Arkreen NetWork

Audit Report

Thu May 30 2024

✉ contact@bitslab.xyz https://twitter.com/scalebit_

ScaleBit

Arkreen NetWork Audit Report

1 Executive Summary

1.1 Project Information

Description	Arkreen is a Web3-powered digital infrastructure for globally distributed renewable energy resources.
Type	DePIN+ReFi
Auditors	ScaleBit
Timeline	Mon Mar 18 2024 - Thu May 30 2024
Languages	Solidity
Platform	Polygon
Methods	Architecture Review, Unit Testing, Manual Review
Source Code	https://github.com/arkreen/ArkreenCore
Commits	f171c91d057400c50e0fbe398fbdb6e3770a155c 0e5c4228584a2e70c6e8d497e2d4c026985e9b67

1.2 Files in Scope

The following are the SHA1 hashes of the original reviewed files.

ID	File	SHA-1 Hash
ABS	contracts/ArkreenBuilderStorage.sol	567a8dc11ac61349471eae051417be1855874c20
ARECIE	contracts/ArkreenRECIssuanceExt.sol	87b8f10324272c85c2df1f503dd6d1cd398cb74b
ARECI	contracts/ArkreenRECIssuance.sol	18b626b1278627c00a093189fa080998b9f31437
ABU	contracts/ArkreenBuilder.sol	03778d6af912596185c0e228ba56cccee4ac2835
AMI	contracts/ArkreenMiner.sol	9e0f8b4ace276cf55d4c47258ac965914ce5da9d
UUPSP	contracts/UUPSProxy.sol	a123a720ddfff6d1ffc4a402b03132d61a61525d
ARECIT	contracts/ArkreenRECIssuanceType.sol	287a397b69ba0786079c41f56538d2283e6f8a27
ABS1	contracts/ArkreenBadgeStorage.sol	295e2fd805ec35734ddb4ad6d89a606b6bdc568f
ABA	contracts/ArkreenBadge.sol	b6cbbf582b221de8ee5b387f5ba05f81761f90db
GBTCI	contracts/GreenBTCImage.sol	281e784ea650cfdbc5cf86b991299d0798fe461b
ARE	contracts/ArkreenRegistry.sol	af142b0a9a906d7baf7adef7de4a93ed5b36e2b1

ABI	contracts/ArkreenBadgeImage.sol	d37d3d8b1304bf71cfed6332cadc211f89d28d87
GBTCT	contracts/GreenBTCType.sol	edc36e80c81dfc838485ad478b50dd7aedab8cf8
ARECIS	contracts/ArkreenRECIssuanceStorage.sol	840b281f138a93fdc92771d36491e6f0c3aebb91
ATL	contracts/Arkreen/ArkreenTimeLock.sol	becb3c8525a778588c7f14bde92c1a7868162b4c
COW	contracts/Arkreen/ConfirmedOwner.sol	4647887b3ace977925c7268acd98339ed4f41c29
AKREV	contracts/Arkreen/AKREVesting.sol	01b8e444f9fedbd68ed564db035640adf35f91ea
ANO	contracts/Arkreen/ArkreenNotary.sol	88a0249990e3839c4b40a56781a81cfc08266adb
ATO	contracts/Arkreen/ArkreenToken.sol	8f1fb416aee0277477064b2bbacca10ba8111527
AGO	contracts/Arkreen/ArkreenGovernor.sol	200077edfc9bfe453a596c764f375de41f2f4dbf
ARE1	contracts/Arkreen/ArkreenReward.sol	945a23158d9a857647535d0a2381d22970e73f8e
GBTC	contracts/GreenBTC.sol	c2073d8128c6890d3ff40d54f9c85764ead16e2a
ARECIIL	contracts/ArkreenRECIssuanceImageLogo.sol	f52bb4c510019bba92be8d2ab600e17a8b2f7a91
ARECT	contracts/ArkreenRECToken.sol	230aa65fd86e70d02634359502cb7491f1821c3c

ARECII	contracts/ArkreenRECIssuanceImage.sol	de35ed4125d1de1be07ee6197e0742c02c6cc864
ARECB	contracts/ArkreenRECBank.sol	f2a15ed8daedfb90b02ff232cea0c2430ba273cb
ABT	contracts/ArkreenBadgeType.sol	64b8dccdbac455690bbcb26eddf03447d65f6205
AMT	contracts/ArkreenMinerTypes.sol	d1ab0abf990d74c08a6453f8e354753493bc00ca
ABT1	contracts/ArkreenBuilderTypes.sol	91b9c1d9049f2e75d0aa9b303346117b3fd2fd41
ARS	contracts/ArkreenRegistryStorage.sol	7983d22c870cc4c3bdf4e2edb2d027d0c5e2c26b

1.3 Issue Statistic

Item	Count	Fixed	Acknowledged
Total	9	6	3
Informational	1	1	0
Minor	1	0	1
Medium	5	3	2
Major	2	2	0
Critical	0	0	0

1.4 ScaleBit Audit Breakdown

ScaleBit aims to assess repositories for security-related issues, code quality, and compliance with specifications and best practices. Possible issues our team looked for included (but are not limited to):

- Transaction-ordering dependence
- Timestamp dependence
- Integer overflow/underflow
- Number of rounding errors
- Unchecked External Call
- Unchecked CALL Return Values
- Functionality Checks
- Reentrancy
- Denial of service / logical oversights
- Access control
- Centralization of power
- Business logic issues
- Gas usage
- Fallback function usage
- tx.origin authentication
- Replay attacks
- Coding style issues

1.5 Methodology

The security team adopted the **"Testing and Automated Analysis"**, **"Code Review"** and **"Formal Verification"** strategy to perform a complete security test on the code in a way that is closest to the real attack. The main entrance and scope of security testing are stated in the conventions in the "Audit Objective", which can expand to contexts beyond the scope according to the actual testing needs. The main types of this security audit include:

(1) Testing and Automated Analysis

Items to check: state consistency / failure rollback / unit testing / value overflows / parameter verification / unhandled errors / boundary checking / coding specifications.

(2) Code Review

The code scope is illustrated in section 1.2.

(3) Audit Process

- Carry out relevant security tests on the testnet or the mainnet;
- If there are any questions during the audit process, communicate with the code owner in time. The code owners should actively cooperate (this might include providing the latest stable source code, relevant deployment scripts or methods, transaction signature scripts, exchange docking schemes, etc.);
- The necessary information during the audit process will be well documented for both the audit team and the code owner in a timely manner.

2 Summary

This report has been commissioned by Arkreen NetWork to identify any potential issues and vulnerabilities in the source code of the Arkreen smart contract, as well as any contract dependencies that were not part of an officially recognized library. In this audit, we have utilized various techniques, including manual code review and static analysis, to identify potential vulnerabilities and security issues.

During the audit, we identified 9 issues of varying severity, listed below.

ID	Title	Severity	Status
ABA-1	Re-Entry Risks	Informational	Fixed
ANO-1	Uninitialized Logic Contracts in UUPS Proxy Mode	Medium	Fixed
ARE-1	Signature Malleability	Medium	Fixed
ARE-2	Missing Emit Event	Minor	Acknowledged
GBT-1	`setNewCaps` has No Limitations	Major	Fixed
IAR-1	Incompatible With Deflationary Token	Medium	Acknowledged
ARE1-1	Possible Inability to Withdraw Funds	Major	Fixed
ARE1-2	May be Lack of Check	Medium	Fixed
ARE1-3	Centralization Risk	Medium	Acknowledged

3 Participant Process

Here are the relevant actors with their respective abilities within the Arkreen Smart Contract :

Admin

- The owner can call the `setRatioFee` function to set the ratio of liquidation fee.
- The owner can utilize the `setIssuerREC` function to change the REC issuance address.
- The owner can invoke the `setRatioFeeToSolidify` function to set the ratio of solidify fee to Solidify from ART to AREC.
- The owner can utilize the `setRatioFeeOffset` function to set the ratio of fee to offset ART as climate action.
-

User

- Users can call the `commitOffset` and `commitOffsetFrom` functions to offset the RE token.
- Users can call the `solidify` function to solidify the ART token to AREC NFT.
- Users can call the `mintRECRequest` function to mint the REC NFT.
- Users can invoke the `certifyRECRequest` function to certify the REC NFT mint request by the REC issuer.
- Users can call the `redeem` and `redeemFrom` functions to redeem the REC NFT by retiring the NFT and registering an offset action.
- Users can invoke the `liquidizeREC` function to liquidize the REC NFT and mint the corresponding ERC20 token.

4 Findings

ABA-1 Re-Entry Risks

Severity: Informational

Status: Fixed

Code Location:

contracts/ArkreenBadge.sol#255;

contracts/ArkreenMiner.sol#345;

contracts/ArkreenRECIssuance.sol#143;

contracts/ArkreenRECIssuanceExt.sol#111

Descriptions:

There are many functions in the contract that have external calls, such as calls to functions such as `_safeMint` and so on, and although this is not a problem right now, it's a good idea to add precautions to avoid the risk of reentry attacks in the future.

Suggestion:

It is recommended to use the `nonReentrant` reentrant lock provided by Openzeppelin.

Resolution:

The client has added the re-entry protection as per our suggestion.

ANO-1 Uninitialized Logic Contracts in UUPS Proxy Mode

Severity: Medium

Status: Fixed

Code Location:

contracts/Arkreen/ArkreenNotary.sol#35-41

Descriptions:

The `ArkreenNotary` contract uses the UUPS proxy model, and there is no restriction in the contract as to whether or not the logical contract can be initialized, i.e. if the project owner doesn't physically initialize the logical contract after deploying the proxy contract, then anyone can initialize the logical contract and become the owner of the logical contract, which feels insecure.

Suggestion:

It is recommended to use the `_disableInitializers` provided by openzeppelin's `initializable` contract to disable the initialization of the logic contract.

Resolution:

The client has improved the code based on the suggestions.

ARE-1 Signature Malleability

Severity: Medium

Status: Fixed

Code Location:

contracts/Arkreen/ArkreenReward.sol#95

Descriptions:

The elliptic curve used in Ethereum for signatures is symmetrical, hence for every `v,r,s` there exists another `v,r,s` that returns the same valid result. Therefore two valid signatures exist which allows attackers to compute a valid signature without knowing the signer's private key. `ecrecover()` is vulnerable to signature malleability [1, 2] so it can be dangerous to use it directly. An attacker can compute another corresponding `v,r,s` that will make this check pass due to the symmetrical nature of the elliptic curve.

Suggestion:

It is recommended to use OpenZeppelin’s ECDSA.sol library and reading the comments above ECDSA's `tryRecover()` function provides very useful information on correctly implementing signature checks to prevent signature malleability vulnerabilities. When using OpenZeppelin's ECDSA library, special care must be taken to use version 4.7.3 or greater, since previous versions contained a signature malleability bug.

Resolution:

The client has improved the code based on the suggestions.

ARE-2 Missing Emit Event

Severity: Minor

Status: Acknowledged

Code Location:

contracts/Arkreen/ArkreenReward.sol#69-78;

contracts/GreenBTC.sol#119,124,129,134,139,492;

contracts/ArkreenRECIssuance.sol#478,483,487,491,509

Descriptions:

The smart contract lacks appropriate events for monitoring sensitive operations(such as managing assets and modifying key configs), which could make it difficult to track important actions or detect potential issues.

Suggestion:

It is recommended to emit events for these sensitive functions to make it easier to track important actions or detect potential issues.

Resolution:

The client is already aware of the recommendation.

GBT-1 `setNewCaps` has No Limitations

Severity: Major

Status: Fixed

Code Location:

contracts/GreenBTC.sol#492-496

Descriptions:

The `setNewCaps` function changes the values of the `newNormalCap` , `newOvertimeCap` , and `newRemoveCap` parameters. `newNormalCap` determines how many blocks can be revealed at a time in the revealBoxes function, according to the comments, is expected to reveal 200 blocks once a time, but the `setNewCaps` function is public and has no restrictions, anyone can call it, it does not seem to be a function that can be called arbitrarily, too small caps may lead to a lot of unnecessary overtime boxes.

Suggestion:

It is recommended to confirm that this situation is in line with the design concept.

Resolution:

The client has added restrictions to address this issue.

IAR-1 Incompatible With Deflationary Token

Severity: Medium

Status: Acknowledged

Code Location:

contracts/interfaces/IArkreenRECBank.sol#164,176

Descriptions:

In the `_buyART` function, due to the unknown address of `tokenPay`, when the token is deflationary, the number of tokens transferred to the contract by the user may not be accurate.

Suggestion:

Since it's not known exactly what type of token this is, it's recommended to confirm whether such a question would conflict with the design philosophy.

Resolution:

The client said they will not accept any deflationary token in this contract.

ARE1-1 Possible Inability to Withdraw Funds

Severity: Major

Status: Fixed

Code Location:

contracts/ArkreenRECBank.sol#243-259

Descriptions:

`saleIncome` records the revenue from each combination of `artToken` and `payToken`. Each artToken's `controller` can extract the `payToken` stored in the `ArkreenRECBank` address, and this value is recorded by `mountReceived`, which is increased at each purchase of the `artToken`, however, in the `withdraw` function, this value is not cleared to zero after the `controller` extracts all the sales incomes, which leads to the fact that if there are subsequent users who continue to buy `artTokens`, the `mountReceived` continues to increase, but there are not enough funds stored in the pool, the `controller` will have no way to withdraw the funds in the pool, so all the funds passed to the `ArkreenRECBank` address will not be able to be withdrawn.

Suggestion:

It is recommended to zero out the value of `mountReceived` after the `controller` has withdrawn the funds.

Resolution:

The client has improved the code based on the suggestions.

ARE1-2 May be Lack of Check

Severity: Medium

Status: Fixed

Code Location:

contracts/ArkreenRECIssuance.sol#134

Descriptions:

Call `mintRECRequest` function must require that the incoming token is `tokenAKRE` or the corresponding rateToIssue is not equal to 0. Finally, the price that the user needs to pay is determined by `rateToIssue` and `amountREC`. Since we do not know the rateToIssue corresponding to `tokenAKRE`, if the `rateToIssue` is equal to zero then the user does not need to pay the tokens for casting NFTs.

Suggestion:

It is recommended to confirm that this is compatible with the design concept.

Resolution:

The client changed the code to ensure that `rateToIssue` is not 0.

ARE1-3 Centralization Risk

Severity: Medium

Status: Acknowledged

Code Location:

contracts/ArkreenRECIssuance.sol#402;

contracts/Arkreen/ArkreenToken.sol#46;

contracts/Arkreen/AKREVesting.sol#132

Descriptions:

Centralization risk was identified in the smart contract.

- The owner can pause and unpause the token transfers.
- The owner can with any amount of `AKREVesting`.
- The owner can change the infomation of NFT `ArkreenBadge`.
- The owner can Add/update/remove `AREC isssaunce` payment price.

Suggestion:

It is recommended to take measures to mitigate this issue.

Resolution:

The client will transfer the smart contract ownership to the Foundation multi-sig Safe wallet while the application is stable to fix this issue.

Appendix 1

Issue Level

- **Informational** issues are often recommendations to improve the style of the code or to optimize code that does not affect the overall functionality.
- **Minor** issues are general suggestions relevant to best practices and readability. They don't post any direct risk. Developers are encouraged to fix them.
- **Medium** issues are non-exploitable problems and not security vulnerabilities. They should be fixed unless there is a specific reason not to.
- **Major** issues are security vulnerabilities. They put a portion of users' sensitive information at risk, and often are not directly exploitable. All major issues should be fixed.
- **Critical** issues are directly exploitable security vulnerabilities. They put users' sensitive information at risk. All critical issues should be fixed.

Issue Status

- **Fixed:** The issue has been resolved.
- **Partially Fixed:** The issue has been partially resolved.
- **Acknowledged:** The issue has been acknowledged by the code owner, and the code owner confirms it's as designed, and decides to keep it.

Appendix 2

Disclaimer

This report is based on the scope of materials and documents provided, with a limited review at the time provided. Results may not be complete and do not include all vulnerabilities. The review and this report are provided on an as-is, where-is, and as-available basis. You agree that your access and/or use, including but not limited to any associated services, products, protocols, platforms, content, and materials, will be at your own risk. A report does not imply an endorsement of any particular project or team, nor does it guarantee its security. These reports should not be relied upon in any way by any third party, including for the purpose of making any decision to buy or sell products, services, or any other assets. TO THE FULLEST EXTENT PERMITTED BY LAW, WE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, IN CONNECTION WITH THIS REPORT, ITS CONTENT, RELATED SERVICES AND PRODUCTS, AND YOUR USE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NOT INFRINGEMENT.