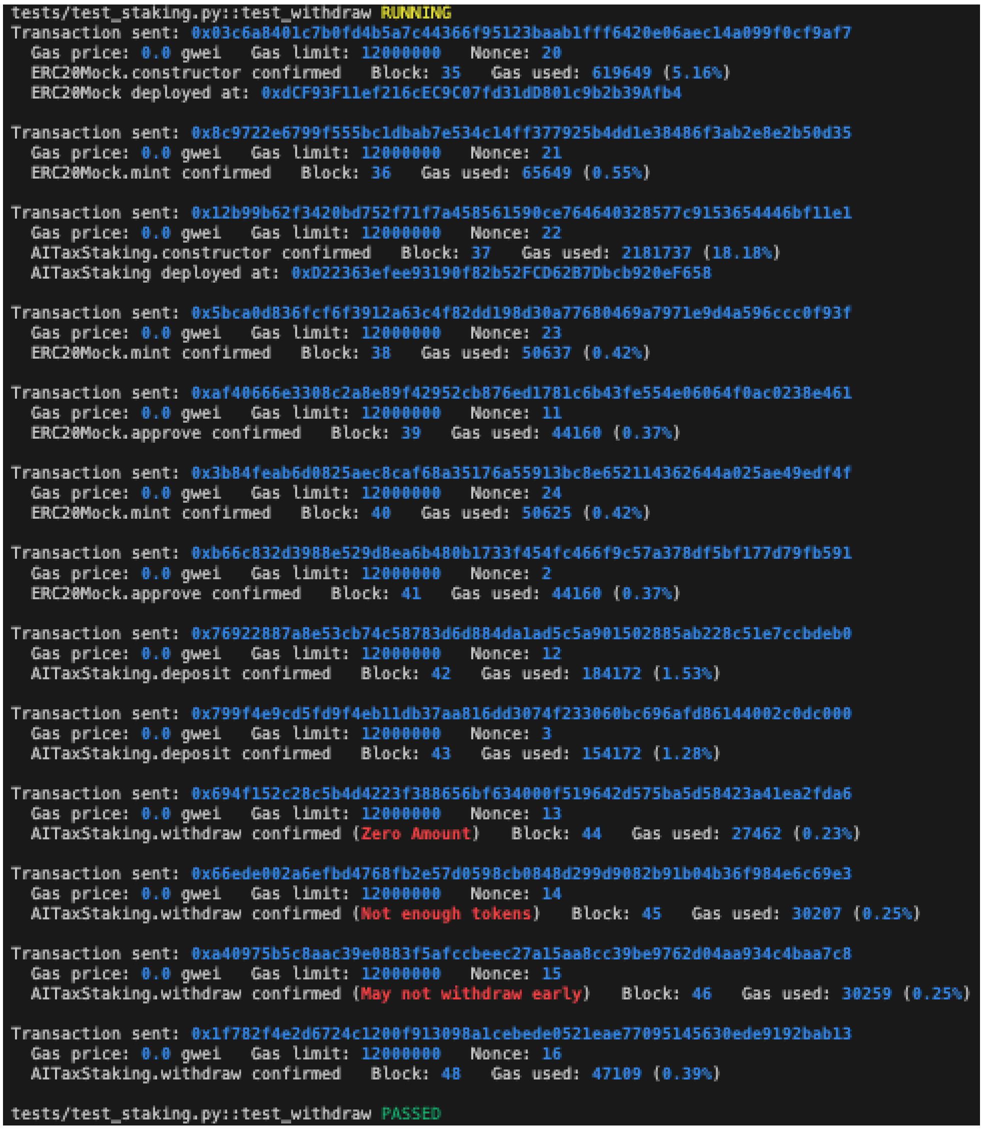
```
tests/test_eth_splitter.py::test_update_address RUNNING
Transaction sent: 0xc433244a495db4fd7ac8cb5edaeb7eb9bf7d718f3f363302c95137ff4e538cc2
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 0
  ERC20Mock.constructor confirmed   Block: 1   Gas used: 619649 (5.16%)
  ERC20Mock deployed at: 0x3194cBDC3dbcd3E11a07892e7bA5c3394048Cc87

Transaction sent: 0x498a24cc0bcf09f538e2ebd1a3860b0ee996cebb5fbd75844240c7068feda650
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 1
  ERC20Mock.mint confirmed   Block: 2   Gas used: 65649 (0.55%)

Transaction sent: 0xd379d29a7dda082a361a3452ff3dcebb200fbad457ec3832dc94fd36c7ef433e
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 2
  AITaxStaking.constructor confirmed   Block: 3   Gas used: 2181737 (18.18%)
  AITaxStaking deployed at: 0xE7eD6747FaC5360f88a2EFC03E00d25789F69291

Transaction sent: 0x4df2a751a50ae1cae3f4755a547c882c15b282e35149e566c2351a8c57118269
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 3
  AITaxEthSplitter.constructor confirmed   Block: 4   Gas used: 645995 (5.38%)
  AITaxEthSplitter deployed at: 0x6951b5Bd815043E3F842c1b026b0Fa888Cc2DD85

Transaction sent: 0xb247635eb20114d2b1024d7d1d1684abce5f466a2cea05a91d28639844f59208
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 0
  AITaxEthSplitter.updateOperationsAddress confirmed (Ownable: caller is not the owner)   Block: 5   Gas used: 22718 (0.19%)

Transaction sent: 0x0cab51cbc265756084c8bce4280f27aafbdce9e99881d55ac30efc3ed6f947ef
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 4
  AITaxEthSplitter.updateOperationsAddress confirmed (cannot set to 0 address)   Block: 6   Gas used: 22537 (0.19%)

Transaction sent: 0xc3a225fbec0606295214bb00dc9432945ac04672c0d79d1df1b6d50e72d1d9f1
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 5
  AITaxEthSplitter.updateOperationsAddress confirmed   Block: 7   Gas used: 28521 (0.24%)

Transaction sent: 0x1a523e3dfeb02d47cd3d6edd3e3742b5a942b05acb1f177751210eb5d172d66e
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 1
  AITaxEthSplitter.updateBuyBackAddress confirmed (Ownable: caller is not the owner)   Block: 8   Gas used: 22741 (0.19%)

Transaction sent: 0x67a4755f2632361a18d22f5acc0801ba10b312cdbb9f79183b853379f2808ca6
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 6
  AITaxEthSplitter.updateBuyBackAddress confirmed (cannot set to 0 address)   Block: 9   Gas used: 22560 (0.19%)

Transaction sent: 0x8f7b02bb2a8f6afdd01b98974058943d82f8000243188f6bc40b7fa3049777bc
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 7
  AITaxEthSplitter.updateBuyBackAddress confirmed   Block: 10   Gas used: 28544 (0.24%)

tests/test_eth_splitter.py::test_update_address PASSED
```



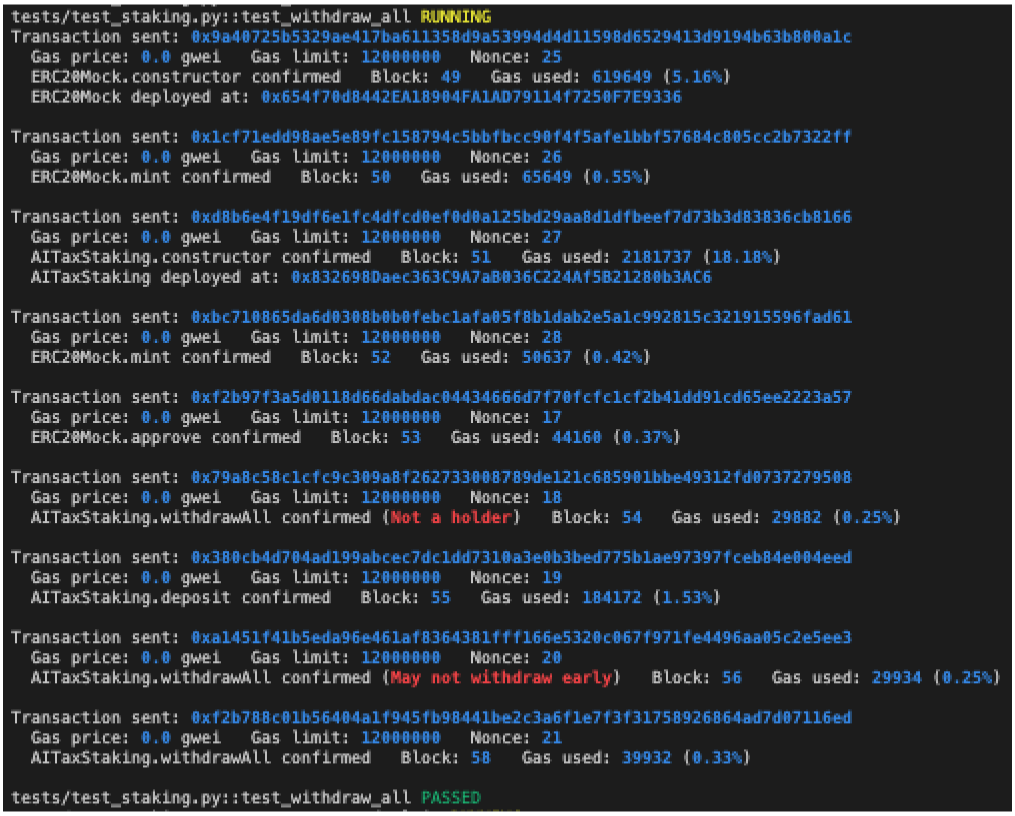
```
tests/test_eth_splitter.py::test_distribute_eth RUNNING
Transaction sent: 0x51402a581306273f8fb21cea482fcb5758a5b15a64094b243ded53a6178b0464
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 8
  ERC20Mock.constructor confirmed   Block: 11   Gas used: 619649 (5.16%)
  ERC20Mock deployed at: 0x420b1099B9eF5baba6D92029594eF45E19A04A4A

Transaction sent: 0xb6627eabbc7e16cdb754bb0ea22daf6e2bfc364f227fe158707a74bbe3437d54
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 9
  ERC20Mock.mint confirmed   Block: 12   Gas used: 65649 (0.55%)

Transaction sent: 0xd3d50aa217cf8ae712f07f18c45858eadf5f774343f9c3df125ccf49591352af
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 10
  AITaxStaking.constructor confirmed   Block: 13   Gas used: 2181737 (18.18%)
  AITaxStaking deployed at: 0xb6286fAFd0451320ad6A8143089b216C2152c025

Transaction sent: 0x4305c00729c46ceb9d931ebd0edb7767a96d6459921eee4ff319b782a21f29fc
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 11
  AITaxEthSplitter.constructor confirmed   Block: 14   Gas used: 646007 (5.38%)
  AITaxEthSplitter deployed at: 0x7a3d735ee6873f17Dbdcab1d51B604928dc10d92

Transaction sent: 0xbda4e73a0aead13da52cf95f804feff7a2fb4c94a6bf9404e735a5f453e41e93
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 12
  AITaxEthSplitter.updateOperationsAddress confirmed   Block: 15   Gas used: 28533 (0.24%)

Transaction sent: 0x18c4659cc0b3a296266f3c5d21188c71f4da12e4715ab90342ba4ff0774b288d
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 13
  AITaxEthSplitter.updateBuyBackAddress confirmed   Block: 16   Gas used: 28544 (0.24%)

Transaction sent: 0xe8ec7787a2336736dd1fbe406ac4781b8e5e5cc495a28d7a6796930a489dd1fe
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 14
  ERC20Mock.mint confirmed   Block: 18   Gas used: 50637 (0.42%)

tests/test_eth_splitter.py::test_distribute_eth PASSED
```



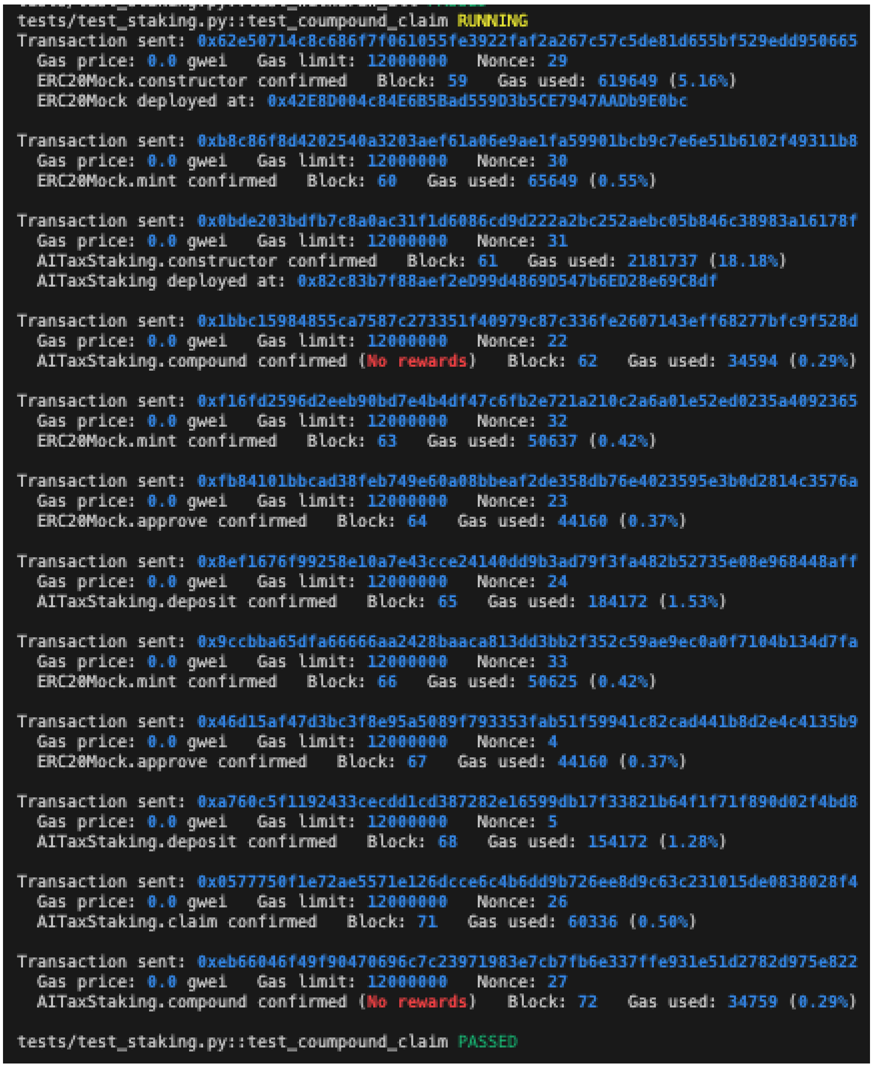
```
tests/test_staking.py::test_deposit RUNNING
Transaction sent: 0x8e175628512e925197a8925e52468f69493476f984d33f2c8a00955e1094f7bb
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 15
  ERC20Mock.constructor confirmed   Block: 20   Gas used: 619649 (5.16%)
  ERC20Mock deployed at: 0x30375B532345B01cB8c2AD12541b09E9Aa53A93d

Transaction sent: 0x5e2ba59dd98c8819189630425760609771693230f97baf5dd117c55cb29da47f
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 16
  ERC20Mock.mint confirmed   Block: 21   Gas used: 65649 (0.55%)

Transaction sent: 0x6935517df2f511126883dccb5d8a34053840504b60cb98db0f6fb85ae5a37e6d
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 17
  AITaxStaking.constructor confirmed   Block: 22   Gas used: 2181737 (18.18%)
  AITaxStaking deployed at: 0xFbD588c72B438faD4Cf7cD879c8F730Faa213Da0

Transaction sent: 0x1277d8725380e07cbd612e735267780f60b81236485c9b1ea8e7dbf4c1d0a9ab
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 4
  AITaxStaking.deposit confirmed (Zero Amount)   Block: 23   Gas used: 27461 (0.23%)

Transaction sent: 0x2d4eecbcb0331c5beca4a213ef6b25ab354f8df93314b8dbe9ab7597645b34e0
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 5
  AITaxStaking.deposit confirmed (ERC20: insufficient allowance)   Block: 24   Gas used: 35827 (0.30%)

Transaction sent: 0xc118a8e864a55d7b9314840be3e2a693544a9ca75ec85aaebcd841f41449afea
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 6
  ERC20Mock.approve confirmed   Block: 25   Gas used: 44160 (0.37%)

Transaction sent: 0x5137255dd42eb2e377f2671a6d3117cd54f8a770d2cb0fb4d408fa5f73006707
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 7
  AITaxStaking.deposit confirmed (ERC20: transfer amount exceeds balance)   Block: 26   Gas used: 43983 (0.37%)

Transaction sent: 0xb8013143eb5a35e6b8e0872b416287d262ec1b8612c3d17bf159d47f46bbde27
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 18
  ERC20Mock.mint confirmed   Block: 27   Gas used: 50637 (0.42%)

Transaction sent: 0xcb4275dfec8db3e479049456c0bf5399bcc826bebf71b13d6bb0be4efc5d7034
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 8
  AITaxStaking.deposit confirmed   Block: 28   Gas used: 169172 (1.41%)

Transaction sent: 0xaf1dac8d681a9cd97189a6c4ad4dc644ee1bc5aa657154d0e99f9afb7e57398b
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 9
  ERC20Mock.approve confirmed   Block: 30   Gas used: 44160 (0.37%)

Transaction sent: 0xdf332bb9c5b1e61549bd97a374bb1273ae50070b6a0194348d06ee012b8e4ea9
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 10
  AITaxStaking.deposit confirmed   Block: 31   Gas used: 69824 (0.58%)

Transaction sent: 0x2dc40f91fe8a18fb9d4e246527e5bdbe6eeb527a74a8160af870665f26137d74
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 19
  ERC20Mock.mint confirmed   Block: 32   Gas used: 50625 (0.42%)

Transaction sent: 0x086751300951b8cf617f53d7910b67918b31c0242d9263d60d67cc0f10975269
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 0
  ERC20Mock.approve confirmed   Block: 33   Gas used: 44160 (0.37%)

Transaction sent: 0xa3bafbadb753ed601e32b7f127e0b97247cb4439070ffe3a5e671eafcd288ef0
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 1
  AITaxStaking.deposit confirmed   Block: 34   Gas used: 154172 (1.28%)

tests/test_staking.py::test_deposit PASSED
```



```
tests/test_staking.py::test_withdraw RUNNING
Transaction sent: 0x03c6a8401c7b0fd4b5a7c44366f95123baab1fff6420e06aec14a099f0cf9af7
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 20
  ERC20Mock.constructor confirmed   Block: 35   Gas used: 619649 (5.16%)
  ERC20Mock deployed at: 0xdCF93F11ef216cEC9C07fd31dD801c9b2b39Afb4

Transaction sent: 0x8c9722e6799f555bc1dbab7e534c14ff377925b4dd1e38486f3ab2e8e2b50d35
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 21
  ERC20Mock.mint confirmed   Block: 36   Gas used: 65649 (0.55%)

Transaction sent: 0x12b99b62f3420bd752f71f7a458561590ce764640328577c9153654446bf11e1
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 22
  AITaxStaking.constructor confirmed   Block: 37   Gas used: 2181737 (18.18%)
  AITaxStaking deployed at: 0xD22363efee93190f82b52FCD62B7Dbcb920eF658

Transaction sent: 0x5bca0d836fcf6f3912a63c4f82dd198d30a77680469a7971e9d4a596ccc0f93f
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 23
  ERC20Mock.mint confirmed   Block: 38   Gas used: 50637 (0.42%)

Transaction sent: 0xaf40666e3308c2a8e89f42952cb876ed1781c6b43fe554e06064f0ac0238e461
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 11
  ERC20Mock.approve confirmed   Block: 39   Gas used: 44160 (0.37%)

Transaction sent: 0x3b84feab6d0825aec8caf68a35176a55913bc8e652114362644a025ae49edf4f
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 24
  ERC20Mock.mint confirmed   Block: 40   Gas used: 50625 (0.42%)

Transaction sent: 0xb66c832d3988e529d8ea6b480b1733f454fc466f9c57a378df5bf177d79fb591
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 2
  ERC20Mock.approve confirmed   Block: 41   Gas used: 44160 (0.37%)

Transaction sent: 0x76922887a8e53cb74c58783d6d884da1ad5c5a901502885ab228c51e7ccbdeb0
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 12
  AITaxStaking.deposit confirmed   Block: 42   Gas used: 184172 (1.53%)

Transaction sent: 0x799f4e9cd5fd9f4eb11db37aa816dd3074f233060bc696afd86144002c0dc000
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 3
  AITaxStaking.deposit confirmed   Block: 43   Gas used: 154172 (1.28%)

Transaction sent: 0x694f152c28c5b4d4223f388656bf634000f519642d575ba5d58423a41ea2fda6
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 13
  AITaxStaking.withdraw confirmed (Zero Amount)   Block: 44   Gas used: 27462 (0.23%)

Transaction sent: 0x66ede002a6efbd4768fb2e57d0598cb0848d299d9082b91b04b36f984e6c69e3
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 14
  AITaxStaking.withdraw confirmed (Not enough tokens)   Block: 45   Gas used: 30207 (0.25%)

Transaction sent: 0xa40975b5c8aac39e0883f5afccbeec27a15aa8cc39be9762d04aa934c4baa7c8
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 15
  AITaxStaking.withdraw confirmed (May not withdraw early)   Block: 46   Gas used: 30259 (0.25%)

Transaction sent: 0x1f782f4e2d6724c1200f913098a1cebede0521eae77095145630ede9192bab13
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 16
  AITaxStaking.withdraw confirmed   Block: 48   Gas used: 47109 (0.39%)

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



```
tests/test_staking.py::test_withdraw_all RUNNING
Transaction sent: 0x9a40725b5329ae417ba611358d9a53994d4d11598d6529413d9194b63b800a1c
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 25
  ERC20Mock.constructor confirmed   Block: 49   Gas used: 619649 (5.16%)
  ERC20Mock deployed at: 0x654f70d8442EA18904FA1AD79114f7250F7E9336

Transaction sent: 0x1cf71edd98ae5e89fc158794c5bbfbcc90f4f5afe1bbf57684c805cc2b7322ff
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 26
  ERC20Mock.mint confirmed   Block: 50   Gas used: 65649 (0.55%)

Transaction sent: 0xd8b6e4f19df6e1fc4dfcd0ef0d0a125bd29aa8d1dfbeef7d73b3d83836cb8166
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 27
  AITaxStaking.constructor confirmed   Block: 51   Gas used: 2181737 (18.18%)
  AITaxStaking deployed at: 0x832698Daec363C9A7aB036C224Af5B21280b3AC6

Transaction sent: 0xbc710865da6d0308b0b0febc1afa05f8b1dab2e5a1c992815c321915596fad61
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 28
  ERC20Mock.mint confirmed   Block: 52   Gas used: 50637 (0.42%)

Transaction sent: 0xf2b97f3a5d0118d66dabdac04434666d7f70fcfc1cf2b41dd91cd65ee2223a57
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 17
  ERC20Mock.approve confirmed   Block: 53   Gas used: 44160 (0.37%)

Transaction sent: 0x79a8c58c1cfc9c309a8f262733008789de121c685901bbe49312fd0737279508
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 18
  AITaxStaking.withdrawAll confirmed (Not a holder)   Block: 54   Gas used: 29882 (0.25%)

Transaction sent: 0x380cb4d704ad199abcec7dc1dd7310a3e0b3bed775b1ae97397fceb84e004eed
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 19
  AITaxStaking.deposit confirmed   Block: 55   Gas used: 184172 (1.53%)

Transaction sent: 0xa1451f41b5eda96e461af8364381fff166e5320c067f971fe4496aa05c2e5ee3
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 20
  AITaxStaking.withdrawAll confirmed (May not withdraw early)   Block: 56   Gas used: 29934 (0.25%)

Transaction sent: 0xf2b788c01b56404a1f945fb98441be2c3a6f1e7f3f31758926864ad7d07116ed
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 21
  AITaxStaking.withdrawAll confirmed   Block: 58   Gas used: 39932 (0.33%)

tests/test_staking.py::test_withdraw_all PASSED
```



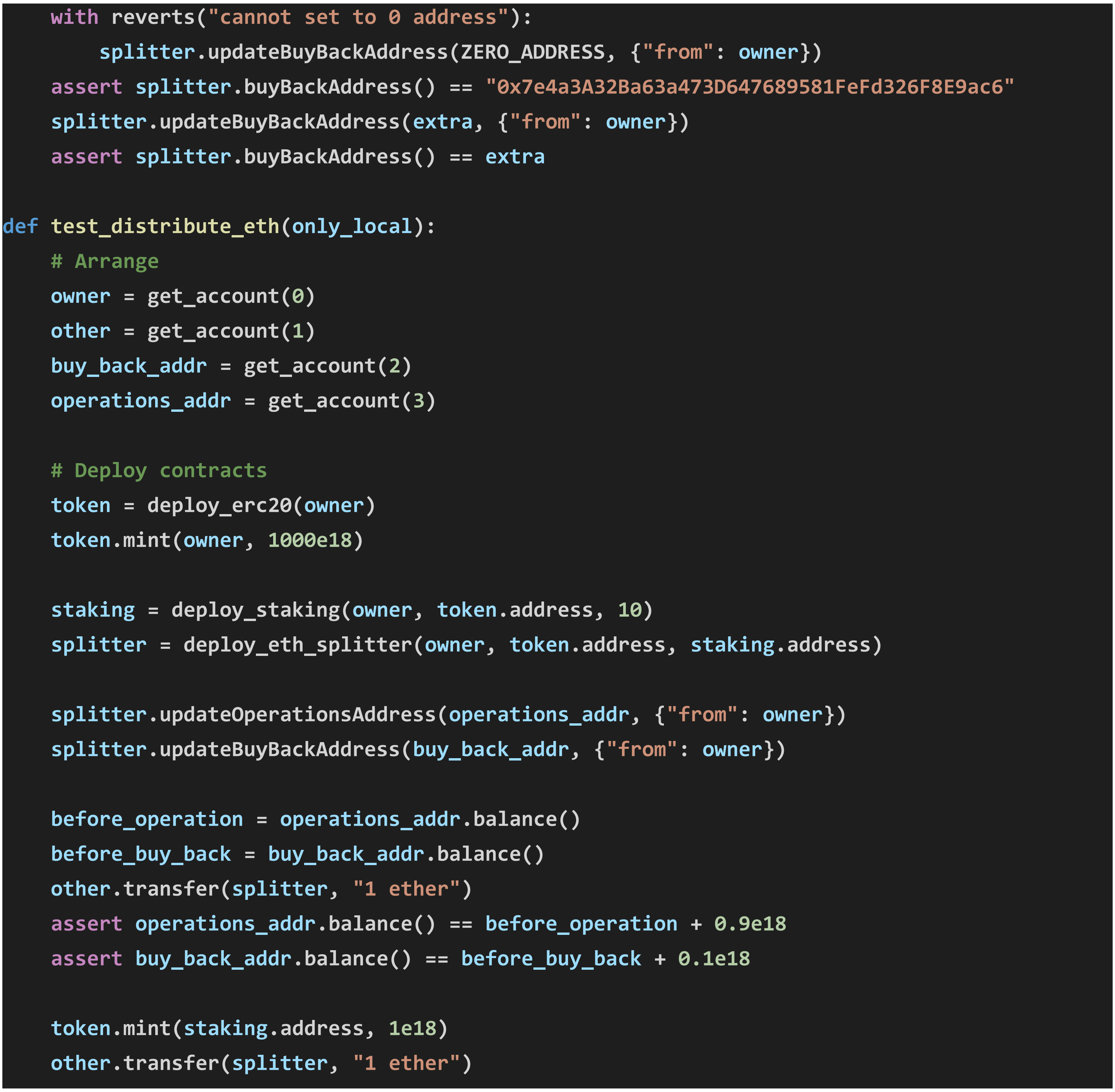
```
tests/test_staking.py::test_coumpound_claim RUNNING
Transaction sent: 0x62e50714c8c686f7f061055fe3922faf2a267c57c5de81d655bf529edd950665
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 29
  ERC20Mock.constructor confirmed   Block: 59   Gas used: 619649 (5.16%)
  ERC20Mock deployed at: 0x42E8D004c84E6B5Bad559D3b5CE7947AADb9E0bc

Transaction sent: 0xb8c86f8d4202540a3203aef61a06e9ae1fa59901bcb9c7e6e51b6102f49311b8
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 30
  ERC20Mock.mint confirmed   Block: 60   Gas used: 65649 (0.55%)

Transaction sent: 0x0bde203bdfb7c8a0ac31f1d6086cd9d222a2bc252aebc05b846c38983a16178f
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 31
  AITaxStaking.constructor confirmed   Block: 61   Gas used: 2181737 (18.18%)
  AITaxStaking deployed at: 0x82c83b7f88aef2eD99d4869D547b6ED28e69C8df

Transaction sent: 0x1bbc15984855ca7587c273351f40979c87c336fe2607143eff68277bfc9f528d
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 22
  AITaxStaking.compound confirmed (No rewards)   Block: 62   Gas used: 34594 (0.29%)

Transaction sent: 0xf16fd2596d2eeb90bd7e4b4df47c6fb2e721a210c2a6a01e52ed0235a4092365
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 32
  ERC20Mock.mint confirmed   Block: 63   Gas used: 50637 (0.42%)

Transaction sent: 0xfb84101bbcad38feb749e60a08bbeaf2de358db76e4023595e3b0d2814c3576a
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 23
  ERC20Mock.approve confirmed   Block: 64   Gas used: 44160 (0.37%)

Transaction sent: 0x8ef1676f99258e10a7e43cce24140dd9b3ad79f3fa482b52735e08e968448aff
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 24
  AITaxStaking.deposit confirmed   Block: 65   Gas used: 184172 (1.53%)

Transaction sent: 0x9ccbba65dfa66666aa2428baaca813dd3bb2f352c59ae9ec0a0f7104b134d7fa
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 33
  ERC20Mock.mint confirmed   Block: 66   Gas used: 50625 (0.42%)

Transaction sent: 0x46d15af47d3bc3f8e95a5089f793353fab51f59941c82cad441b8d2e4c4135b9
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 4
  ERC20Mock.approve confirmed   Block: 67   Gas used: 44160 (0.37%)

Transaction sent: 0xa760c5f1192433cecdd1cd387282e16599db17f33821b64f1f71f890d02f4bd8
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 5
  AITaxStaking.deposit confirmed   Block: 68   Gas used: 154172 (1.28%)

Transaction sent: 0x0577750f1e72ae5571e126dcce6c4b6dd9b726ee8d9c63c231015de0838028f4
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 26
  AITaxStaking.claim confirmed   Block: 71   Gas used: 60336 (0.50%)

Transaction sent: 0xeb66046f49f90470696c7c23971983e7cb7fb6e337ffe931e51d2782d975e822
  Gas price: 0.0 gwei   Gas limit: 12000000   Nonce: 27
  AITaxStaking.compound confirmed (No rewards)   Block: 72   Gas used: 34759 (0.29%)

tests/test_staking.py::test_coumpound_claim PASSED
```

Annexes

Testing code:

`eth_splitter.py:`

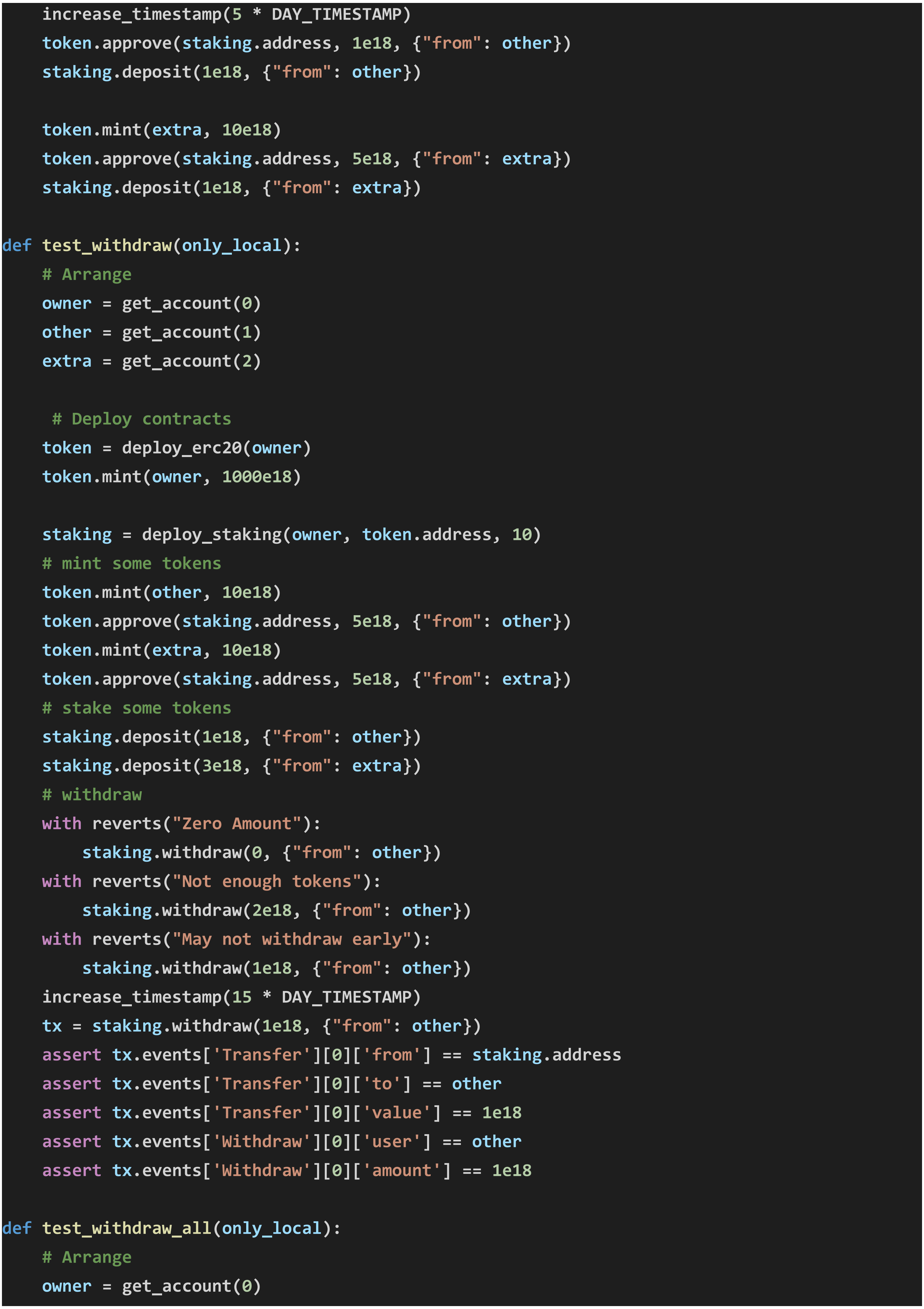
```python
from brownie import (
    reverts
)

from scripts.helpful_scripts import (
    ZERO_ADDRESS,
    get_account,
)

from scripts.deploy import (
    deploy_eth_splitter,
    deploy_staking,
    deploy_erc20,
)

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

    # Deploy contracts
    token = deploy_erc20(owner)
    token.mint(owner, 1000e18)

    staking = deploy_staking(owner, token.address, 10)
    splitter = deploy_eth_splitter(owner, token.address, staking.address)

    with reverts("Ownable: caller is not the owner"):
        splitter.updateOperationsAddress(extra, {"from": other})
    with reverts("cannot set to 0 address"):
        splitter.updateOperationsAddress(ZERO_ADDRESS, {"from": owner})
    assert splitter.operationsAddress() == "0x832bb8DC475F4cF9C0e19Fb4118F7A57e893147e"
    splitter.updateOperationsAddress(extra, {"from": owner})
    assert splitter.operationsAddress() == extra

    with reverts("Ownable: caller is not the owner"):
        splitter.updateBuyBackAddress(extra, {"from": other})
```



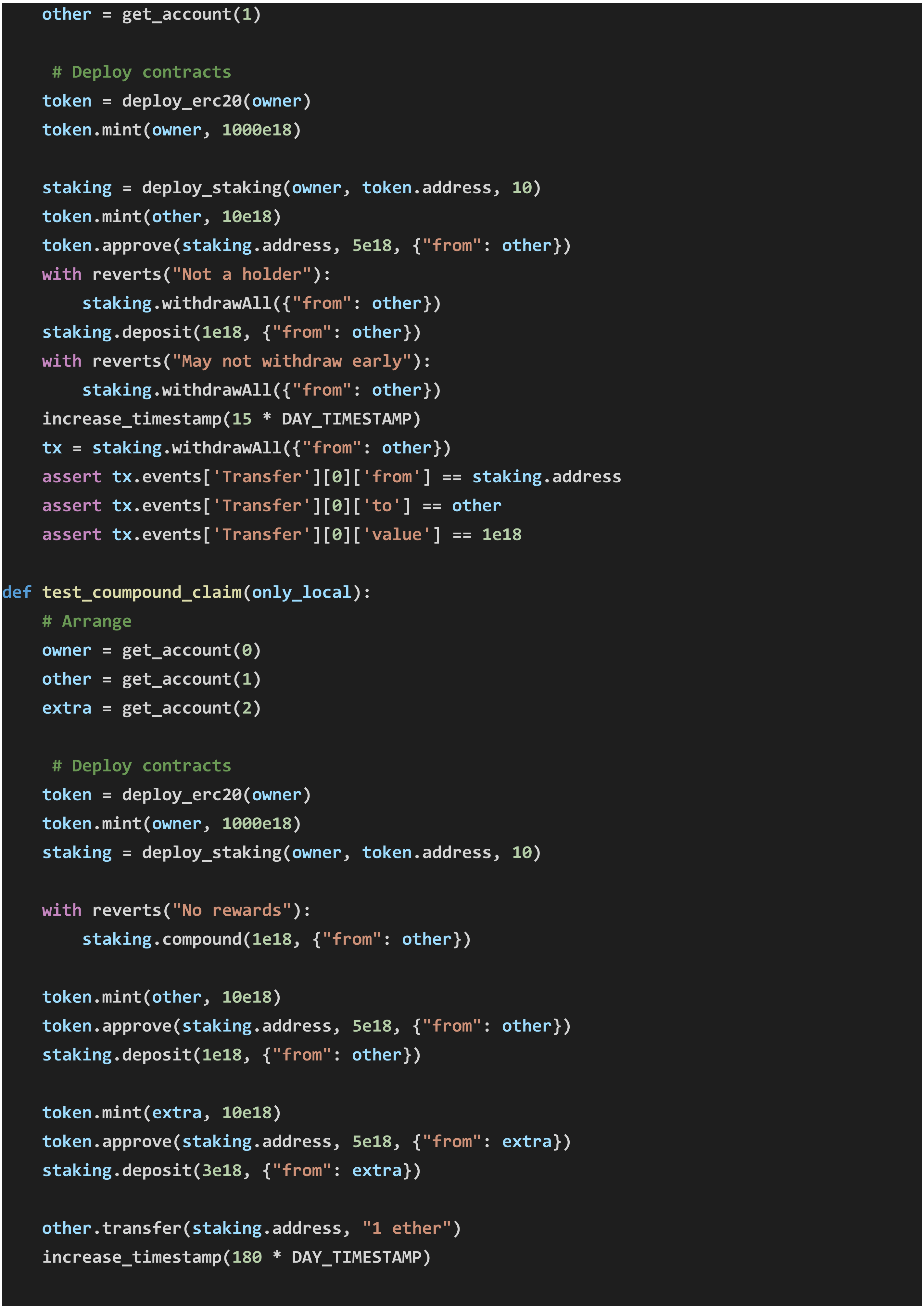
```python
    with reverts("cannot set to 0 address"):
        splitter.updateBuyBackAddress(ZERO_ADDRESS, {"from": owner})
    assert splitter.buyBackAddress() == "0x7e4a3A32Ba63a473D647689581FeFd326F8E9ac6"
    splitter.updateBuyBackAddress(extra, {"from": owner})
    assert splitter.buyBackAddress() == extra

def test_distribute_eth(only_local):
    # Arrange
    owner = get_account(0)
    other = get_account(1)
    buy_back_addr = get_account(2)
    operations_addr = get_account(3)

    # Deploy contracts
    token = deploy_erc20(owner)
    token.mint(owner, 1000e18)

    staking = deploy_staking(owner, token.address, 10)
    splitter = deploy_eth_splitter(owner, token.address, staking.address)

    splitter.updateOperationsAddress(operations_addr, {"from": owner})
    splitter.updateBuyBackAddress(buy_back_addr, {"from": owner})

    before_operation = operations_addr.balance()
    before_buy_back = buy_back_addr.balance()
    other.transfer(splitter, "1 ether")
    assert operations_addr.balance() == before_operation + 0.9e18
    assert buy_back_addr.balance() == before_buy_back + 0.1e18

    token.mint(staking.address, 1e18)
    other.transfer(splitter, "1 ether")
```

eth_staking.py:

```
from brownie import (
    reverts
)

from brownie.network.contract import Contract

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

from scripts.deploy import (
    deploy_eth_splitter,
    deploy_staking,
    deploy_erc20,
)

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

    # Deploy contracts
    token = deploy_erc20(owner)
    token.mint(owner, 1000e18)

    staking = deploy_staking(owner, token.address, 10)
    with reverts("Zero Amount"):
        staking.deposit(0, {"from": other})
    with reverts("ERC20: insufficient allowance"):
        staking.deposit(1e18, {"from": other})
    token.approve(staking.address, 1e18, {"from": other})
    with reverts("ERC20: transfer amount exceeds balance"):
        staking.deposit(1e18, {"from": other})
    token.mint(other, 10e18)
    tx = staking.deposit(1e18, {"from": other})
    assert tx.events['Transfer'][0]['from'] == other
    assert tx.events['Transfer'][0]['to'] == staking.address
    assert tx.events['Transfer'][0]['value'] == 1e18
```



```
    increase_timestamp(5 * DAY_TIMESTAMP)
    token.approve(staking.address, 1e18, {"from": other})
    staking.deposit(1e18, {"from": other})

    token.mint(extra, 10e18)
    token.approve(staking.address, 5e18, {"from": extra})
    staking.deposit(1e18, {"from": extra})

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

     # Deploy contracts
    token = deploy_erc20(owner)
    token.mint(owner, 1000e18)

    staking = deploy_staking(owner, token.address, 10)
    # mint some tokens
    token.mint(other, 10e18)
    token.approve(staking.address, 5e18, {"from": other})
    token.mint(extra, 10e18)
    token.approve(staking.address, 5e18, {"from": extra})
    # stake some tokens
    staking.deposit(1e18, {"from": other})
    staking.deposit(3e18, {"from": extra})
    # withdraw
    with reverts("Zero Amount"):
        staking.withdraw(0, {"from": other})
    with reverts("Not enough tokens"):
        staking.withdraw(2e18, {"from": other})
    with reverts("May not withdraw early"):
        staking.withdraw(1e18, {"from": other})
    increase_timestamp(15 * DAY_TIMESTAMP)
    tx = staking.withdraw(1e18, {"from": other})
    assert tx.events['Transfer'][0]['from'] == staking.address
    assert tx.events['Transfer'][0]['to'] == other
    assert tx.events['Transfer'][0]['value'] == 1e18
    assert tx.events['Withdraw'][0]['user'] == other
    assert tx.events['Withdraw'][0]['amount'] == 1e18

def test_withdraw_all(only_local):
    # Arrange
    owner = get_account(0)
```



```
    other = get_account(1)

     # Deploy contracts
    token = deploy_erc20(owner)
    token.mint(owner, 1000e18)

    staking = deploy_staking(owner, token.address, 10)
    token.mint(other, 10e18)
    token.approve(staking.address, 5e18, {"from": other})
    with reverts("Not a holder"):
        staking.withdrawAll({"from": other})
    staking.deposit(1e18, {"from": other})
    with reverts("May not withdraw early"):
        staking.withdrawAll({"from": other})
    increase_timestamp(15 * DAY_TIMESTAMP)
    tx = staking.withdrawAll({"from": other})
    assert tx.events['Transfer'][0]['from'] == staking.address
    assert tx.events['Transfer'][0]['to'] == other
    assert tx.events['Transfer'][0]['value'] == 1e18

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

     # Deploy contracts
    token = deploy_erc20(owner)
    token.mint(owner, 1000e18)
    staking = deploy_staking(owner, token.address, 10)

    with reverts("No rewards"):
        staking.compound(1e18, {"from": other})

    token.mint(other, 10e18)
    token.approve(staking.address, 5e18, {"from": other})
    staking.deposit(1e18, {"from": other})

    token.mint(extra, 10e18)
    token.approve(staking.address, 5e18, {"from": extra})
    staking.deposit(3e18, {"from": extra})

    other.transfer(staking.address, "1 ether")
    increase_timestamp(180 * DAY_TIMESTAMP)
    tx = staking.claim({"from": other})
    assert tx.events['DividendWithdrawn'][0]['to'] == other
    assert tx.events['Claim'][0]['account'] == other

    with reverts(""):
        staking.compound(1e17, {"from": other})
```

Technical Findings Summary

Findings

Vulnerability Level	Total	Pending	Not Apply	Acknowledged	Partially Fixed	Fixed
🔴 High	2					
🟠 Medium	3					
🟡 Low	1					
🟢 Informational	0					

Assessment Results

Score Results

Review	Score
Global Score	**35/100**
Assure KYC	https://www.assuredefi.com/projects/aitax/
Audit Score	30/100

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 FAILED

Following our comprehensive security audit of the staking contract for AITAX project, the audit has failed due to multiple high-severity issues detected in the smart contract. These include improper variable naming in the distributeETH() function and the lack of error handling after payable calls in both distributeETH() and withdrawStuckETH(), which prevent successful compilation and can lead to uncaught errors in transactions. Addressing these critical issues is essential to ensure the security and operational integrity of the smart contract plus reviewing and fixing all the medium issues.

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 advice 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 advice, 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 services 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 services, 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 services may access, and depend upon, multiple layers of third parties.