FESSCHAIN SMART CONTRACT AUDIT REPORT

by QuillAudits, July 2019

Introduction :

This Audit Report highlights the overall security of **FessChain** Smart Contract. With this report, we have tried to ensure the reliability of their smart contract by complete assessment of their system's architecture and the smart contract codebase.

Auditing Approach and Methodologies applied :

Quillhash team has performed thorough testing of the project starting with analysing the code design patterns in which we reviewed the smart contract architecture to ensure it is structured and safe use of third party smart contracts and libraries.

Our team then performed a formal line by line inspection of the Smart Contract in order to find any potential issues like race conditions, transaction-ordering dependence, timestamp dependence, and denial of service attacks.

In the Unit testing Phase we coded/conducted Custom unit tests written for each function in the contract to verify that each function works as expected. In Automated Testing, We tested the Smart Contract with our in-house developed tools to identify vulnerabilities and security flaws.

The code was tested in collaboration of our multiple team members and this included -

1. Testing the functionality of the Smart Contract to determine proper logic has been followed throughout.
2. Analyzing the complexity of the code by thorough, manual review of the code, line-by-line.
3. Deploying the code on testnet using multiple clients to run live tests
4. Analysing failure preparations to check how the Smart Contract performs in case of bugs and vulnerabilities.
5. Checking whether all the libraries used in the code are on the latest version.
6. Analyzing the security of the on-chain data.

Audit Details

- Project Name: FESSCHAIN
- website/Etherscan Code : website
- Languages: Solidity (Smart contract), Javascript (Unit Testing)

Summary of FESSCHAIN Smart Contract :

QuillAudits conducted a security audit of a smart contract of FESSCHAIN. Fesschain smart contract is used to create the ERC20 token which is a FESS Token, Smart contract contain basic functionalities of ERC20 token with total supply of 10b and some advance functionalities of sending tokens of tokenomics and their locking and releasing period.

Total Supply : 10000000000 (10 b)
Token Name : FESS
Token Symbol : FESS
Decimal : 18

Tokenomics

Total Supply : 10000000000 (10 b)

Tokens For Sale = 600000000
Team Tokens = 2400000000
Maintenance Tokens = 1000000000
Marketing Tokens = 10000000
AirDrop In IEO Tokens = 20000000
Bounty In IEO Tokens = 30000000
Minting Tokens = 2250000000
AirDrop With Dapps Tokens = 3690000000

There is a lock period for marketing tokens for 8 months, team tokens will be initially sent 5% and quarterly team members can call function to **withdraw** 10% of tokens.

Audit Goals

The focus of the audit was to verify that the smart contract system is secure, resilient and working according to its specifications. The audit activities can be grouped in the following three categories:

Security: Identifying security related issues within each contract and within the system of contracts.

Sound Architecture: Evaluation of the architecture of this system through the lens of established smart contract best practices and general software best practices.

Code Correctness and Quality: A full review of the contract source code. The primary areas of focus include:

- Correctness
- Readability
- Sections of code with high complexity
- Quantity and quality of test coverage

Security Level references :

Every issue in this report was assigned a severity level from the following:

High severity issues will bring problems and should be fixed.

Medium severity issues could potentially bring problems and should eventually be fixed.

Low severity issues are minor details and warnings that can remain unfixed but would be better fixed at some point in the future.

Number of issues per severity

	Low	Medium	High
Open	0	0	0
Closed	0	0	0

Unit Testing

Test Suite

Contract: FessChain Token Contracts

✓ Should correctly initialize constructor values of FessChain Token Contract (314ms)

✓ Should check the Total Supply of FessChain Tokens (89ms)

✓ Should check the Name of a token of FESS contract (63ms)

✓ Should check the symbol of a token of Fesschain contract (69ms)

✓ Should check the decimal of a token of Fesschainf contract (66ms)

✓ Should check the balance of an Owner (49ms)

✓ Should check Tokens for Sale of Fess Chain

✓ Should check Team Tokens of Fess Chain (49ms)

✓ Should check Maintenance Tokens of Fess Chain (68ms)

✓ Should check Marketing Tokens of Fess Chain (60ms)

✓ Should check air Drop IEO Tokens of Fess Chain (39ms)

✓ Should check bounty In IEO Tokens of Fess Chain (48ms)

✓ Should check Minting Tokens of Fess Chain (42ms)

✓ Should check airDrop With Dapps Tokens of Fess Chain (40ms)

✓ Should check tokens Released of Fess Chain

✓ Should be able to pause Fesschain Token contract (174ms)

✓ Should Not be able to transfer tokens to accounts[1] of Sale tokens when paused (161ms)

✓ Should be able unPause Token contract (175ms)

✓ Should be able to transfer tokens to accounts[1] of Sale tokens only by Owner when not paused (490ms)

✓ Should be able to pause Fesschain Token contract (179ms)

✓ Should Not be able to send marketing Tokens by owner When it is Paused (133ms)

✓ Should be able unPause Token contract (268ms)

✓ Should be able to send marketing Tokens by Non Owner Account (227ms)

✓ Should be able to send marketing Tokens by owner only (240ms)

✓ Should be able to check marketing Token after sent to beneficiary (40ms)

✓ Should Not be able to transfer tokens to accounts[6] by marketing token holder before 8 months (129ms)

✓ should Approve address to spend specific token (103ms)

✓ Should be able to check marketing Token holder or not

✓ Should be able to pause Fesschain Token contract (133ms)

✓ Should Not be able to send Maintenance Tokens by owner when paused (123ms)

✓ Should be able unPause Token contract (132ms)

✓ Should Not be able to send Maintenance Tokens by owner Non owner when not paused (120ms)

✓ Should be able to send Maintenance Tokens by owner only when not paused (206ms)

✓ Should be able to check Maintenance Token after sent to beneficiary

✓ Should be able to pause Fesschain Token contract (122ms)

✓ Should Not be able to send Airdrop In IEO Tokens by owner when paused (101ms)

✓ Should be able unPause Token contract (125ms)

✓ Should Not be able to send Airdrop In IEO Tokens by Non owner account (95ms)

✓ Should be able to send Airdrop In IEO Tokens by owner only (185ms)

✓ Should be able to check Air drop in IEO after sent to beneficiary

✓ Should Not be able to Send bounty In IEO In IEO Tokens by Non owner account (93ms)

✓ Should be able to Send bounty In IEO In IEO Tokens by owner only (213ms)

✓ Should be able to check bounty in IEO after sent to beneficiary

✓ Should Not be able to Send Minting Tokens by Non owner account (203ms)

✓ Should be able to Send Minting Tokens by owner only (280ms)

✓ Should be able to check Minting tokens after sent to beneficiary

✓ Should Not be able to Send Air Drop With Dapps tokens by Non owner account (105ms)

✓ Should be able to Send Airdrop With Dapps tokens by owner only (228ms)

✓ Should be able to check airDropWithDappsLater after sent to beneficiary

✓ Should be able to pause Fesschain Token contract (142ms)

✓ Should Not be able unPause Token contract from Non Owner Accounts (73ms)

✓ Should be able unPause Token contract (138ms)

✓ should Approve address to spend specific token (99ms)

✓ should Increase the Approval (180ms)

✓ should Decrease the Approval (295ms)

✓ Should be able to transfer ownership of token Contract (79ms)

✓ Should be able to Accept ownership of token Contract (150ms)

✓ Should be able to transfer Tokens on the behalf of accounts[0] (209ms)

✓ Should be able to pause Fesschain Token contract (136ms)

✓ Should be able to send team Tokens by owner only (171ms)

✓ Should be able unPause Token contract (137ms)

✓ Should be able to send team Tokens by owner only (540ms)

✓ Should Not be able to withdraw tokens when caller is not a team member (83ms)

✓ Should be able to withdraw 10% tokens after 3 months (269ms)

✓ Should be able to withdraw 20% tokens after 6 months (246ms)

✓ Should be able to withdraw 30% tokens after 9 months (325ms)

✓ Should be able to withdraw 40% tokens after 12 months (241ms)

✓ Should be able to withdraw 50% tokens after 15 months (280ms)

✓ Should be able to withdraw 60% tokens after 18 months (237ms)

✓ Should be able to withdraw 70% tokens after 21 months (242ms)

✓ Should be able to withdraw 80% tokens after 24 months (204ms)

✓ Should be able to withdraw 90% tokens after 27 months (217ms)

✓ Should be able to pause Fesschain Token contract (109ms)

✓ Should Not be able to burn tokens when paused (64ms)

✓ Should be able unPause Token contract (107ms)

✓ Should be able to withdraw 100% tokens after 30 months (197ms)

✓ Should Not be able to withdraw tokens after all tokens sent (135ms)

✓ Should be able to burn tokens (108ms)

✓ Should check the Total Supply of FessChain Tokens after token burnt

✓ Should be able check owner of the contract (66ms)

Contract: FessChain Token Contract, Test for Maximum values

✓ Should correctly initialize constructor values of FessChain Token Contract (136ms)

✓ Should check Tokens for Sale of Fess Chain (44ms)

✓ Should check Team Tokens of Fess Chain

✓ Should check Maintenance Tokens of Fess Chain

✓ Should check Marketing Tokens of Fess Chain

✓ Should check air Drop IEO Tokens of Fess Chain (43ms)

✓ Should check bounty In IEO Tokens of Fess Chain

✓ Should check Minting Tokens of Fess Chain

✓ Should check airDrop With Dapps Tokens of Fess Chain (64ms)

✓ Should check tokens Released of Fess Chain (45ms)

✓ Should be able to transfer tokens to accounts[1] of Sale tokens only by Owner when not paused (184ms)

✓ Should be able to send marketing Tokens by owner only (175ms)

✓ Should be able to check marketing Token after sent to beneficiary

✓ Should be able to send Maintainance Tokens by owner only when not paused (159ms)

✓ Should be able to check Maintainance Token after sent to beneficiary

✓ Should be able to send Air Drop In IEO Tokens by owner only (244ms)

✓ Should be able to check Air drop in IEO after sent to beneficiary (285ms)

✓ Should be able to Send Minting Tokens by owner only (279ms)

✓ Should be able to check Minting tokens after sent to beneficiary (38ms)

✓ Should be able to Send bounty In IEO Tokens by owner only (230ms)

✓ Should be able to check bounty in IEO after sent to beneficiary

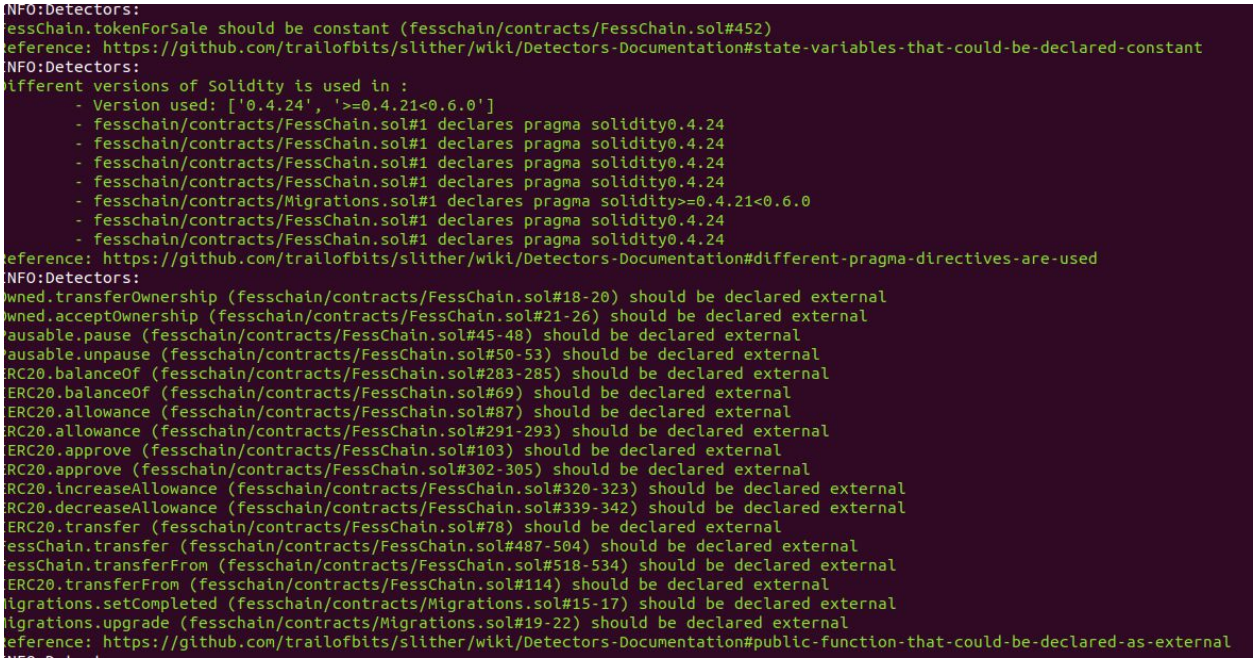
✓ Should be able to send team Tokens by owner only (268ms)

Final Result of Test:

✓ 102 Passing (16s) PASSED

❌ 0 Failed

Slither Tool Result :



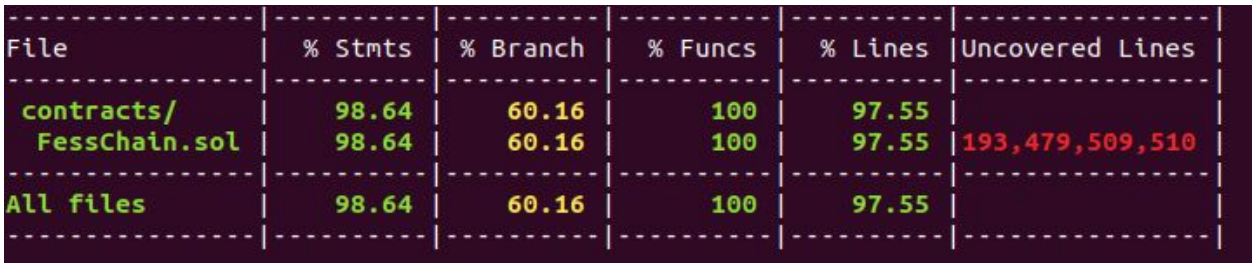
Manual Transactions

Network : Ropsten, Remix ethereum

Create Token contract	0x8593b36a2b068ff5b095c696a016e42e7cd73a8b3f2ac2d71e120ca16767eeb1
Pause function	0x206c40c64a941ae9aba68a5f45e5ae6034206398d9fbe89285b3f3c270d4fdb2
unPause function	0x621558b5259e3bd9acf3cf8e53b928e764487a85fae266d18e122a1dc586aff0
Owner transfer tokens	0xf2b5d7247a8feaf8ea8c10da3a377da6d4ab1609112b8463ac4f0a48b63d853e
Send airDrop/bounty dapps tokens	0x4eb6222abce950fec0767953fe72280b98b9075df68ffa872bbad98af310a288
transferOwnership	0x9778f6fab4ca34a3c9c5286694291e247fee998057c99d9739d1102bd48dabcd
Accept ownership	0x810549397e484607335d505b0a24acd0efa5b1dddc10a4b5f3c0bf274d1a59fe
sendAirDropIEO	0xe5bf489e0189e0b5a7e6087d69c47b9990ea557dc9bd2ca73a83a37cd16bb737

sendBountyIEO	0xd7869ce7688d8c21369cd31f1ba1b589bc8b430966a89e8e75ea90a8c1733a5e
sendmaintainanceTokens	0xfbdae3b2faa708e7a9f718e30493f5b4adfb8c0d54c2525a592ca8edf5359b92
sendMarketingTokens	0xa43a5cbd5001094a6417ede0a1866ec505ef52af9c7b647388a9f63c5413ba1a
transfer marketing tokens before 8 months	0x5c2d6ca9dd0b4b5db4277e8e19be89d2fc5588d2d4d5d6d95f15dfd550d16a79
sendTeamTokens	0x92c35d288950d97f87a537b0a2e3a1cc61084f1acce194c459c327fcfadf9a4a
Cannot withdraw tokens before 3 months	0x04fca619327880ab090a3285c5290cdf6f67b96b90ce235ce4af99da83474c72

Coverage Report



File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Lines
contracts/	98.64	60.16	100	97.55	
FessChain.sol	98.64	60.16	100	97.55	193,479,509,510
All files	98.64	60.16	100	97.55	

Coverage report defines how much our test cases touching solidity code.

Our test cases covered 100% functions and 97.55% of line of code of solidity contract.

98.64% of statements are covered with unit testing

Coverage report give assurance that our unit testing is 100% touching smart contract code.

Surya Tool Result

File Name	SHA-1 Hash
FessChain.sol	c4cc1482011d1bb2515dbb459e34743b7cdb3a3c

Contracts Description Table

Contract	Type	Bases		
└	**Function Name**	**Visibility**	**Mutability**	**Modifiers**
FessChain	Implementation	ERC20		
└	\<Constructor\>	Public ❗	🛑	Owned
└	transfer	Public ❗	🛑	whenNotPaused
└	transferFrom	Public ❗	🛑	whenNotPaused
└	sendTeamTokens	External ❗	🛑	whenNotPaused onlyOwner
└	sendMarketingTokens	External ❗	🛑	whenNotPaused onlyOwner
└	sendMaintainanceTokens	External ❗	🛑	whenNotPaused onlyOwner
└	sendAirDropIEO	External ❗	🛑	whenNotPaused onlyOwner
└	sendBountyIEO	External ❗	🛑	whenNotPaused onlyOwner
└	sendAirDropAndBountyDapps	External ❗	🛑	whenNotPaused onlyOwner
└	sendMintingTokens	External ❗	🛑	whenNotPaused onlyOwner
└	burn	External ❗	🛑	whenNotPaused
└	withdrawTeamTokens	External ❗	🛑	whenNotPaused

Legend

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable