Vaults and Actions

Smart Contract Audit

Vaults and Actions

Smart Contract Audit

V220406

Prepared for FIATDAO • February 2022

1. Executive Summary

In **February 2022, FIATDAO** engaged Coinspect to perform a source code review of the FIAT Protocol's **Vaults and Actions**. The objective of the project was to continue to evaluate the security of the smart contracts.

This audit focused on the smart contracts implementing FIATDAO's Vaults logic and their related user actions contracts.

Overall, Coinspect found the smart contracts to be properly designed and implemented. The source code is extensively documented and follows smart contract security best practices.

The third party contracts and protocols integrated by the new vaults were outside the scope of this engagement.

No issues were identified during the assessment.

High Risk	Medium Risk	Low Risk
0	0	0
Fixed -	Fixed -	Fixed -

2. Assessment and Scope

The audit started on **February 21, 2022** and was conducted over the files located at two different git repositories:

1. Vault logic contracts located at https://github.com/fiatdao/vaults on the main branch as of commit `10fabad32cf75dd0289be230d1eed0108f4d7a78`
2. Actions contracts located at https://github.com/fiatdao/actions on the main branch as of commit `198e328e8d98a7d19d1bc8cdbd9d3fc34eaf042f`

The scope of the audit was limited to the following Solidity source files, with their `sha256sum` hash:

```
85f25ddf2c8401563c7140af64fe2a67d6eda59ccfe87ed9e4103874baefafc6  VaultEPT.sol
6286de08ed4f4af2750629177aa0fcb6cc7faa7cb6fc7b8a34f0d8493e6cdc49  VaultFactory.sol
4630faee5fd82cfbf862bc1ca2caba4cace74b9085b7b5deee41915229f3a640  VaultFC.sol
312081ae7cc3530e32bc12dd887c62f522d7286a57a7cd1f81e9eab3a8252530  Vault.sol
3534391be044d049bad7186eaaf490022554b4b7aca71321e2965e71ae285bd7  VaultSY.sol
23ebefa506fbc3eb93ad87e7b8a3a59f3cb57a5ca7023d68531fef1953424e64  Vault1155Actions.sol
10722c736e8a95426977cd95e4329b81aa7fd0e00def5f4b543ce0ae5cfc2574  Vault20Actions.sol
57a502addfba63856aa294da7ab8d69512bc50a7e3e95f05aef76874f727eebf  VaultActions.sol
8dfe79a0ee7d787d03642ffd539c77955403bd8a30ad004a6a3b9ab5453dc9b7  VaultEPTActions.sol
a0eb4a709c808d0d5d02f21a9d91bff9985871537e0791240b45e17fcc3967b5  VaultFCActions.sol
e6e9a20f24e4a61dea0444882e5c45ea8213030dc45866387153ea464c4eab3d  VaultSYActions.sol
```

The new FIATDAO Vaults interact with contracts deployed and controlled by third parties: Barnbridge Smart Yield Bonds, Notional Finance fCash Vault and Element Finance Principal Tokens, which were not part of the scope for this audit.

The Vault contracts are critical security wise because as noted in Coinspect's previous audit, they are implicitly trusted by the Codex contract responsible for the accounting in the platform. Coinspect verified the correctness of the Vault code implementation and focused on the safety of the interactions with those external contracts. The VaultEPT contract is an EIP1167 minimal proxy implementation, and its instances must be created through the VaultFactory contract, which atomically creates the contract and calls the initialization function atomically.

The Actions contracts' purpose is to enable EOAs to encapsulate multiple calls that interact with the FIAT protocol. They are intended to be called from each user's proxy only, and calling them directly could result in funds being lost or stolen as clearly stated in the source code. Even though the proxy implementation was not fully audited, the overall design and usage pattern was reviewed in order to evaluate if the contract assumptions were correct. Each Action contract interacts with the corresponding Vault and allows operating with different assets as required per each external protocol being integrated. In the VaultEPTActions contract, token swaps are performed through Balancer. The user actions contracts are intended to be **always** executed from a user proxy described at https://github.com/fiatdao/proxy/tree/fiatdao-dev which was not part of this engagement.

In **March 2022**, **FIATDAO** split the original fiat-lux repository in three, renaming the `fiat-lux` components in scope to "Actions" and "Vaults", respectively. Coinspect updated the report, after reviewing the changes performed to the repositories at:

1. Vault logic contracts located at https://github.com/fiatdao/vaults on the main branch as of commit `76b1f5a28ab77b6e9080b375783898fee6bc90a9`
2. Actions contracts located at https://github.com/fiatdao/actions on the main branch as of commit `69eae2f46c579170a3cbc53a1e511b205974f866`

The updated Solidity source files, with their `sha256sum` hash:

```
b41a86cbbdf3c254e515a8bca7691f76864fdb856665786f383146f340edcf6c  VaultEPT.sol
6620f48418d86daf7d2fadfcd89c0ed7f3a5b3cb73afc6ffc87a807de438a6d3  VaultFactory.sol
15ec6e9937b893f2b201eb63623bfd85e79a4f82146fe8084b6516fc7f368fc1  VaultFC.sol
e65f5521304b056ff980e687cf012e0efbbb07ba954ac59f9efa65e0af0dad91  Vault.sol
c8f95e3f48b9e141328542e0eeed0b56e4aa6b45d8b8e1813e6f611f6dfc8cc3  VaultSY.sol
23ebefa506fbc3eb93ad87e7b8a3a59f3cb57a5ca7023d68531fef1953424e64  Vault1155Actions.sol
10722c736e8a95426977cd95e4329b81aa7fd0e00def5f4b543ce0ae5cfc2574  Vault20Actions.sol
57a502addfba63856aa294da7ab8d69512bc50a7e3e95f05aef76874f727eebf  VaultActions.sol
8dfe79a0ee7d787d03642ffd539c77955403bd8a30ad004a6a3b9ab5453dc9b7  VaultEPTActions.sol
fb8532c5b6fff5089063e5b20d4efc418343bfc1106f42cfd7944a1a29c78756  VaultFCActions.sol
e6e9a20f24e4a61dea0444882e5c45ea8213030dc45866387153ea464c4eab3d  VaultSYActions.sol
```

No findings were identified during this engagement.

Minor Issues and Suggestions

1. The following comment in the VaultSY contract is incorrect:
   ```
   /// @dev Caller has to set allowance for this contract
   function fairPrice(
   ```
2. The following comment in the VaultSYActions contract is incorrect:
   ```
   /// @dev Implements virtual method defined in VaultActions for ERC20 tokens
   function exitVault(
   ```

3. Compiler emits 2 warnings related to imported Notional SDK:

```
Warning: Statement has no effect.
   --> lib/notional-solidity-sdk/contracts/lib/EncodeDecode.sol:167:30:
    |
167 |         if (isIdiosyncratic) (bytes32(0), false);
    |                              ^^^^^^^^^^^^^^^^^^^

Warning: Function state mutability can be restricted to pure
   --> lib/notional-solidity-sdk/contracts/lib/EncodeDecode.sol:177:5:
    |
177 |     function encodeOffsettingTradesFromPortfolio(
    |     ^ (Relevant source part starts here and spans across multiple lines).
```

3. Disclaimer

The information presented in this document is provided "as is" and without warranty. The present security audit does not cover any off-chain systems or frontends that communicate with the contracts, nor the general operational security of the organization that developed the code.